IAEA

60 Years

Atoms for Peace and Development

Lowering the ENDF-6 entrance barrier for evaluators

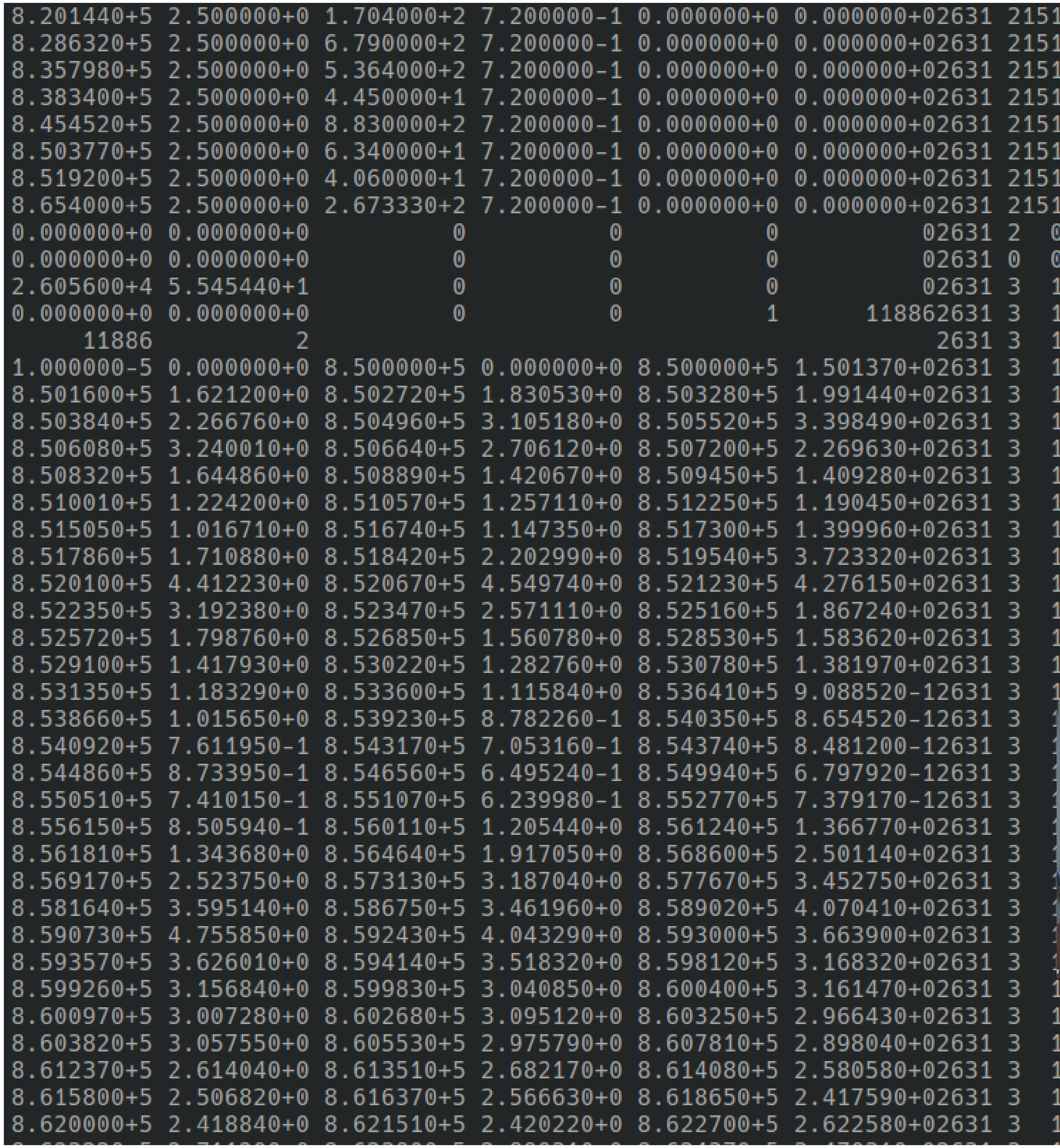
Georg Schnabel
g.schnabel [at] iaea.org

Nuclear Data Section
Division of Physical and Chemical Sciences NAPC
Department for Nuclear Sciences and Applications
IAEA, Vienna

INDEN CM on Structural Materials

18 December 2022

Acknowledgment



```
8.201440+5 2.500000+0 1.704000+2 7.200000-1 0.000000+0 0.000000+02631 2151   320
8.286320+5 2.500000+0 6.790000+2 7.200000-1 0.000000+0 0.000000+02631 2151   321
8.357980+5 2.500000+0 5.364000+2 7.200000-1 0.000000+0 0.000000+02631 2151   322
8.383400+5 2.500000+0 4.450000+1 7.200000-1 0.000000+0 0.000000+02631 2151   323
8.454520+5 2.500000+0 8.830000+2 7.200000-1 0.000000+0 0.000000+02631 2151   324
8.503770+5 2.500000+0 6.340000+1 7.200000-1 0.000000+0 0.000000+02631 2151   325
8.519200+5 2.500000+0 4.060000+1 7.200000-1 0.000000+0 0.000000+02631 2151   326
8.654000+5 2.500000+0 2.673330+2 7.200000-1 0.000000+0 0.000000+02631 2151   327
0.000000+0 0.000000+0          0          0          0          02631 2  099999
0.000000+0 0.000000+0          0          0          0          02631 0  0     0
2.605600+4 5.545440+1          0          0          0          02631 3  1     1
0.000000+0 0.000000+0          0          0          1      118862631 3  1     2
     11886          2                                         2631 3  1     3
1.000000-5 0.000000+0 8.500000+5 0.000000+0 8.500000+5 1.501370+02631 3  1     4
8.501600+5 1.621200+0 8.502720+5 1.830530+0 8.503280+5 1.991440+02631 3  1     5
8.503840+5 2.266760+0 8.504960+5 3.105180+0 8.505520+5 3.398490+02631 3  1     6
8.506080+5 3.240010+0 8.506640+5 2.706120+0 8.507200+5 2.269630+02631 3  1     7
8.508320+5 1.644860+0 8.508890+5 1.420670+0 8.509450+5 1.409280+02631 3  1     8
8.510010+5 1.224200+0 8.510570+5 1.257110+0 8.512250+5 1.190450+02631 3  1     9
8.515050+5 1.016710+0 8.516740+5 1.147350+0 8.517300+5 1.399960+02631 3  1    10
8.517860+5 1.710880+0 8.518420+5 2.202990+0 8.519540+5 3.723320+02631 3  1    11
8.520100+5 4.412230+0 8.520670+5 4.549740+0 8.521230+5 4.276150+02631 3  1    12
8.522350+5 3.192380+0 8.523470+5 2.571110+0 8.525160+5 1.867240+02631 3  1    13
8.525720+5 1.798760+0 8.526850+5 1.560780+0 8.528530+5 1.583620+02631 3  1    14
8.529100+5 1.417930+0 8.530220+5 1.282760+0 8.530780+5 1.381970+02631 3  1    15
8.531350+5 1.183290+0 8.533600+5 1.115840+0 8.536410+5 9.088520-12631 3  1    16
8.538660+5 1.015650+0 8.539230+5 8.782260-1 8.540350+5 8.654520-12631 3
8.540920+5 7.611950-1 8.543170+5 7.053160-1 8.543740+5 8.481200-12631 3
8.544860+5 8.733950-1 8.546560+5 6.495240-1 8.549940+5 6.797920-12631 3
8.550510+5 7.410150-1 8.551070+5 6.239980-1 8.552770+5 7.379170-12631 3
8.556150+5 8.505940-1 8.560110+5 1.205440+0 8.561240+5 1.366770+02631 3
8.561810+5 1.343680+0 8.564640+5 1.917050+0 8.568600+5 2.501140+02631 3
8.569170+5 2.523750+0 8.573130+5 3.187040+0 8.577670+5 3.452750+02631 3
8.581640+5 3.595140+0 8.586750+5 3.461960+0 8.589020+5 4.070410+02631 3
8.590730+5 4.755850+0 8.592430+5 4.043290+0 8.593000+5 3.663900+02631 3
8.593570+5 3.626010+0 8.594140+5 3.518320+0 8.598120+5 3.168320+02631 3
8.599260+5 3.156840+0 8.599830+5 3.040850+0 8.600400+5 3.161470+02631 3  1    27
8.600970+5 3.007280+0 8.602680+5 3.095120+0 8.603250+5 2.966430+02631 3  1    28
8.603820+5 3.057550+0 8.605530+5 2.975790+0 8.607810+5 2.898040+02631 3  1    29
8.612370+5 2.614040+0 8.613510+5 2.682170+0 8.614080+5 2.580580+02631 3  1    30
8.615800+5 2.506820+0 8.616370+5 2.566630+0 8.618650+5 2.417590+02631 3  1    31
8.620000+5 2.418840+0 8.621510+5 2.420220+0 8.622700+5 2.622580+02631 3  1    32
```

