The PyNE Software Library

A Framework for ENSDF?

R. N. Slaybaugh
Univ. of Cal. Berkeley

Nuclear Data Week
6 November 2014
Brookhaven National Laboratory

PyNE can be the ENSDF processing tool

Figure 1 : Python for Nuclear Engineering

Figure 2 : Can be the tool for ENSDF processing

Outline

- PyNE [1]: what is it?
- PyNE as an ENSDF framework
- Other current initiatives
- Get involved!

What is PyNE?

PyNE is the open source nuclear engineering toolkit.

- PyNE is a library of composable tools used to build nuclear science and engineering applications
- It is permissively licensed (2-clause BSD)
- It supports both a c++ and a Python API
- The name 'PyNE' is a bit of a misnomer since most of the code base is in c++ but most daily usage happens in Python
- v0.4 is the current, stable release
- As an organization, PyNE was born in April 2011 (however, core parts of PyNE have existed since 2007)

What can PyNE do?

The idea is to be able to easily combine components and avoid redeveloping utilities someone else has developed.

- Nuclear data and cross-section reading/processing
- Material handling
- Canonical nuclide and reaction naming conventions
- Mesh operations
- MCNP and Serpent input/output parsing
- Fuel cycle functionality (transmutation, enrichment)
- There's more, and the list continues to grow

Nuclear Data in PyNE

PyNE already has some nuclear data support:

- ENSDF level and decay data
  - parser
  - conversion to hdf5
  - access in c++ and Python
- European Activation File cross sections
- Atomic mass data (KAERI)
- ENDF format cross section reader
- ACE format cross section reader

Nuclear Data in PyNE

Structure Dcheata

- 177 471 entries from IAEA ENSDF data
- Spin, parity, energy level
- Half-life, decay type, branching ratio

Decay Data

- Energy, intensity, initial, and final levels for:
  - 116 598 gamma lines
  - 13 230 electron capture/beta +
  - 11 788 betas
  - 2 552 alphas
- 3 868 unique primary decays

Let's look at a quick example!

Expanding into an ENSDF framework

Three iniatives:

1. Add handling of ENSDF reaction data
2. Wrap ENSDF Analysis and(?) Utility programs
   - NSDFLIB
   - ALPHAD, BrIcc, DELTA, GABS, GTOL, HSICC, LOGFT, PANDORA, RadList, RULER
   - ADDGAM, Avetools, Visual Averaging Library, ENSDAT and ComTrans, FMTCK, TREND (or replace through Python)
3. Add comparison utilities (based on community input)

Consistency in ENSDF N and PN records

An issue reported by a PyNE developer and now corrected:

- N record has photon intensity normalization (NR) and branching ratio (BR)
- PN record has entry for NR x BR
- ENSDF manual "recommends" NR x BR
- NR x BR is not always consistent (or meaningful)
- Consistency not checked with current tools
- 18 records contained significant inconsistencies
  - NR x BR = 1 or NR x BR = 0 or mismatch $> 1\%$
  - including ^{241}Pu α-decay

CONSISTENCY IN ENSDF N AND PN RECORDS

Datasets added or modified within the previous month

There are 27 corresponding XUNDL (unevaluated) sets

Revise Search

Matching datasets in ENSDF

Retrieve selected ENSDF datasets:

View in web format | Download selected ENSDF datasets | View in ENSDF format

Nuclide	Dataset	Last Revised	References
	Select All		
^{27}Na	28NE B-N DECAY	2011-08	All references
^{29}Mg	31NA B-2N DECAY	2012-05	All references
^{48}Cr	49FE B+P DECAY	2006-07	All references
^{97}Y	97Y IT DECAY (142 MS)	2010-04	All references
^{105}In	105SN EC DECAY (34 S)	2005-11	All references
^{111}Pd	111PD IT DECAY (5.5 H)	2009-05	All references
^{111}Ag	111PD B- DECAY (5.5 H)	2009-05	All references
^{111}Cd	111AG B- DECAY (64.8 S)	2009-05	All references
^{114}Cd	114IN EC DECAY (49.51 D)	2012-03	All references
^{139}Cs	252CF SF DECAY: ?	2001-04	All references
^{154}Sm	154EU EC DECAY	2009-10	All references
^{213}Ac	217PA A DECAY (1.2 MS)	2007-04	All references
^{237}U	241PU A DECAY	2006-08	All references
^{254}Fm	254ES B- DECAY (39.3 H)	2005-11	All references

Figure 3 : This "changes in the last month" snapshot includes most of the records above

GND and Fudge Support

We are planning to add an interface for Fudge to:

