

CanSAS 1-D Data Format Manual, v1.0

COLLABORATORS

	<i>TITLE :</i> CanSAS 1-D Data Format Manual, v1.0		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Pete R. Jemian	September 25, 2009	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Wiki Manual	1
1.1	Disclaimer	1
1.2	Objective	1
1.2.1	Status	1
1.3	General Layout of the XML Data	1
1.3.1	Overview	2
1.3.1.1	Basic elements of the cansas1d/1.0 standard	3
1.3.1.2	Required XML file header	3
1.3.2	Rules	3
1.3.3	Compatibility of Geometry Definitions	6
1.3.4	Converting data into the XML format	7
1.4	Documentation and Definitions	7
1.4.1	XML Schema	7
1.4.2	XML Stylesheets	7
1.4.3	Examples and Case Studies	8
1.4.3.1	XML layout for multiple experiments	8
1.4.4	Foreign Elements	9
1.4.5	Support tools for Visualization & Analysis software	9
1.4.6	Software repositories (for canSAS1d/1.0 standard)	9
1.5	Validation of XML against the Schema	9
1.6	Help for XML	10
2	Elements of the CanSAS XML standard	11
2.1	"{any}" element	11
3	Bindings and Software Support	12
3.1	Fortran binding	12
3.1.1	Software Development Kits	12
3.1.2	canSAS 1-D SAS XML v1.0 support	12
4	Other matters	13
4.1	XML Help	13

List of Figures

1.1	block diagram of minimum elements required for cansas1d/1.0 standard	2
1.2	definition of Q geometry for small-angle scattering	4
1.3	definition of translation and orientation geometry as viewed from the detector towards the source	5
1.4	definition of translation and orientation geometry as viewed from the source towards the detector	6

List of Tables

1.1 Basic elements of the CanSAS 1-D standard 3

Preface



CanSAS 1-D Data Format Manual, v1.0

Chapter 1

Wiki Manual

1.1 Disclaimer

This description is meant to inform the community how to layout the information within the XML files. However, should the information in this document and the [cansas1d/1.0 SAS XML Schema](http://svn.smallangles.net/trac/canSAS/browser/ldwg/trunk/cansas1d/1.0/SAS_XML_Schema) (http://svn.smallangles.net/trac/canSAS/browser/ldwg/trunk/cansas1d/1.0/SAS_XML_Schema) differ, the XML Schema will be deemed to have the most correct description of the standard.

1.2 Objective

One of the first aims of the canSAS (Collective Action for Nomadic Small-Angle Scatterers) forum of users, software developers, and facility staff was to discuss better sharing of SAS data analysis software. [CanSAS](http://www.smallangles.net/canSAS) (<http://www.smallangles.net/canSAS>) identified that a significant need within the SAS community can be satisfied by a robust, self-describing, text-based, standard format to communicate reduced one-dimensional small-angle scattering data, "I(Q)", between users of our facilities. Our goal has been to define such a format that leaves the data file instantly human-readable, editable in the simplest of editors, and importable by simple text import filters in programs that need not recognise advanced structure in the file nor require advanced programming interfaces. The file should contain both the primary data of "I(Q)" and also any other descriptive information (metadata) about the sample, measurement, instrument, processing, or analysis steps.

The cansas1d/1.0 standard meets the objectives for a 1D standard, incorporating metadata about the measurement, parameters and results of processing or analysis steps. Even multiple measurements (related or unrelated) may be included within a single XML file.

1.2.1 Status

Version 1.0 was tagged from the subversion repository on 2009-05-12 as no changes were committed since January 2009. Use this command to checkout the tagged release.

```
svn checkout http://svn.smallangles.net/svn/canSAS/ldwg/tags/v1.0 cansas1dwg-1.0
```

1.3 General Layout of the XML Data

The canSAS 1-D standard for reduced 1-D SAS data is implemented using XML files. A single file can contain SAS data from a single experiment or multiple experiments. All types of relevant data ("I(Q)", metadata) are described for each experiment. More details are provided below.