Daniel Lopez Aldama
(IAEA Consultant)

https://github.com/iaea-nds/acemaker

How to put nuclear data into an ENDF-6 formatted file?

Paths to ENDF-6 file creation

- Model code systems (e.g. Talys and Empire)
- Statistical evaluation systems (e.g. GANDR)
- Packages to interact with ENDF-6 formatted files

Popular and recent packages

Reading & Writing

- SANDY
- DeCE
- ENDFtk?

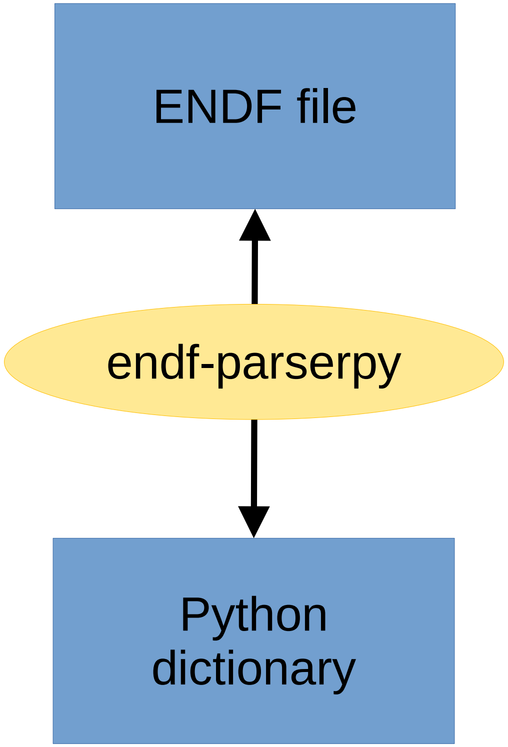
Only reading

- Endf-parser
- PyNE

Two (+one) requirements

- Support reading and writing of entire ENDF-6 format
- Convenient use from a data science language (e.g. Python)
- *Provably correct*

