An EXFOR JSON database and its potential for nuclear data

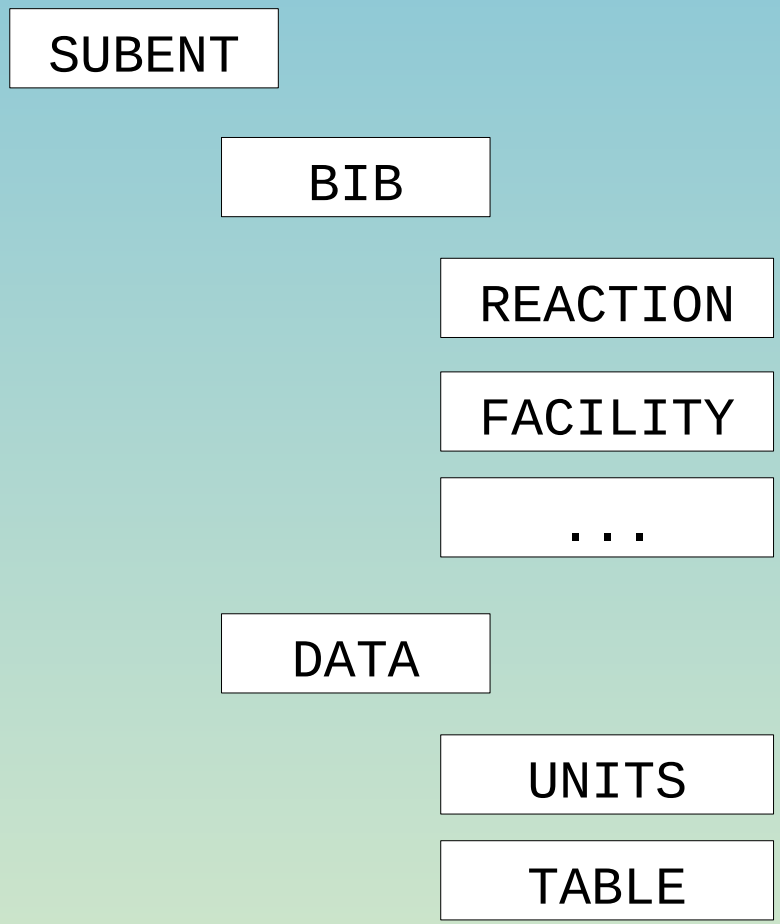
Georg Schnabel

Nuclear Data Section
International Atomic Energy Agency
Department for Nuclear Sciences and Applications

SG 50 – Developing an Automatically Readable, Comprehensive and Curated Experimental Reaction Database – 14.9.2020

Hierarchical EXFOR

```
SUBENT        23171003   20170913   20180129   20180126       2265
BIB                11         26
REACTION   (26-FE-56(N,2N)26-FE-55,,SIG)
DECAY-DATA (26-FE-55,2.73YR)
FACILITY  1(VDG,2ZZZGEL) Van de Graaff accelerator at IRMM.
          2(ACCEL,2AUSIRK) For AMS (accelerator mass spectrometry)
            at VERA lab. for radionuclides 10Be, 14C, 26Al, 55Fe,
           (CCW,2GERDRE)
METHOD    1(ACTIV)  Activation technique combined with
          2(AMS)    mass spectrometric technique
SAMPLE      Natural iron samples were irradiated at TU Dresden
            and IRMM.
INC-SOURCE1(D-T) T(d,n)He-4 .
INC-SPECT 1 Quasi-monoenergetic neutrons with energies between
            13.4 and 14.8 MeV;
          2 from 13 to 20 MeV.
FLAG       (1.) Experiment in 2007 yr,   TUD/VERA
           (2.) Experiment in 2010 yr,   TUD/VERA
           (3.) Experiment in 2010 yr,   IRMM/VERA
ERR-ANALYS (EN-ERR) finite neutron energy distribution and the
                   absolute uncertainty in the neutron energy
                      (namely it is EN-ERR+EN-RSL).
STATUS     (PRELM) Preliminary results (decleared by A.Wallner,
                   2017-08-30) under STATUS).
           (TABLE) Data received from the author (2017-08-30)
HISTORY    (20170913R) SD: Data received from the author were
           added. FLAG was added. BIB update according to
           comments from author.
ENDBIB              26
NOCOMMON             0          0
DATA                 5         15
EN         EN-ERR     DATA       DATA-ERR   FLAG
MEV        MEV         MB         MB         NO-DIM
      13.35       0.15       300.       100.         3.
      13.49       0.04      322.4       16.1         1.
      ...
ENDDATA             17
ENDSUBENT           48
```



Hierarchical format

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Philosophy of EXFOR parser

- Keep it simple – keep it EXFOR, only introduce some commas, double-colons, and curly brackets to make it JSON
- Keep field names, keep field content, keep pointers (i.e., DATA … 1 → DATA#1)
- More complex transformations can be done in Python et al.