1.3.1 Overview

The basic elements of the cansas1d/1.0 standard are shown in the following table. After an XML header, the root element of the file is `[[cansas1d_SASroot | SASroot]]` which contains one or more `[[cansas1d_SASentry | SASentry]]` elements, each of which describes a single experiment (data set, time-slice, step in a series, new sample, etc.). Details of the `[[cansas1d_SASentry | SASentry]]` element are also shown in the next figure. Refer to the `[[cansas1d_block_diagrams | block diagrams]]` for alternative depictions. See <http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d.xml> for an example XML file. Examples, Case Studies, and other background information are below. More discussion can be found on the `[[1D_Data_Formats_Working_Group|canSAS 1D Data Formats Working Group]]` page and its `[[Talk:1D_Data_Formats_Working_Group|Talk:1D_Data_Formats_Working_Group]]` page. Details about each specific field (XPath string, XML elements and attributes) are described on the `[[cansas1d_definition_of_terms|cansas1d_definition_of_terms]]` page.

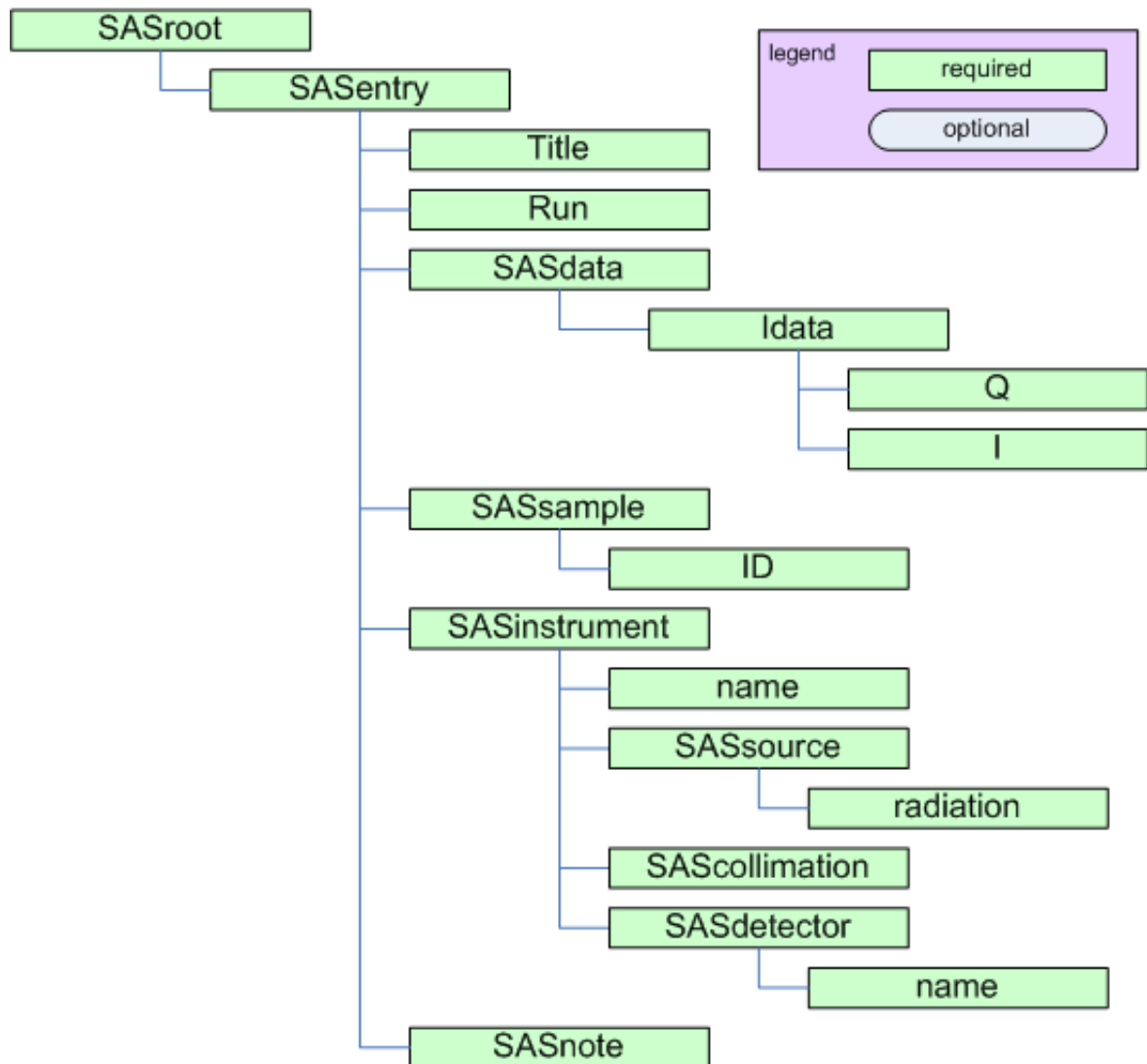


Figure 1.1: block diagram of minimum elements required for cansas1d/1.0 standard

- SASroot: the root element of the file (after the XML header)
- SASentry: describes a single experiment (data set, time-slice, step in a series, new sample, etc.)
- block diagrams
- cansas1d.xml example XML file

- discussion of this format: basic more
- Seek outside help for XML
- Definition of terms: Details about each specific field (XPath string, XML elements and attributes)

1.3.1.1 Basic elements of the cansas1d/1.0 standard

Table 1.1: Basic elements of the CanSAS 1-D standard

Element	Description
XML Header	descriptive info required at the start of every XML file
SASroot	root element of XML file
SASentry	data set, time-slice, step in a series, new sample, etc.
:::Title	for this particular SASentry
:::Run	run number or ID number of experiment
:::[cansas1d_any {any}]	any non-cansas1d/1.0 element can be used at this point
:::[cansas1d_SASdata SASdata]	this is where the reduced 1-D SAS data is stored
:::[cansas1d_SASdata Idata]	a single data point in the dataset
:::[cansas1d_any {any}]	any non-cansas1d/1.0 element can be used at this point
:::[cansas1d_SASsample SASsample]	description of the sample
:::[cansas1d_SASinstrument SASinstrument]	description of the instrument
:::[cansas1d_SASsource SASsource]	description of the source
:::[cansas1d_SAScollimation SAScollimation]	description of the collimation
:::[cansas1d_SASdetector SASdetector]	description of the detector
:::[cansas1d_SASprocess SASprocess]	for each processing or analysis step
:::[cansas1d_SASnote SASnote]	anything at all