Basic design

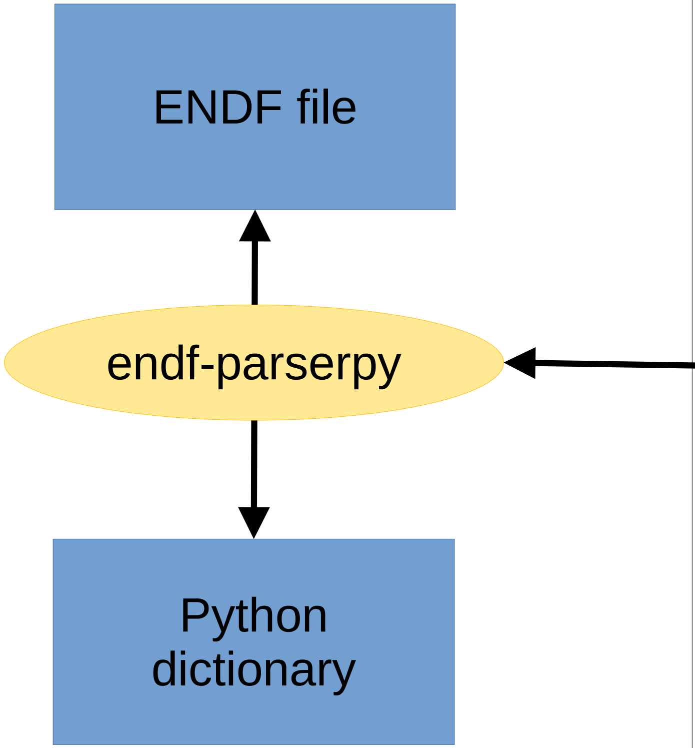


ENDF file

endf-parserpy

Python dictionary

Basic design

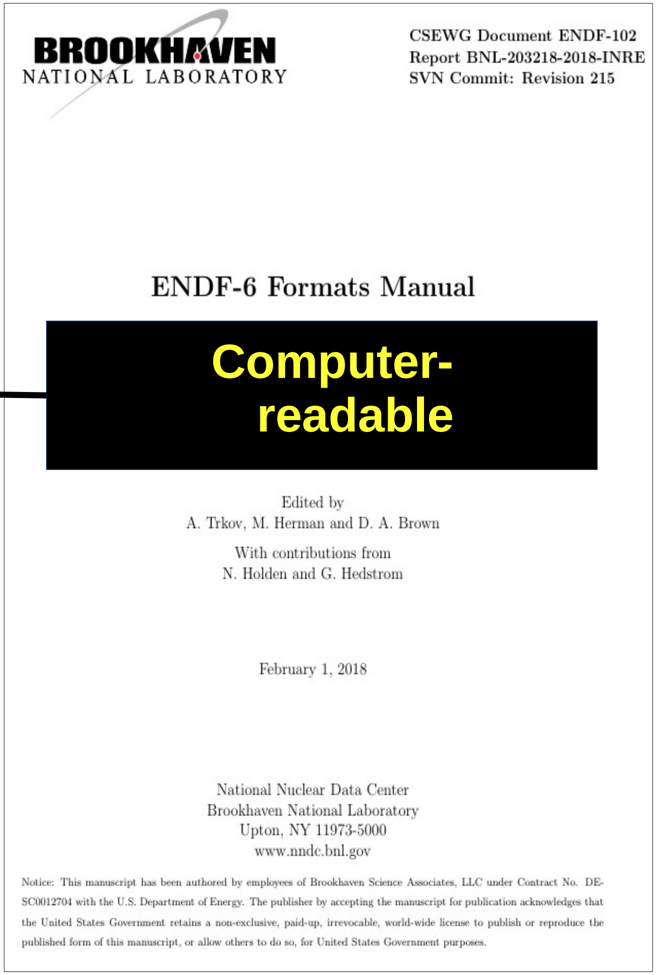


ENDF file

endf-parserpy

Python dictionary

Computer-readable



BROOKHAVEN
NATIONAL LABORATORY

CSEWG Document ENDF-102
Report BNL-203218-2018-INRE
SVN Commit: Revision 215

ENDF-6 Formats Manual

Edited by
A. Trkov, M. Herman and D. A. Brown

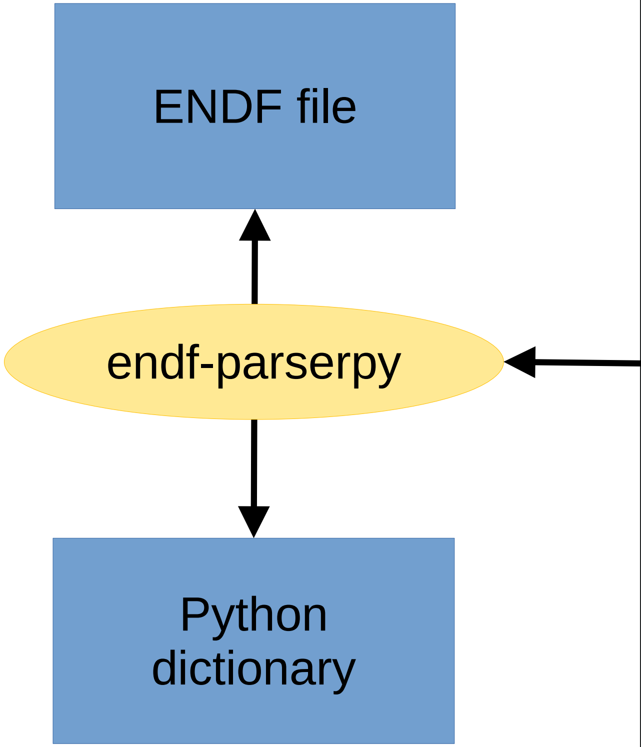
With contributions from
N. Holden and G. Hedstrom

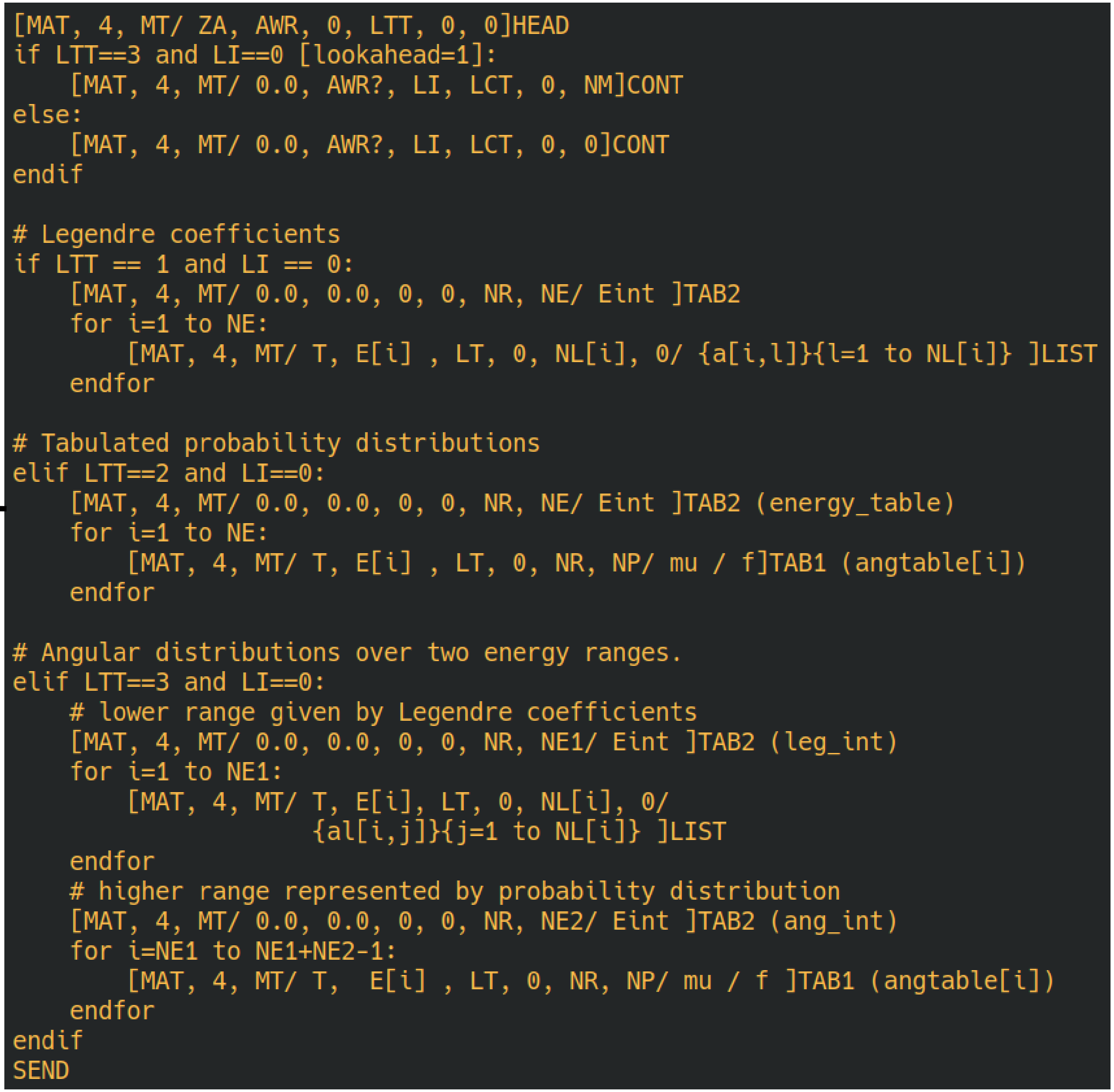
February 1, 2018

National Nuclear Data Center
Brookhaven National Laboratory
Upton, NY 11973-5000
www.nndc.bnl.gov

Notice: This manuscript has been authored by employees of Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy. The publisher by accepting the manuscript for publication acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