A computational EXFOR database, arXiv:1908.00209

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

EXFOR as JSON


```
{
   "ID": "23171003",

   "DATEMOD": "20170913",

   "BIB":{

      "REACTION": "(26-FE-56(N,2N)26-FE-55,,SIG)",

      "DECAY-DATA": "(26-FE-55,2.73YR)",
                                              "

      "FACILITY": [

         "(VDG,2ZZZGEL) Van de Graaff accelerator ...",

         "(ACCEL,2AUSIRK) For AMS ... at VERA lab ..."
      ],

      "ERR-ANALYS": "(EN-ERR) finite neutron energy
                     distribution and the absolute
                     uncertainty in the neutron energy
                     (namely it is EN-ERR+EN-RSL).",

      ...
   },

   ...
}
```


R Code for creation of JSON

```
library(exforParser)

filepath = "<PATH TO EXFOR FILE>"

exforText = readChar(filepath,
file.info(filepath)$size)

entry = parseEntry(exforText)
convToJSON(entry$SUBENT[[3]])
```

https://github.com/gschnabel/exforParser

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Convenient access & modifications

From R

```
...
entry = parseEntry(exforText)

entry$SUBENT[[3]]$BIB$REACTION

entry$SUBENT[[3]]$DATA$UNIT

entry$SUBENT[[3]]$DATA$TABLE
```

From Python

```
import json

with open('exforEntryFile') as json_file:
    entry = json.load(json_file)

entry['SUBENT'][3]['BIB']['REACTION']

...
```

From command line

```
jq ".SUBENT[3].BIB.REACTION" exforEntryFile.json

jq ".SUBENT[3].DATA.TABLE.EN" exforEntryFile.json
```


Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Towards a computational format

… pertinent transformations of this "raw" JSON format can be done in Python et al. For example, ~150 lines of R code achieve:

- Standardization of units (MeV and mBarn)
- Merging of first subentry to subsequent ones
- Merging of common blocks into data tables

function transformSubent in
https://github.com/gschnabel/exforParser

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Use case UQ and ML

```
# get entry
subent = exfor['30651002']

# extract description text to ERR-x components
err_str = subent['BIB']['ERR-ANALYS'].replace('\n','')
err_str = re.sub(' +', ' ', err_str)
pat = r'(\(ERR-.\))([^(]+)'
unc_tuples = [x for x in re.findall(pat, err_str)]
df = pd.DataFrame(unc_tuples)

# access to a values of ERR-x components
subent['DATA']['TABLE']['ERR-1']
```

https://github.com/IAEA-NDS/exfor-couchdb-docker

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Use case UQ and ML

	comp_tag	description
0	(ERR-T)	Following errors are combined in quadrature:
1	(ERR-1)	The error in estimating the peak area - 5%
2	(ERR-S)	The statistical error in the number of Li-6
3	(ERR-2)	The error in estimating the attenuation ofthe neutron beam - 20%
4	(ERR-3)	Uncertainty in the integrated number of He-3 events - 2%
5	(ERR-4)	The error in counting the number of He-3 events - 0.5%
6	(ERR-5)	The scintillator thickness quoted by the manufacturer - 0.5%
7	(ERR-6)	Uncertainty in the multiple scattering - 1%

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Document-oriented databases

- Organized storage of JSON files
- Powerful search capabilities to locate and extract data

https://www.mongodb.com

https://couchdb.apache.org/

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org

Conclusions

- EXFOR to JSON conversion without changing logical structure
- Easier to derive computational formats in a high-level language, e.g., Python and R
- EXFOR JSON database available using on MongoDB and CouchDB on GitHub (license-wise CouchDB to be preferred)

Georg Schnabel, IAEA Nuclear Data Section
e-mail: g.schnabel@iaea.org
Web: https://www-nds.iaea.org