1.3.1.2 Required XML file header

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="cansasxml-html.xsl" ?>
<SASroot version="1.0"
  xmlns="cansas1d/1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="cansas1d/1.0
    http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansas1d.xsd"
  >
```

1.3.2 Rules

1. canSAS1d/1.0 XML data files will adhere to the standard if they can successfully [[cansas1d_documentation#Validation_of_XML_ | validate]] against the established XML Schema ([http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d.xsd cansas1d.xsd])
- 2.

$$Q = (4\pi/\lambda)\sin(\theta) \quad (1.1)$$

where λ is the wavelength of the radiation

and 2θ is the angle through which the detected radiation has been scattered.

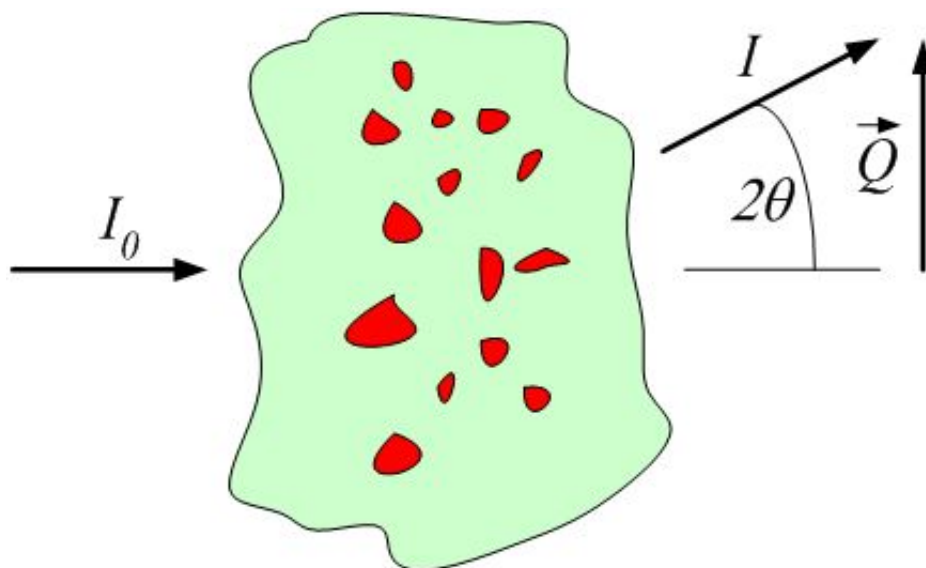


Figure 1.2: definition of Q geometry for small-angle scattering

3. units to be given in standard SI abbreviations (eg, m, cm, mm, nm, K) with the following exceptions:
 - a. um=micrometres
 - b. C=celsius
 - c. A=Angstroms
 - d. percent=%.
 - e. fraction
 - f. a.u.=arbitrary units
 - g. none=no units are relevant (such as dimensionless)
4. where reciprocal units need to be quoted the format shall be "1/abbreviation"
5. when raised to a power, use similar to "A^3" or "1/m^4" (and not "A3" or "m-4")
6. axes:
 - a. z is along the flight path (positive value in the direction of the detector)
 - b. x is orthogonal to z in the horizontal plane (positive values increase to the right when viewed towards the incoming radiation)
 - c. y is orthogonal to z and x in the vertical plane (positive values increase upwards)

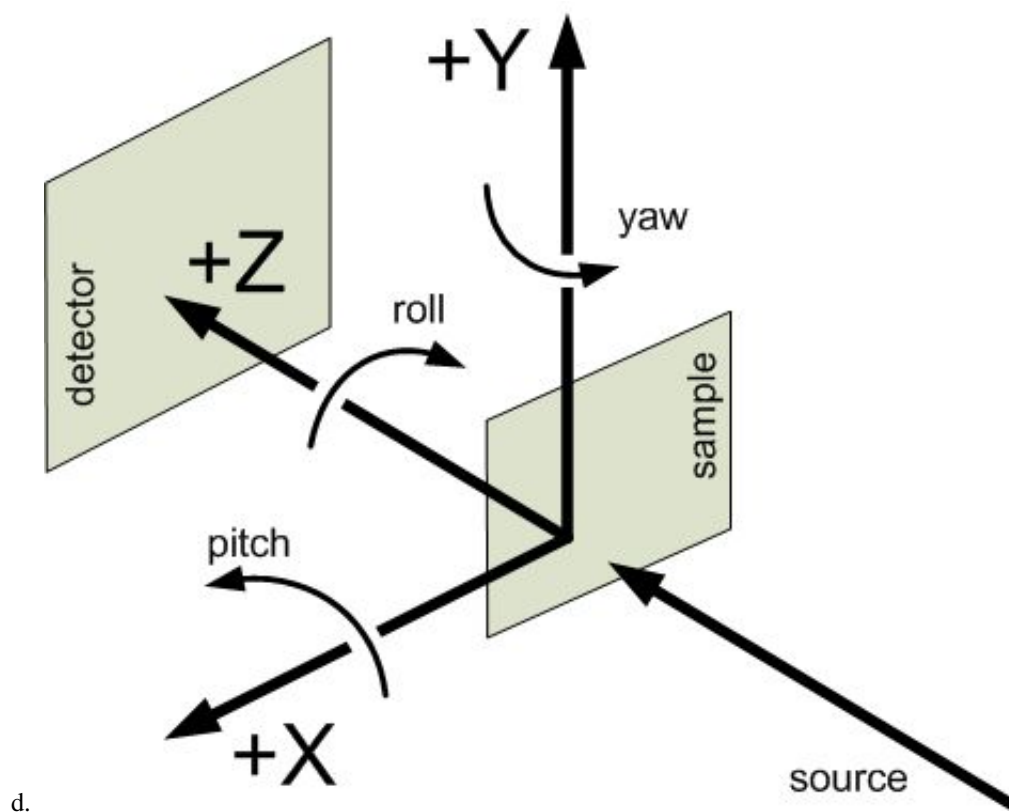


Figure 1.3: definition of translation and orientation geometry as viewed from the detector towards the source

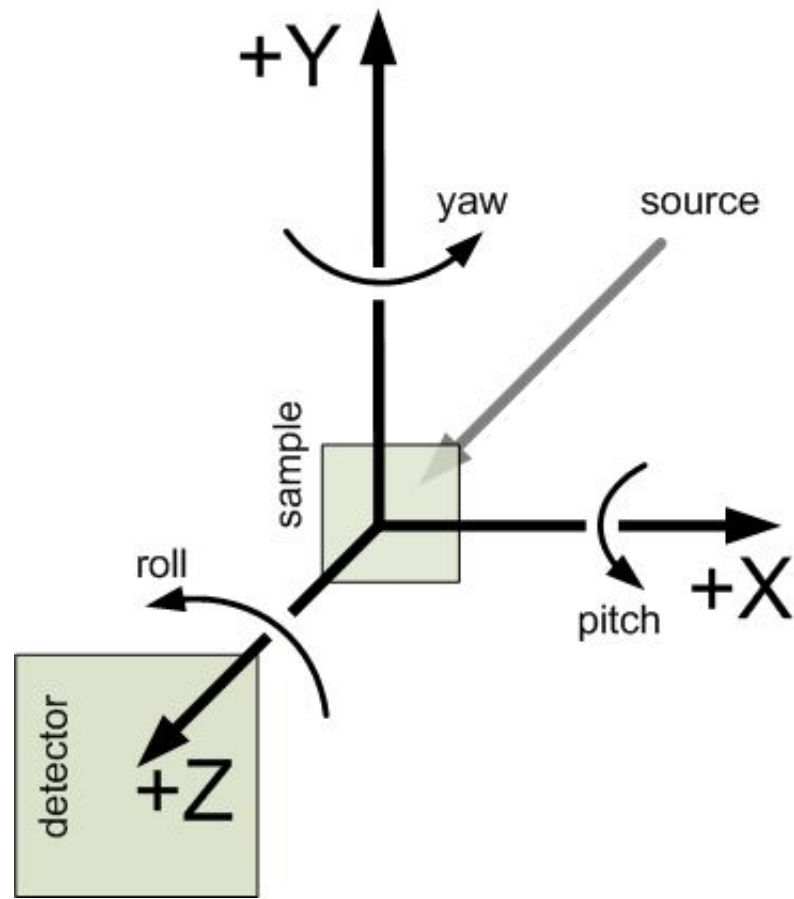


Figure 1.4: definition of translation and orientation geometry as viewed from the source towards the detector

7. orientation (angles) describes one-axis rotations (rotations about multiple axes require more information):
 - a. roll is about z
 - b. pitch is about x
 - c. yaw is about y