Formal ENDF format description



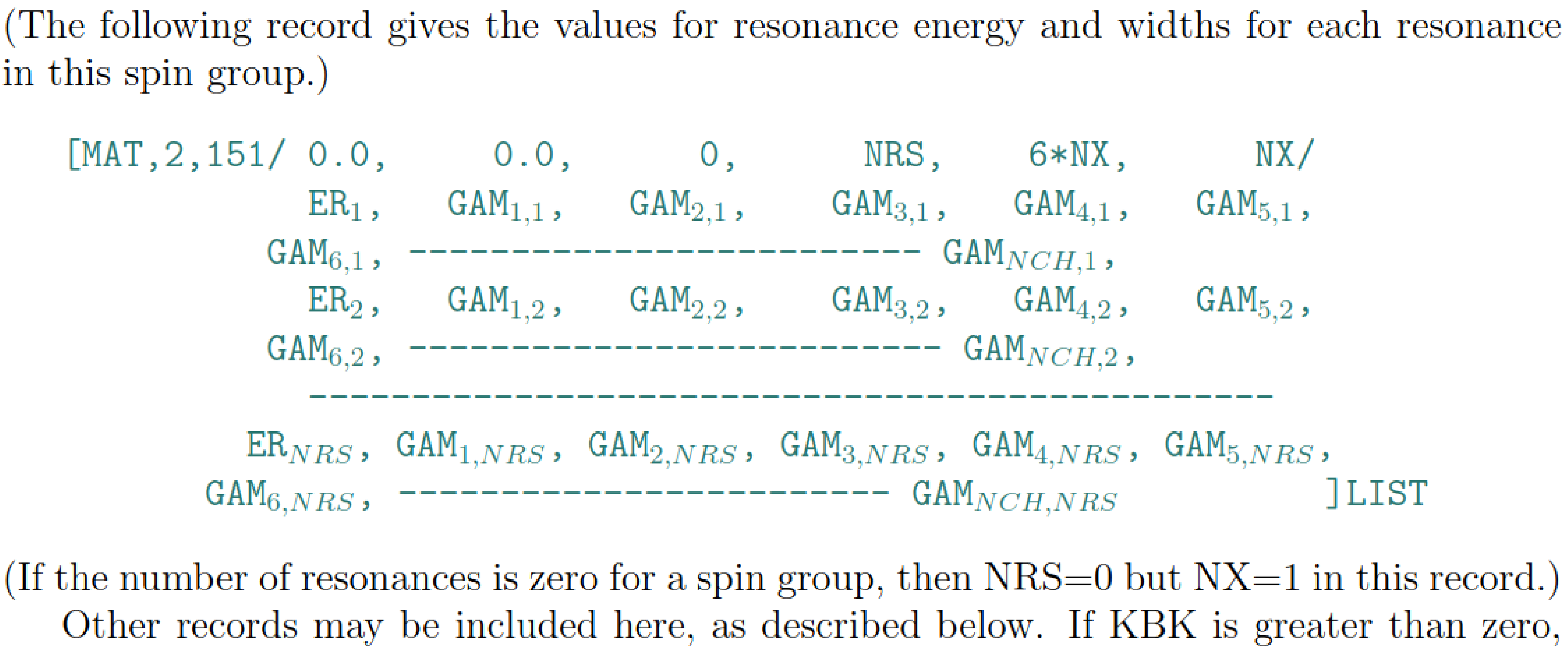


```
[MAT, 4, MT/ ZA, AWR, 0, LTT, 0, 0]HEAD
if LTT==3 and LI==0 [lookahead=1]:
    [MAT, 4, MT/ 0.0, AWR?, LI, LCT, 0, NM]CONT
else:
    [MAT, 4, MT/ 0.0, AWR?, LI, LCT, 0, 0]CONT
endif

# Legendre coefficients
if LTT == 1 and LI == 0:
    [MAT, 4, MT/ 0.0, 0.0, 0, 0, NR, NE/ Eint ]TAB2
    for i=1 to NE:
        [MAT, 4, MT/ T, E[i] , LT, 0, NL[i], 0/ {a[i,l]}{l=1 to NL[i]} ]LIST
    endfor

# Tabulated probability distributions
elif LTT==2 and LI==0:
    [MAT, 4, MT/ 0.0, 0.0, 0, 0, NR, NE/ Eint ]TAB2 (energy_table)
    for i=1 to NE:
        [MAT, 4, MT/ T, E[i] , LT, 0, NR, NP/ mu / f]TAB1 (angtable[i])
    endfor

# Angular distributions over two energy ranges.
elif LTT==3 and LI==0:
    # lower range given by Legendre coefficients
    [MAT, 4, MT/ 0.0, 0.0, 0, 0, NR, NE1/ Eint ]TAB2 (leg_int)
    for i=1 to NE1:
        [MAT, 4, MT/ T, E[i], LT, 0, NL[i], 0/
                     {al[i,j]}{j=1 to NL[i]} ]LIST
    endfor
    # higher range represented by probability distribution
    [MAT, 4, MT/ 0.0, 0.0, 0, 0, NR, NE2/ Eint ]TAB2 (ang_int)
    for i=NE1 to NE1+NE2-1:
        [MAT, 4, MT/ T,  E[i] , LT, 0, NR, NP/ mu / f ]TAB1 (angtable[i])
    endfor
endif
SEND
```

“How to explain ENDF-6 to computers: A formal ENDF description language”, arXiv:2312.08249

Provably correct

- The ENDF-6 format manual is incomplete/misleading* for LRF=7
- Result: some divergence between packages



(The following record gives the values for resonance energy and widths for each resonance in this spin group.)

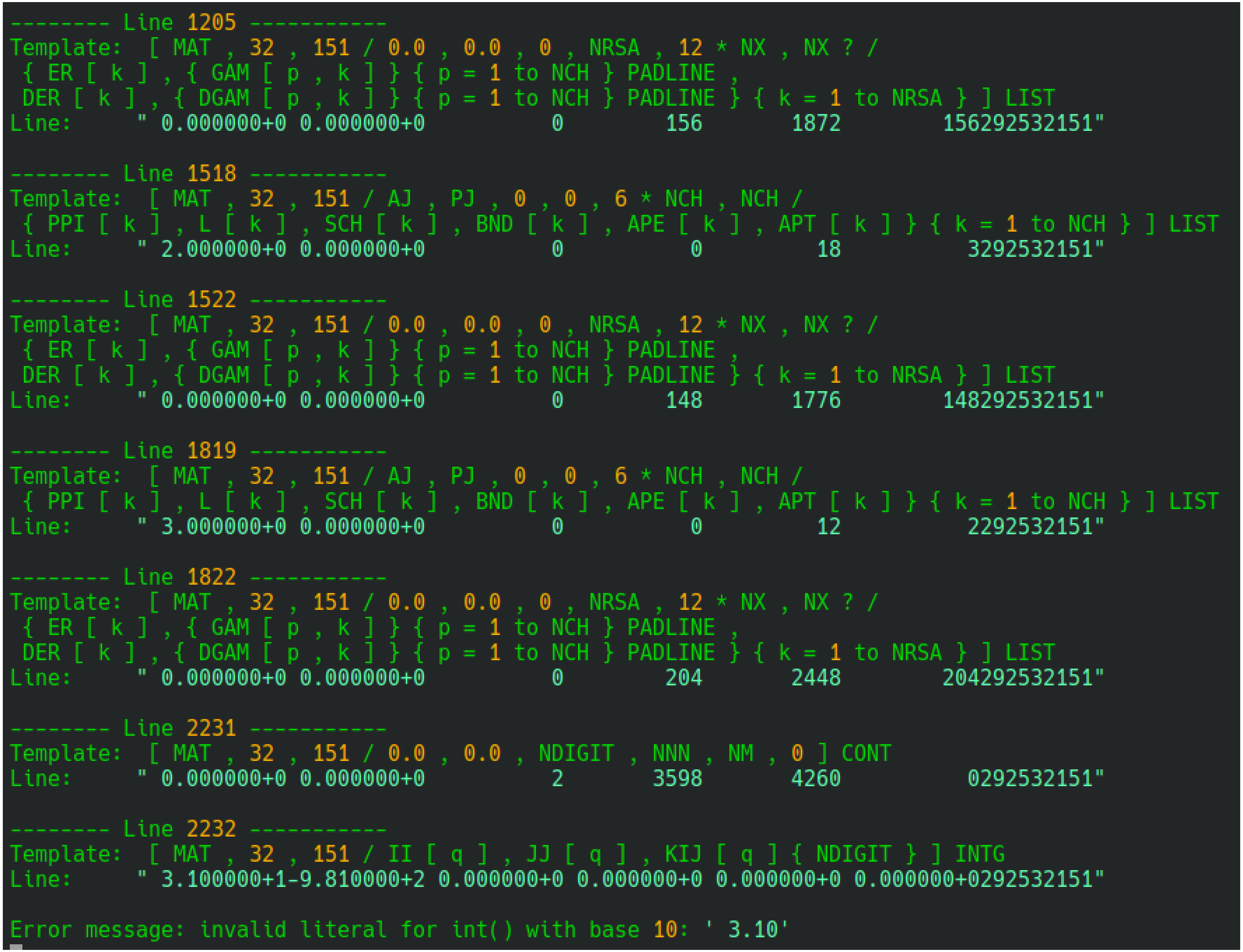
[MAT,2,151/ 0.0, 0.0, 0, NRS, 6*NX, NX/
ER_1, $\mathrm{GAM}_{1,1}$, $\mathrm{GAM}_{2,1}$, $\mathrm{GAM}_{3,1}$, $\mathrm{GAM}_{4,1}$, $\mathrm{GAM}_{5,1}$,
$\mathrm{GAM}_{6,1}$, -------------------------- $\mathrm{GAM}_{NCH,1}$,
ER_2, $\mathrm{GAM}_{1,2}$, $\mathrm{GAM}_{2,2}$, $\mathrm{GAM}_{3,2}$, $\mathrm{GAM}_{4,2}$, $\mathrm{GAM}_{5,2}$,
$\mathrm{GAM}_{6,2}$, --------------------------- $\mathrm{GAM}_{NCH,2}$,
----
ER_{NRS}, $\mathrm{GAM}_{1,NRS}$, $\mathrm{GAM}_{2,NRS}$, $\mathrm{GAM}_{3,NRS}$, $\mathrm{GAM}_{4,NRS}$, $\mathrm{GAM}_{5,NRS}$,
$\mathrm{GAM}_{6,NRS}$, ------------------------- $\mathrm{GAM}_{NCH,NRS}$]LIST

(If the number of resonances is zero for a spin group, then NRS=0 but NX=1 in this record.)