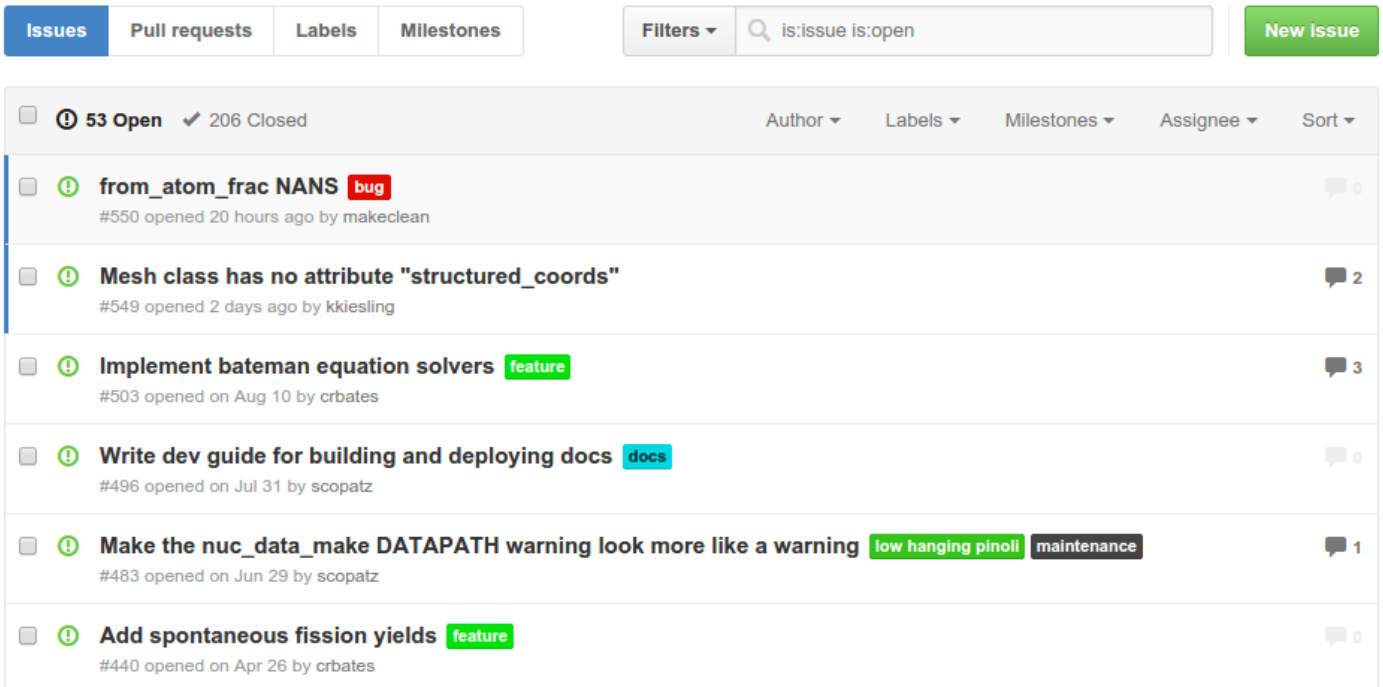
- Create a cross section data source for GND backed by Fudge
- Create an ENDF data source backed by Fudge
- Investigate their processing routines
- Have a GND viewer (if Fudge doesn't)

The Fudge interface should happen after their next release (waiting on licensing issues)

WHAT ELSE IS HAPPENING IN PYNE?

The biggest push: V&V $\rightarrow$ methodically making PyNE compliant with the QA standards we've ratified, which are based on the ASME NQA-1 standards [2]

Many other items (large and small) in our "ticket" list



Verification and Validation

Verification: Have we built the software correctly?
Validation: Have we built the correct software?

Strategies employed by PyNE:

- Version control
- Formal review process
- Documentation: theory manual, user's guide, developer's guide, API, ticket system
- Test suite
- Continuous Integration

Why Would I Get Involved?

As a user:

- You could do your work or research with PyNE
- You get the rest of PyNE's functionality
- You can take advantage of the assurance of the V&V about maintenance!

As a developer:

- You should be selfish
- Contribute to PyNE in ways that support the work that you are doing
- If a feature you want is not in PyNE right now, chances are that other people want to see that feature too
- This will help your future self as much as future other people

How Can I Get Involved?