8. Unicode characters MUST NOT be used
9. Binary data is not supported

1.3.3 Compatibility of Geometry Definitions

Note: translation and orientation geometry used by canSAS are consistent with:

- http://en.wikipedia.org/wiki/Cartesian_coordinate_system
- http://en.wikipedia.org/wiki/Right-hand_rule
- http://www.nexusformat.org/Coordinate_Systems
- <http://mcstas.risoe.dk/documentation/tutorial/node6.html>
- <http://webhost5.nts.jhu.edu/reza/book/kinematics/kinematics.htm>

The translation and orientation geometry definitions used here are different than those used by "SHADOW" (<http://www.nanotech.wisc.edu>) where the "y" and "z" axes are swapped and the direction of "x" is changed.

1.3.4 Converting data into the XML format

A WWW form (<http://www.smallangles.net/canSAS/xmlWriter/> [canSAS/xmlWriter](http://www.smallangles.net/canSAS/xmlWriter/)) is available to translate three-column ASCII text data into the canSAS1d/1.0 XML format. This form will help you in creating an XML file with all the required elements in the correct places. The form requests the SAS data of Q, I, and Idev (defined elsewhere on this page) and some basic metadata (title, run, sample info, ...). Press the `<nowiki>Submit</nowiki>` button and you will receive a nicely formatted WWW page with the SAS data. If you then choose "View page source" (from one of your browser menus), you will see the raw XML of the canSAS1d/1.0 XML format and you can copy/paste this into an XML file.

The SAS data that you paste into the form box is likely to be copied directly from a 3-column ASCII file from a text editor. Line breaks are OK, they will be treated as white-space as will tabs and commas. Do not be concerned that the data looks awful in the form entry box, just check the result to see that it comes out OK.

1.4 Documentation and Definitions

- **Documentation:** [\[\[cansas1d_documentation\]\]](#) (this page)
- **Definitions:** [\[\[cansas1d_definition_of_terms\]\]](#)
- **Block diagrams:** [\[\[cansas1d_block_diagrams\]\]](#)

1.4.1 XML Schema