Other records may be included here, as described below. If KBK is greater than zero,

* or at the very least inconsistent with ENDF/B-VIII.0

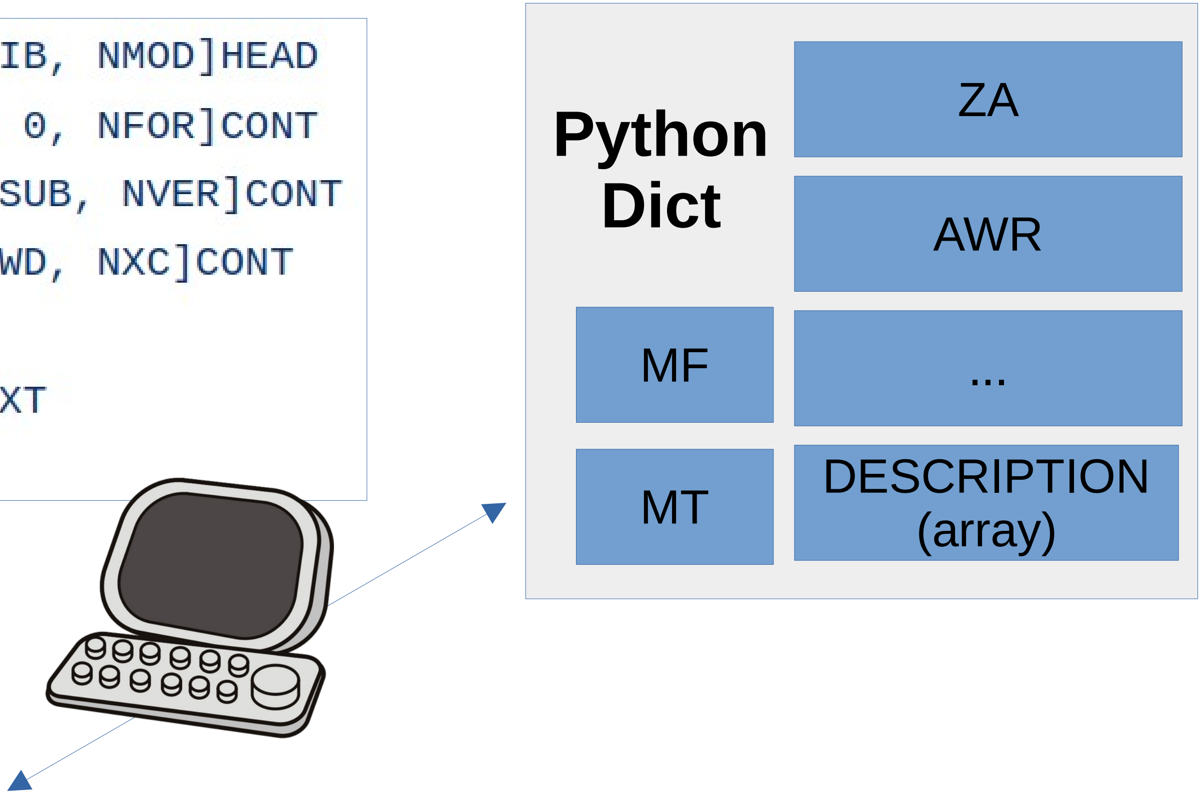
Example of format checking: Cu-63

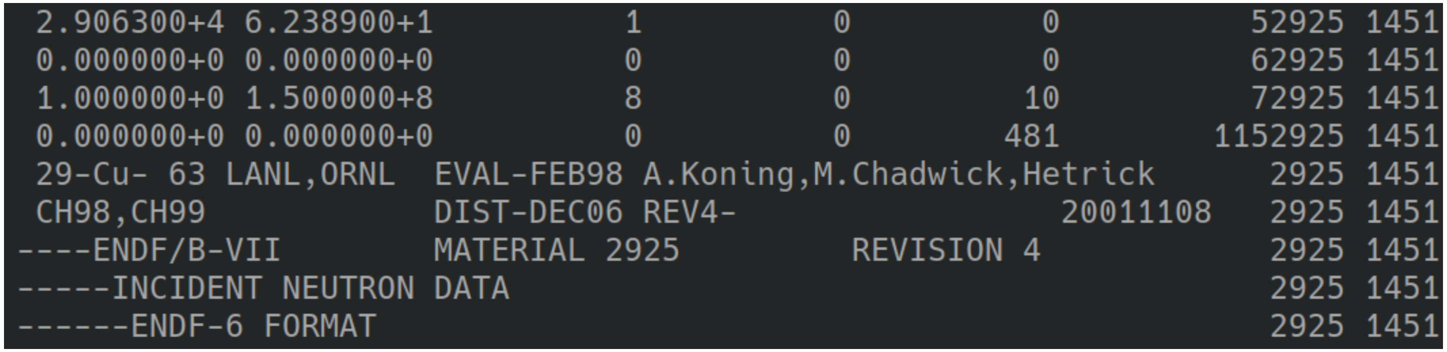


```
-------- Line 1205 -----------
Template:  [ MAT , 32 , 151 / 0.0 , 0.0 , 0 , NRSA , 12 * NX , NX ? /
 { ER [ k ] , { GAM [ p , k ] } { p = 1 to NCH } PADLINE ,
 DER [ k ] , { DGAM [ p , k ] } { p = 1 to NCH } PADLINE } { k = 1 to NRSA } ] LIST
Line:     " 0.000000+0 0.000000+0          0        156       1872        156292532151"

-------- Line 1518 -----------
Template:  [ MAT , 32 , 151 / AJ , PJ , 0 , 0 , 6 * NCH , NCH /
 { PPI [ k ] , L [ k ] , SCH [ k ] , BND [ k ] , APE [ k ] , APT [ k ] } { k = 1 to NCH } ] LIST
Line:     " 2.000000+0 0.000000+0          0          0         18          3292532151"

-------- Line 1522 -----------
Template:  [ MAT , 32 , 151 / 0.0 , 0.0 , 0 , NRSA , 12 * NX , NX ? /
 { ER [ k ] , { GAM [ p , k ] } { p = 1 to NCH } PADLINE ,
 DER [ k ] , { DGAM [ p , k ] } { p = 1 to NCH } PADLINE } { k = 1 to NRSA } ] LIST
Line:     " 0.000000+0 0.000000+0          0        148       1776        148292532151"

-------- Line 1819 -----------
Template:  [ MAT , 32 , 151 / AJ , PJ , 0 , 0 , 6 * NCH , NCH /
 { PPI [ k ] , L [ k ] , SCH [ k ] , BND [ k ] , APE [ k ] , APT [ k ] } { k = 1 to NCH } ] LIST
Line:     " 3.000000+0 0.000000+0          0          0         12          2292532151"

-------- Line 1822 -----------
Template:  [ MAT , 32 , 151 / 0.0 , 0.0 , 0 , NRSA , 12 * NX , NX ? /
 { ER [ k ] , { GAM [ p , k ] } { p = 1 to NCH } PADLINE ,
 DER [ k ] , { DGAM [ p , k ] } { p = 1 to NCH } PADLINE } { k = 1 to NRSA } ] LIST
Line:     " 0.000000+0 0.000000+0          0        204       2448        204292532151"

-------- Line 2231 -----------
Template:  [ MAT , 32 , 151 / 0.0 , 0.0 , NDIGIT , NNN , NM , 0 ] CONT
Line:     " 0.000000+0 0.000000+0          2       3598       4260          0292532151"

-------- Line 2232 -----------
Template:  [ MAT , 32 , 151 / II [ q ] , JJ [ q ] , KIJ [ q ] { NDIGIT } ] INTG
Line:     " 3.100000+1-9.810000+2 0.000000+0 0.000000+0 0.000000+0 0.000000+0292532151"

Error message: invalid literal for int() with base 10: ' 3.10'
```

