cansas1d binding Python

From canSAS

Contents

NIST/NCNR support

Specific support (http://danse.us/trac/sans) for the cansas1d/1.0 data standard in Python is being developed by NIST/NCNR (http://www.ncnr.nist.gov/) as part of their contribution to the DANSE (http://wiki.cacr.caltech.edu/danse) project.

Here are some terse instructions to get you started:

```
svn co http://danse.us/trac/sans/browser/trunk/DataLoader DataLoader
cd DataLoader
python setup.py install
```

The release notes (http://danse.us/trac/sans/browser/trunk/DataLoader/release_notes.txt) have a list of the dependencies.

Comments

Other constructive suggestions (that predate the NIST/NCNR support) have been gathered on this page.

gnosis.xml.objectify

The GnosisUtils (http://freshmeat.net/projects/gnosisxml/) offer a method to read any XML file into Python data structures. This utility does not validate the XML against a specific XML Schema which can be both good (flexible, especially when XML Foreign Namespace elements are used) and not so good (XML content not guaranteed to be *by the rules*).

A quick test of this turned up an acceptable result in that it was able to read several of the canSAS test XML files, including those with foreign namespaces. And ***it was very easy.*** (Especially with some help from http://www.xml.com/pub/a/2003/07/02/py-xml.html)

Here is a quick example.

installation

Here is the condensed installation (without all that output) steps. Your system may have gnosis already installed. You may also need sysAdmin privileges. You may need ...

```
cd /tmp
wget http://freshmeat.net/redir/gnosisxml/22028/url_tgz/Gnosis_Utils-1.2.2.tar.gz
tar xzf Gnosis_Utils-1.2.2.tar.gz
cd Gnosis_Utils-1.2.2/
python setup.py install_all
```

quick test in Python

Here is the Python code (without all that output) (called *python-test.py*):

```
import gnosis.xml.objectify

sasxml = gnosis.xml.objectify.XML_Objectify('bimodal-test1.xml').make_instance()
print sasxml.SASentry.Title.PCDATA
print sasxml.SASentry.Run.PCDATA
print sasxml.SASentry.SASinstrument.name.PCDATA
print sasxml.SASentry.SASdata.Idata[0].Q.unit,  sasxml.SASentry.SASdata.Idata[0].I.unit
print sasxml.SASentry.SASdata.Idata[0].Q.PCDATA,  sasxml.SASentry.SASdata.Idata[0].I.PCDATA,  sasxml.SASentr
```

A revision of this example is on the TRAC site: http://svn.smallangles.net/trac/canSAS/browser/1dwg/trunk/python/python-test.py

execution of that Python code

```
[Pete@regitte,2441,cansas1dwg-regitte]$ ./python-test.py
SAS bimodal test1
1992
simulated SAS calculation
1/A 1/cm
0.0040157139 3497.473 90.72816
```

full session output

```
[Pete@regitte,2429,/tmp]$ cd /tmp
/tmp
[Pete@regitte,2430,/tmp]$ wget http://freshmeat.net/redir/gnosisxml/22028/url_tgz/Gnosis_Utils-1.2.2.tar.gz
--11:43:16--  http://freshmeat.net/redir/gnosisxml/22028/url_tgz/Gnosis_Utils-1.2.2.tar.gz
           => `Gnosis_Utils-1.2.2.tar.gz'
Resolving freshmeat.net... 66.35.250.168
Connecting to freshmeat.net|66.35.250.168|:80... connected.
HTTP request sent, awaiting response... 302 Found
Location: http://www.gnosis.cx/download/Gnosis_Utils.More/Gnosis_Utils-1.2.2.tar.gz [following]
--11:43:16--  http://www.gnosis.cx/download/Gnosis_Utils.More/Gnosis_Utils-1.2.2.tar.gz
           => `Gnosis_Utils-1.2.2.tar.gz'
Resolving www.gnosis.cx... 64.41.64.172
Connecting to www.gnosis.cx|64.41.64.172|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 287,989 (281K) [application/x-tar]

100%[==========================================================================================

11:43:16 (2.47 MB/s) - `Gnosis_Utils-1.2.2.tar.gz' saved [287989/287989]

[Pete@regitte,2431,/tmp]$ tar xzf Gnosis_Utils-1.2.2.tar.gz
[Pete@regitte,2432,/tmp]$ cd Gnosis_Utils-1.2.2/
/tmp/Gnosis_Utils-1.2.2
[Pete@regitte,2433,Gnosis_Utils-1.2.2]$ python setup.py install_all
[Pete@regitte,2434,Gnosis_Utils-1.2.2]$ cd ~/workspace/cansas1dwg-regitte
[Pete@regitte,2435,cansas1dwg-regitte]$ python
Python 2.5.1 (r251:54863, May 18 2007, 16:56:43)
[GCC 3.4.4 (cygming special, gdc 0.12, using dmd 0.125)] on cygwin
Type "help", "copyright", "credits" or "license" for more information.
>>> import gnosis.xml.objectify
>>> sasxml = gnosis.xml.objectify.XML_Objectify('bimodal-test1.xml').make_instance()
>>> print sasxml.SASentry.Title.PCDATA
SAS bimodal test1
>>> print sasxml.SASentry.Run.PCDATA
1992
>>> print sasxml.SASentry.SASinstrument.name.PCDATA
simulated SAS calculation
>>> print sasxml.SASentry.SASdata.Idata[0].Q.unit
1/A
>>> print sasxml.SASentry.SASdata.Idata[0].I.unit
1/cm
>>> print sasxml.SASentry.SASdata.Idata[0].Q.PCDATA,  sasxml.SASentry.SASdata.Idata[0].I.PCDATA,  sasxml.SAS
0.0040157139 3497.473 90.72816
```

Conclusion: OK

This has the promise of being a useful approach to reading this format in Python. Now, how to write back out... ?

generateDS.py

generateDS.py (http://www.rexx.com/~dkuhlman/, http://www.rexx.com/~dkuhlman/generateDS.html) can build a binding (map the structure of the XML file directly into a Python data structure) for Python from an XML Schema. However, the cansas1d/1.0 XML schema (cansas1d.xsd) does not seem to fit the model. It seems, for now, that *generateDS-1.12a* fails on a certain *annotate* line.

```
[Pete@regitte,2402,cansas1dwg-regitte]$ python ~/generateDS-1.12a/generateDS.py -p CS1D_ -o cansas1d.py -s c
Traceback (most recent call last):
  File "/home/Pete/generateDS-1.12a/generateDS.py", line 3997, in <module>
    main()
  File "/home/Pete/generateDS-1.12a/generateDS.py", line 3993, in main
    processIncludes, superModule=superModule)
  File "/home/Pete/generateDS-1.12a/generateDS.py", line 3909, in parseAndGenerate
    root.annotate()
AttributeError: 'NoneType' object has no attribute 'annotate'
```

Conclusion: not ready yet

Either the canSAS standard (by means of the cansas1d.xsd XML Schema) is not ready or *generateDS.py* does not cover the XML Schema requirements we have at this time. Either way, this is not a viable tool to use now. (2008-05-16)

Other suggestions

- http://www.devx.com/ibm/Article/20261
- http://mail.python.org/pipermail/xml-sig/2002-April/007559.html
- http://pywebsvcs.sourceforge.net/

Retrieved from "http://www.smallangles.net/wgwiki/index.php/cansas1d_binding_Python"

- This page was last modified 23:01, 15 October 2008.