The <http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d.xsd> [cansas1d.xsd](http://www.w3schools.com/xsd/XMLSchema) [[http://www.w3schools.com/xsd XML Schema](http://www.w3schools.com/xsd/XMLSchema)] defines the rules for the XML file format (<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d.xsd> TRAC), [<http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansas1d.xsd> SVN]) and is used to validate any XML file for adherence to the format.

1.4.2 XML Stylesheets

- `'''cansasxml-html.xml'''`: [http://www.w3schools.com/xsl/ XSLT stylesheets](http://www.w3schools.com/xsl/XSLTstylesheets)] can be used to extract metadata or to convert into another file format. The default canSAS stylesheet [<http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansasxml-html.xml> [cansasxml-html.xml](http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansasxml-html.xml)] should be copied into each folder with canSAS XML data file(s). It can be used to display the data in a supporting WWW browser (such as Firefox or Internet Explorer) or to import into Microsoft Excel (with the added XML support in Excel). (See the excellent write-up by Steve King, ISIS, at http://www.isis.rl.ac.uk/archive/LargeScale/LOQ/xml/cansas_xml_for_an_example.) By default, MS Windows binds `'''*.xml'''` files to start Internet Explorer. Double-clicking on a canSAS XML data file with the [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansasxml-html.xml> `'''cansasxml-html.xml'''`] stylesheet in the same directory will produce a WWW page with the SAS data and selected metadata.
- Suggestions for support software that writes canSAS1d/1.0 XML data files:
 - be sure to update to the latest SVN repository revision (command: `svn update`)
 - check the output directory to see if it contains the default XSLT file.
 - copy the latest XSLT file to the output directory if either:
 - * the output directory contains an older revision
 - * the output directory does not have the default XSLT file
 - The most recent XSLT file can be identified by examining the file for the `'''$ Revision: '''` string. For example:

```
# \ $Revision: 95 $
```

is version 66 (updated 2009-01-12).

1.4.3 Examples and Case Studies

- [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d.xml> cansas1d.xml] basic example: Note that, for clarity, only one row of data is shown. This is probably a very good example to use as a starting point for creating XML files with a text editor.
- [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/bimodal-test1.xml> bimodal-test1.xml]: Simulated SAS data to test size distribution calculation routines.
- [[Glassy_Carbon_Round_Robin | Glassy Carbon Round Robin]]: Glassy carbon samples measured at several facilities world-wide.
- [[cansas1d_casestudy_collagen | dry chick collagen]]: illustrates the minimum information necessary to meet the requirements of the standard format
- [[cansas1d_casestudy_af1410 | AF1410 steel]]: SANS study using magnetic contrast variation (with multiple samples and multiple data sets for each sample), the files can be viewed from TRAC (no description yet): [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/examples/af1410/>]
- [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cansas1d-template.xml> cansas1d-template.xml]: This is used to test all the rules in the XML Schema. This is probably not a very good example to use as a starting point for creating XML files with a text editor since it tests many of the special-case rules.

1.4.3.1 XML layout for multiple experiments

Each experiment is described with a single "SASentry" element. The fragment below shows how multiple experiments can be included in a single XML file. Full examples of canSAS XML files with multiple experiments include:

- [<http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/W1W2.XML> ISIS LOQ SANS instrument]
- [http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/cs_af1410.xml NIST SANS data]

Here is a brief sketch of how a file would be arranged with multiple SASentry elements and multiple SASdata elements.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="cansasxml-html.xsl" ?>
<SASroot version="1.0"
  xmlns="cansas1d/1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="cansas1d/1.0 http://svn.smallangles.net/svn/canSAS/1dwg/trunk/ ↵
    cansas1d.xsd"
>
  <SASentry name="071121.dat#S22">
    <!-- contents of the first experiment in the file go here -->
  </SASentry>
  <SASentry name="example temperature series">
    <!-- example with two SAS data sets related to the same sample -->
    <Title>title of this series</Title>
    <Run name="run1">42-001</Run>
    <Run name="run2">42-002</Run>
    <SASdata name="run1">
      <!-- data from 42-001 run comes here -->
    </SASdata>
    <SASdata name="run2">
      <!-- data from 42-002 run comes here -->
    </SASdata>
    <!-- other elements come here for this entry -->
  </SASentry>
  <SASentry name="other sample">
    <!-- any number of additional experiments can be included, as desired -->
    <!-- SASentry elements in the same XML file do not have to be related -->
```

```
</SASentry>  
</SASroot>
```