Basic translation

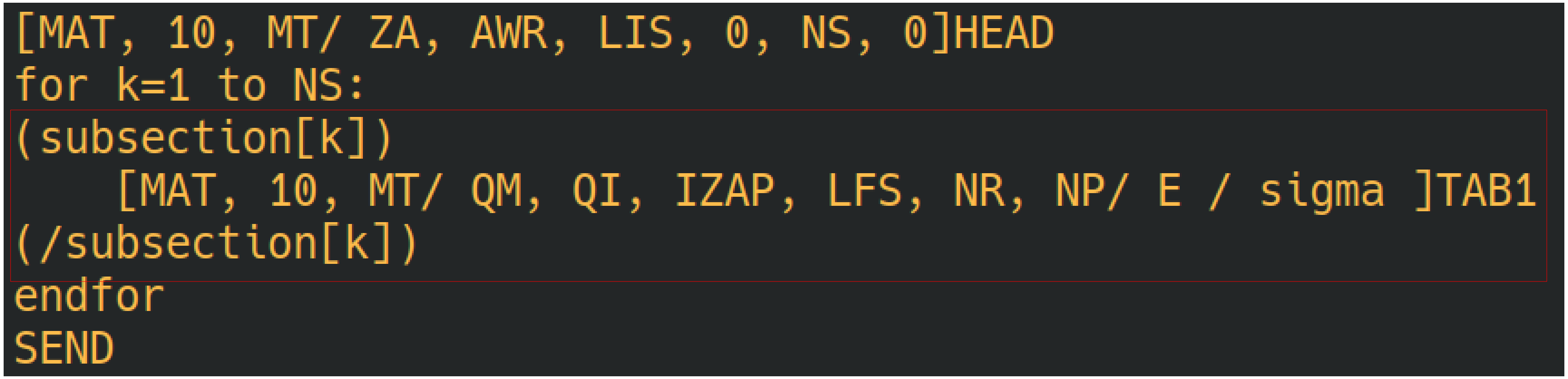
```
[MAT, 1,451/ ZA, AWR, LRP, LFI, NLIB, NMOD]HEAD
[MAT, 1,451/ ELIS, STA, LIS, LISO, 0, NFOR]CONT
[MAT, 1,451/ AWI, EMAX, LREL, 0, NSUB, NVER]CONT
[MAT, 1,451/ TEMP, 0.0, LDRV, 0, NWD, NXC]CONT
for i=1 to NWD:
    [MAT, 1,451/ DESCRIPTION[i]]TEXT
endfor
```



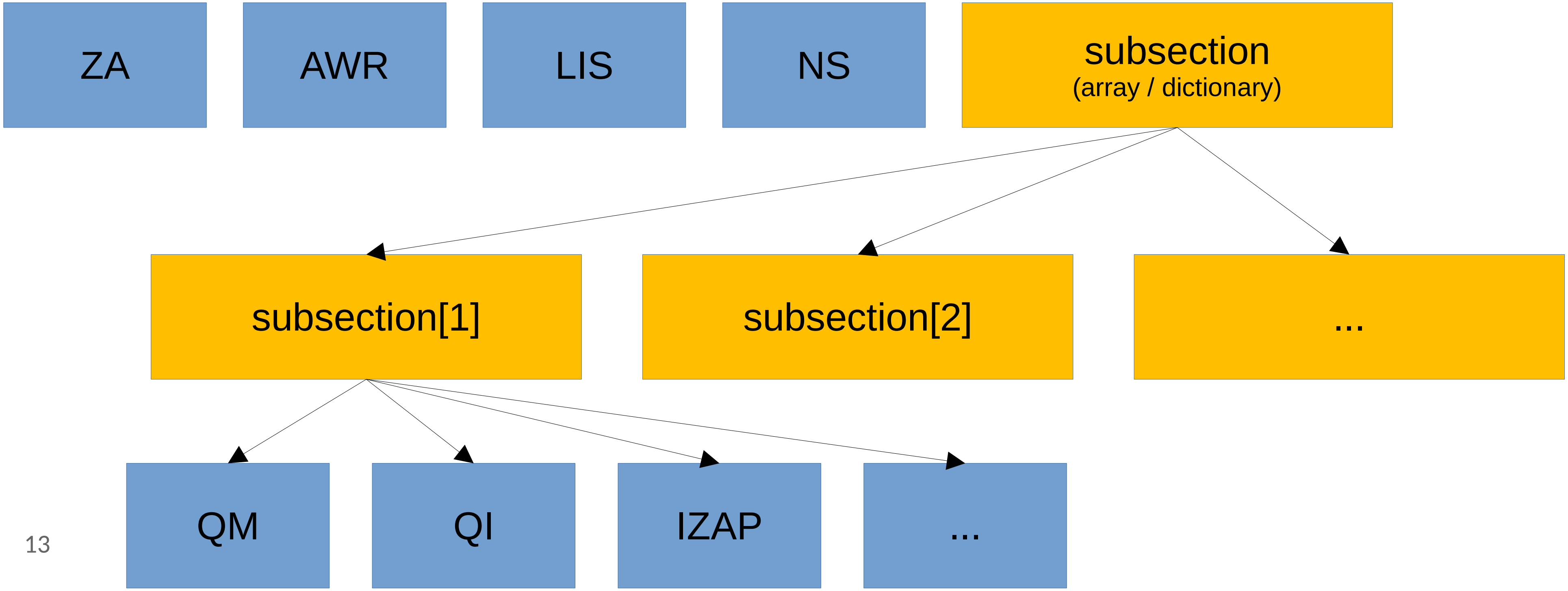


```
 2.906300+4 6.238900+1          1          0          0          52925 1451
 0.000000+0 0.000000+0          0          0          0          62925 1451
 1.000000+0 1.500000+8          8          0         10          72925 1451
 0.000000+0 0.000000+0          0          0        481       1152925 1451
 29-Cu- 63 LANL,ORNL  EVAL-FEB98 A.Koning,M.Chadwick,Hetrick       2925 1451
 CH98,CH99            DIST-DEC06 REV4-            20011108        2925 1451
----ENDF/B-VII        MATERIAL 2925          REVISION 4           2925 1451
-----INCIDENT NEUTRON DATA                                        2925 1451
------ENDF-6 FORMAT                                               2925 1451
```

Allowing for nesting: Sections



```
[MAT, 10, MT/ ZA, AWR, LIS, 0, NS, 0]HEAD
for k=1 to NS:
(subsection[k])
    [MAT, 10, MT/ QM, QI, IZAP, LFS, NR, NP/ E / sigma ]TAB1
(/subsection[k])
endfor
SEND
```