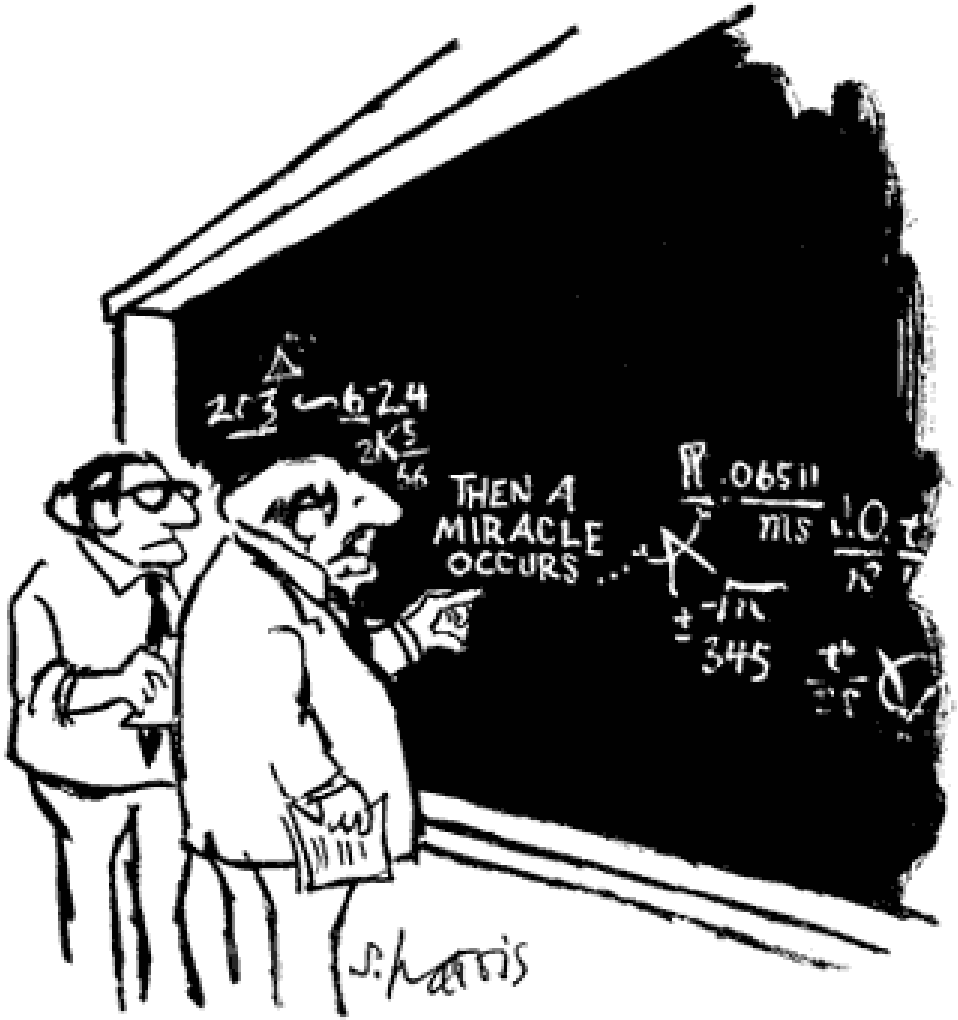
Contact PyNE

- Website: `http://pyne.io/`
- User's Mailing List: `pyne-users@googlegroups.com`
- Developer's List: `pyne-dev@googlegroups.com`
- GitHub: `https://github.com/pyne/pyne`
- Tutorial: `http://pyne.io/tutorial/index.html`

What goes into PyNE?

Anything that is not export controllable, proprietary, or under HIPPA restrictions! (If you have questions, *ask*)

QUESTIONS?



"I think you should be more explicit here in step two."

PyNE In the Literature

- Intro: "PyNE: Python For Nuclear Engineering" [3]
- Progress reports: [4], [5]
- In research: [6], [7], [8]
- V&V: "Quality Assurance within the PyNE Open Source Toolkit" [2]
- Poster at SciPy: [9]

References I


the PyNE Development Team.
PyNE: The Nuclear Engineering Toolkit, 2014.


Elliott Biondo, Anthony Scopatz, Matthew Gidden, Rachel Slaybaugh, and Cameron Bates.
Quality Assurance within the PyNE Open Source Toolkit.
In *Am. Nuc. Soc. Winter Meeting 2014*, volume 111, Anaheim, CA, USA, November 2014.


Anthony Scopatz, Paul K. Romano, Paul P.H. Wilson, and Kathryn D. Huff.
PyNE: Python for Nuclear Engineering.
In *Am. Nuc. Soc. Winter Meeting 2012*, volume 107, San Diego, CA, USA, November 2012.

References II


Anthony Scopatz, Elliott D. Biondo, Carsten Brachem, John Xia, and Paul P. H. Wilson.
PyNE Progress Report.
In *Am. Nuc. Soc. Winter Meeting 2013*, volume 109, Washington, D.C., USA, November 2013.


Cameron Bates, Elliott Biondo, Kathryn Huff, and et al.
PyNE Progress Report.
In *Am. Nuc. Soc. Winter Meeting 2014*, volume 111, Anaheim, CA, USA, November 2014.


E. Biondo, A. Davis, A. Scopatz, and P. P. H. Wilson.
Rigorous Two-Step Activation for Fusion Systems with PyNE.
In *Proc. of the 18th Topical Meeting of the Radiation Protection & Shielding Division of ANS*, Knoxville, TN, 2014.

References III


J.I. Mrquez Damin, J.R. Granada, and D.C. Malaspina.
{CAB} models for water: A new evaluation of the thermal neutron scattering laws for light and heavy water in endf-6 format.
Annals of Nuclear Energy, 65(0):280 – 289, 2014.


Anthony Scopatz.
First & second order approximations to stage numbers in multicomponent enrichment cascades.
In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering (M&C 2013)*, Sun Valley, ID, USA, May 2013.


Anthony Scopatz, Paul Romano, Paul Wilson, Rachel Slaybaugh, Katy Huff, and Eric Relson.
PyNE: Python for Nuclear Engineering.
In *SciPy 2012*, Austin, TX, USA, July 2012.