1.4.4 Foreign Elements

To allow for inclusion of elements that are not defined by the cansas1d.xsd XML Schema, XML "foreign elements" are permitted at select locations in the cansas1d/1.0 format. Please refer to the references (and others) [[#Help_for_XML | below]] for deeper discussions on foreign elements.

No examples exist.

Note

2009-09, PRJ: Actually, the ISIS glassy carbon data is now an example

At present, all examples of canSAS xml files using foreign namespaces have been converted to bring that data into either the ""SASprocessnote"" or ""SASnote"" elements. Refer to the [http://svn.smallangles.net/trac/canSAS/changeset/47 TRAC changes] for an example of arranging the content in ""SASprocessnote"" to avoid the use of foreign namespace elements.

1.4.5 Support tools for Visualization & Analysis software

Support for importing canSAS1d/1.0 files exists for these languages:

Note

Refactor the wiki pages here and link as appropriate.

- **FORTRAN**
- **IgorPro**
- **Java**
- **Microsoft Excel**
- **PHP**
- **Python**
- **XSLT** (useful in a web browser)

1.4.6 Software repositories (for canSAS1d/1.0 standard)

- **TRAC:** <http://svn.smallangles.net/trac/canSAS/browser/1dwg/tags/v1.0>
- **Subversion:** <http://svn.smallangles.net/svn/canSAS/1dwg/tags/v1.0>

1.5 Validation of XML against the Schema

1. open browser to: <http://www.xmlvalidation.com/>
 2. paste content of candidate XML file (with reference in the header to the XML Schema as shown above) into the form
 3. press <validate>
 4. paste content of [<http://svn.smallangles.net/svn/canSAS/1dwg/trunk/cansas1d.xsd> cansas1d.xsd] XSD file into form and press <continue validation>
 5. check the results
-

1.6 Help for XML

The various references for help on XML have been moved to their own wiki page: [\[\[XmlHelp\]\]](#)

Note

chapter="Other" section="XML Help" xml:id="wiki-XML_Help"

Chapter 2

Elements of the CanSAS XML standard

There are various elements (tag names) in the canSAS1d/1.0 standard. Each of these is described below.