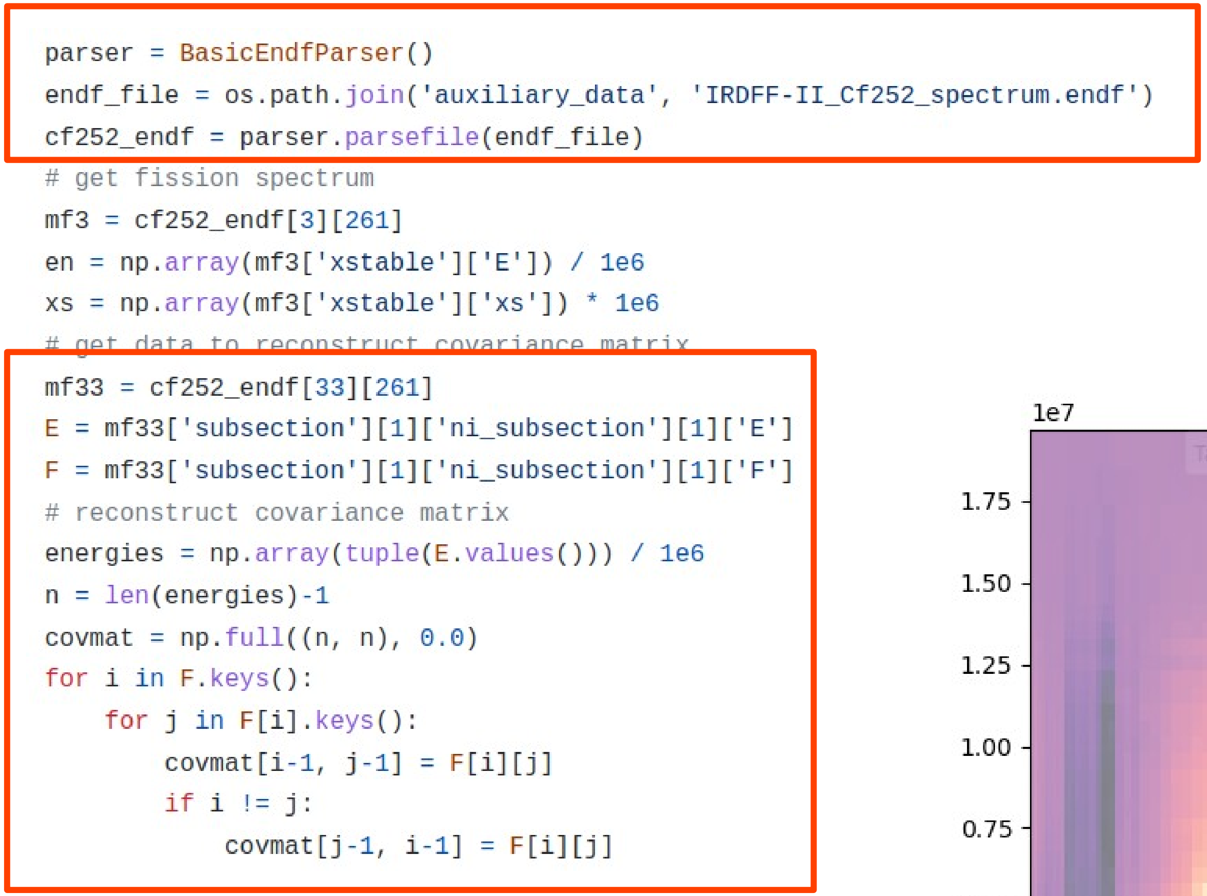
Example of use: Changing (n,tot) cross section

```
from endf_parserpy import ExtEndfParser

parser = ExtEndfParser()
endf_dict = parser.parsefile("input.endf")




parser.writefile("output.endf", endf_dict)
```

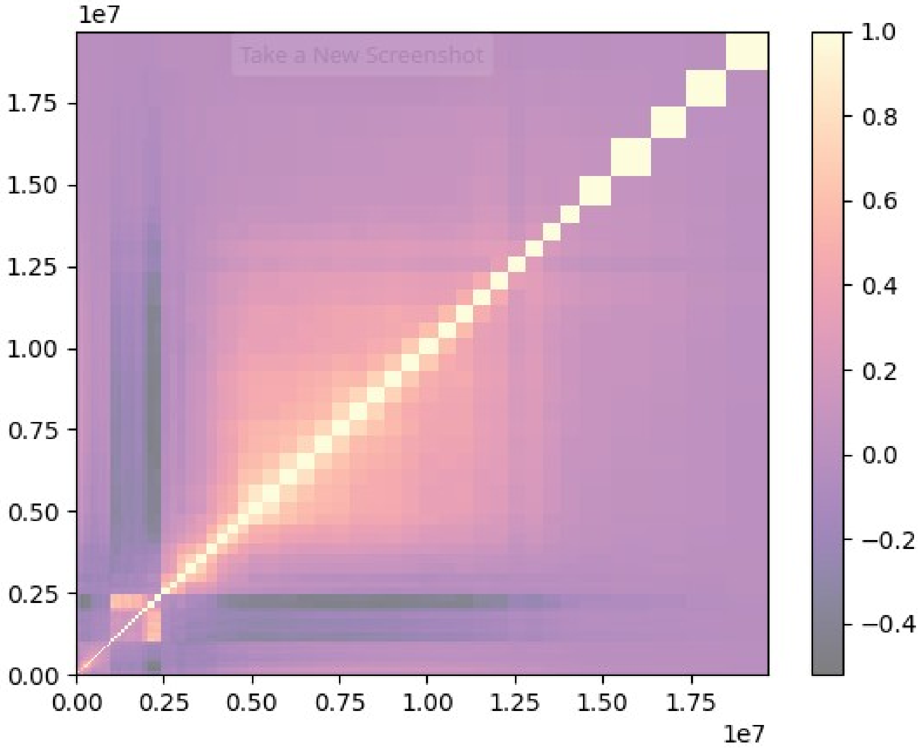
Example of use:
Changing (n,tot) cross section

```
from endf_parserpy import ExtEndfParser

parser = ExtEndfParser()
endf_dict = parser.parsefile("input.endf")

updated_energies = np.linspace(1e6, 1e8, 100)

endf_dict[3][1]['xstable']['E'] = updated_energies
endf_dict[3][1]['xstable']['xs'] = np.sin(updated_energies) + 2

endf_dict[3][1]['xstable']['NBT'] = [len(updated_energies)]
endf_dict[3][1]['xstable']['INT'] = [2]

parser.writefile("output.endf”, endf_dict)
```

Covariance retrieval example



```
parser = BasicEndfParser()
endf_file = os.path.join('auxiliary_data', 'IRDFF-II_Cf252_spectrum.endf')
cf252_endf = parser.parsefile(endf_file)
# get fission spectrum
mf3 = cf252_endf[3][261]
en = np.array(mf3['xstable']['E']) / 1e6
xs = np.array(mf3['xstable']['xs']) * 1e6
# get data to reconstruct covariance matrix
mf33 = cf252_endf[33][261]
E = mf33['subsection'][1]['ni_subsection'][1]['E']
F = mf33['subsection'][1]['ni_subsection'][1]['F']
# reconstruct covariance matrix
energies = np.array(tuple(E.values())) / 1e6
n = len(energies)-1
covmat = np.full((n, n), 0.0)
for i in F.keys():
    for j in F[i].keys():
        covmat[i-1, j-1] = F[i][j]
        if i != j:
            covmat[j-1, i-1] = F[i][j]
```



https://github.com/IAEA-NDS/neutron-standards-database/blob/main/codes/database_modifications/use_irdff_cf252_spectrum.py

Translation between representations

Translation between representations

endf-parserpy

ENDF-6

Python dictionary

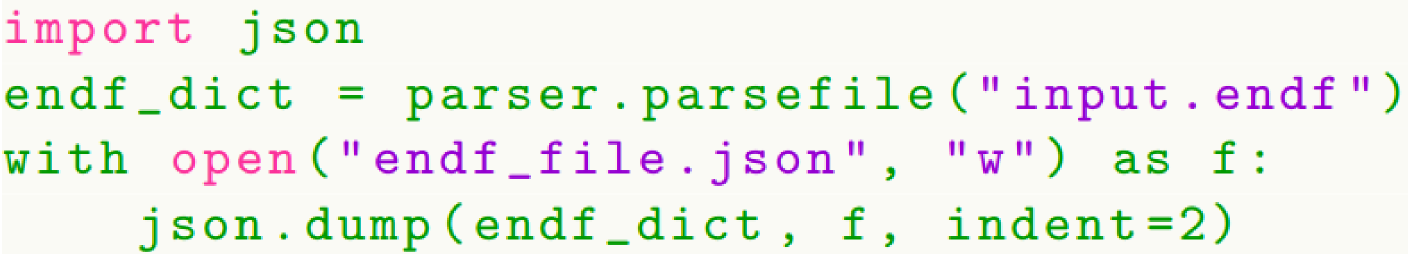
Translation between representations

FUDGE

ENDF-6

GNDS

Translation between representations

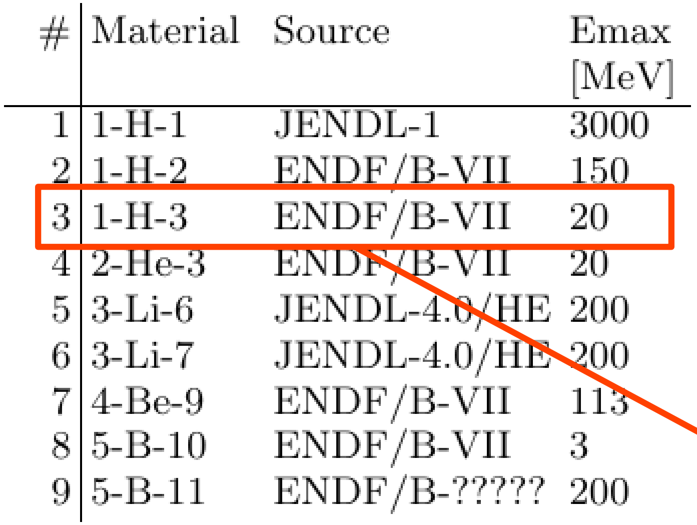


```
import json
endf_dict = parser.parsefile("input.endf")
with open("endf_file.json", "w") as f:
    json.dump(endf_dict, f, indent=2)
```

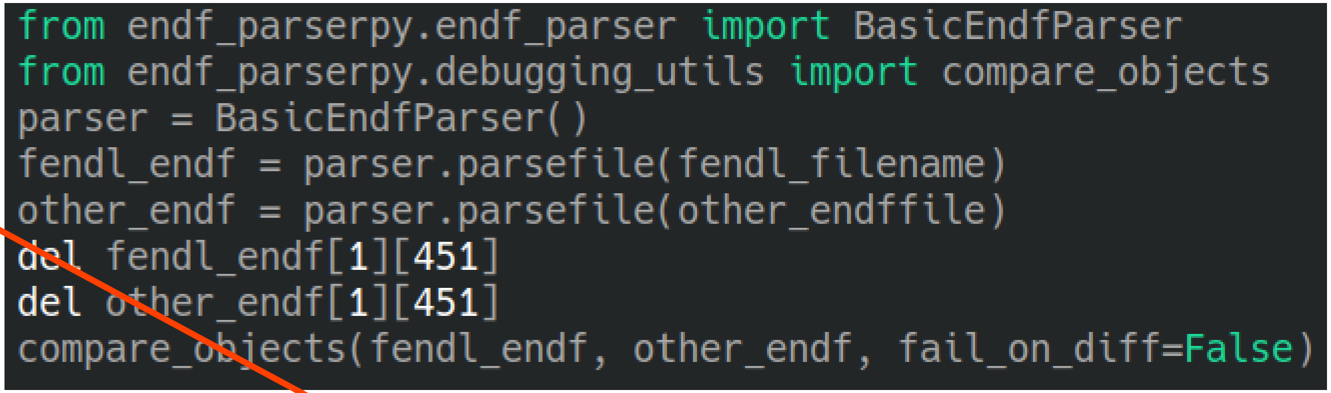
Python dictionary

JSON

Advanced diff functionality (Provenance checking in FENDL)

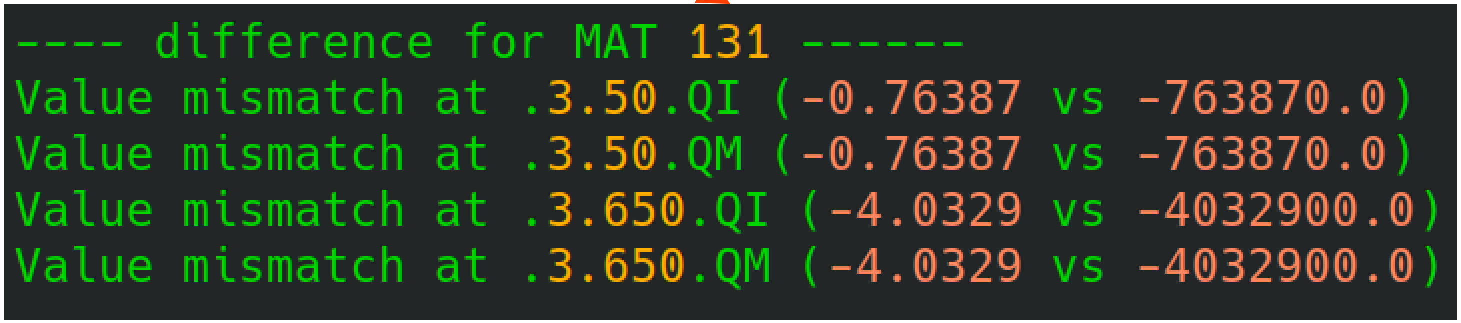


#	Material	Source	Emax [MeV]
1	1-H-1	JENDL-1	3000
2	1-H-2	ENDF/B-VII	150
3	1-H-3	ENDF/B-VII	20
4	2-He-3	ENDF/B-VII	20
5	3-Li-6	JENDL-4.0/HE	200
6	3-Li-7	JENDL-4.0/HE	200
7	4-Be-9	ENDF/B-VII	113
8	5-B-10	ENDF/B-VII	3
9	5-B-11	ENDF/B-?????	200



```
from endf_parserpy.endf_parser import BasicEndfParser
from endf_parserpy.debugging_utils import compare_objects
parser = BasicEndfParser()
fendl_endf = parser.parsefile(fendl_filename)
other_endf = parser.parsefile(other_endffile)
del fendl_endf[1][451]
del other_endf[1][451]
compare_objects(fendl_endf, other_endf, fail_on_diff=False)
```

FENDL 3.2b = ENDF/B.VII.0



```
---- difference for MAT 131 ------
Value mismatch at .3.50.QI (-0.76387 vs -763870.0)
Value mismatch at .3.50.QM (-0.76387 vs -763870.0)
Value mismatch at .3.650.QI (-4.0329 vs -4032900.0)
Value mismatch at .3.650.QM (-4.0329 vs -4032900.0)
```

Comparison with ENDF/B-VII.1

Summary

- New Python package for ENDF-6 formatted files enabling:
  - Reading
  - Writing
  - Translating
  - Verifying
- Supports (nearly) the entire ENDF-6 format
- Provably correct implementation
- Assembling an ENDF-6 file becomes much easier:
  - Working with symbol names instead of locations
  - Leveraging powerful data science capabilities of Python