2.1 "{any}" element

Table 2.1:

Name	Type	occurrence	Description	Attributes
"{any}"	container	[0..unbounded]	Any element(s) not defined in the cansas1d/1.0 standard can be placed at this point. (These are called <i>foreign</i> elements. It is suggested to associate foreign elements with a foreign namespace to differentiate them from the canSAS elements in the XML file.)	<i>xmlns:{foreign-prefix}</i> ={ foreign-namespace }

Chapter 3

Bindings and Software Support

Bindings (import/export drivers) and other software support have been created and contributed. These are listed here by the language or software environment.

3.1 Fortran binding

The development of the FORTRAN language, so beloved of scientists, pre-dates the development of XML. And it shows. FORTRAN is not a language that manipulates strings with ease, and this makes parsing XML decidedly awkward. So unless you *really* have to use FORTRAN, you are probably better off with C/C++ (or something else more 'modern'), see for example Daniel Veillard's LIBXML2 library at <http://xmlsoft.org/> or Frank van den Berghen's parser at <http://www.applied-mathematics.net/tools/xmlParser.html>.

If you have to use a dialect earlier than FORTRAN-90 (F90), then the chances are you will have to code your own parser.

3.1.1 Software Development Kits

For later dialects, there are some SDK's available on the Web:

- F90:
 - XMLPARSE - by Arjen Markus at <http://xml-fortran.sourceforge.net/>
 - FoX - by Toby White others at <http://uszla.me.uk/space/software/FoX/>
- For F95:
 - XML - by Mart Rentmeester at <http://nn-online.org/code/xml/>

3.1.2 canSAS 1-D SAS XML v1.0 support

Steve King[<mailto:s.m.king@rl.ac.uk>] (ISIS) has provided a F77 routine (*SASXML_G77.F*) that will read CanSAS XML v1.0 files.

Chapter 4

Other matters

Various topics have been considered or presented in considering this standard. Some are described below.

4.1 XML Help

Listed below are various references useful in learning XML and related topics.

- **XML:** eXtensible Markup Language
 - <http://www.w3schools.com/xml/>
 - <http://www.w3.org/XML/>
 - <http://en.wikipedia.org/wiki/XML>
 - <http://www.zvon.org/xxl/XPathTutorial/General/examples.html>
 - **XSL (or XSLT):** eXtensible Stylesheet Language (Transformation)
 - <http://www.w3schools.com/xsl/>
 - <http://www.w3.org/Style/XSL/>
 - http://en.wikipedia.org/wiki/Extensible_Stylesheet_Language
 - <http://en.wikipedia.org/wiki/XSLT>
 - **XPath:** XPath is a language for finding information in an XML document.
 - <http://www.w3schools.com/xpath/>
 - <http://www.w3.org/Style/XSL/>
 - <http://en.wikipedia.org/wiki/XPath>
 - **Schema:** An XML Schema describes the structure of an XML document.
 - <http://www.w3schools.com/schema/>
 - <http://www.w3.org/XML/Schema>
 - <http://en.wikipedia.org/wiki/XSD>
 - **XML Namespaces:** XML namespaces are used for providing uniquely named elements and attributes in an XML instance.
 - <http://www.zvon.org/xxl/Namespacetutorial/Output>
 - http://en.wikipedia.org/wiki/XML_namespaces
 - http://www.w3schools.com/XML/xml_namespaces.asp
-

- **XML Foreign Elements:** Inclusion of elements, at select locations, that are not defined by the cansas1d.xsd XML Schema
 - <http://books.xmlschemata.org/relaxng/relax-CHP-11-SECT-4.html>
 - <http://www.w3.org/TR/SVG/extend.html>
 - <http://www.google.com/search?q=XML+foreign+elements>