LA-UR-22-30925

Approved for public release; distribution is unlimited.

Title: Introduction to the ENDF format - Reading and Manipulating ENDF files with ENDFtk

Author(s): Haeck, Wim
Gibson, Nathan Andrew

Intended for: 2022 MCNP User Symposium, 2022-10-17/2022-10-21 (Los Alamos, New Mexico, United States)

Issued: 2022-10-17

Introduction to the ENDF format
Reading and Manipulating ENDF files with ENDFtk

W. Haeck, N. Gibson

2022 MCNP User Symposium, October 17-21, 2022

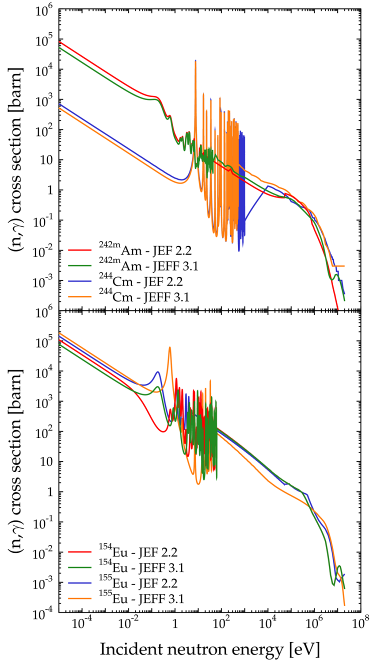
Outline

- Introduction
- What is ENDF?
- Overview of the ENDF format and structure
- The ENDFtk toolkit

Introduction

- Nuclear data is everything needed to describe particle transport and nuclear processes
  - Nuclear reaction data
  - Cross sections, secondary particle angular distributions, etc.
  - Radioactive decay data
  - Uncertainties (covariance data)

- Used by particle simulation codes at LANL
  - Monte Carlo particle transport with MCNP
  - Deterministic particle transport with PARTISN
  - Material irradiation with CINDER

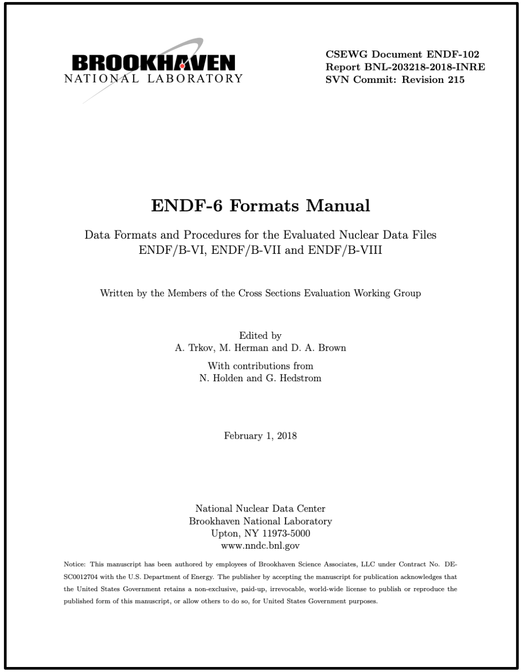


What is ENDF?

- ENDF = Evaluated Nuclear Data File/Format

- Evaluated Nuclear Data Format
  - Format specification for storing/organising nuclear data
  - Format versions are designated with an Arabic number
    - ENDF-6 is the current format version

- Evaluated Nuclear Data File
  - The name of the US nuclear data library
  - Library versions are designated with a Roman numeral
    - ENDF/B-VIII.0 is the latest version, released in February 2018
    - ENDF/B-VIII.1 is currently in beta, to be released in 2024



BROOKHAVEN NATIONAL LABORATORY

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

ENDF-6 Formats Manual

Data Formats and Procedures for the Evaluated Nuclear Data Files
ENDF/B-VI, ENDF/B-VII and ENDF/B-VIII

Written by the Members of the Cross Sections Evaluation Working Group

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

With contributions from
N. Holden and G. Hedstrom

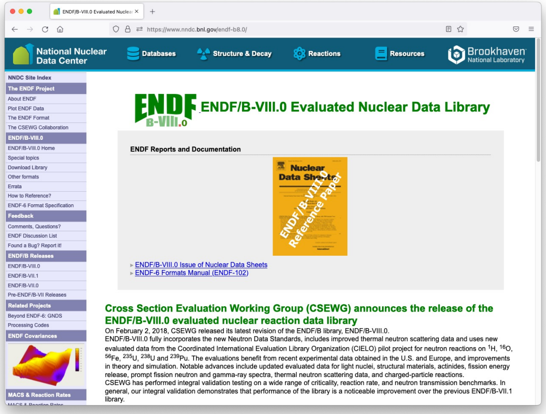
February 1, 2018

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.

What is ENDF?

- ENDF was created in the mid-1960s
  - The format has gone through 6 iterations
  - Used for 8 generations of the ENDF/B library
    - ENDF/B-I in July 1968
    - ENDF/B-VIII.0 in February 2018
    - The future ENDF/B-VIII.1 library

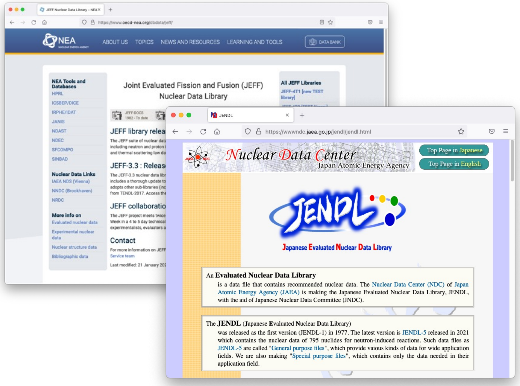


- ENDF is developed and maintained by the NNDC and coordinated by CSEWG
  - NNDC: National Nuclear Data Centre at BNL
  - CSEWG: Cross Section Evaluation Working Group
    - Collaboration between national labs, universities and nuclear industry from the US and Canada
    - International oganisations such as the International Atomic Energy Agency (IAEA)

What is ENDF?

- The ENDF format is the de facto standard for all nuclear data libraries

- There are multiple “independent” libraries
  - Europe: Joint European Fission and Fusion (JEFF)
  - Japan: Japanese Evaluated Nuclear Data Library (JENDL)
  - China: Chinese Evaluated Nuclear Data Library (CENDL)
  - Russia: BROND



- All libraries are freely available from different nuclear data centres:
  - In the US, this is the NNDC at Brookhaven National Laboratory (BNL)
  - There are many data centres in the world: OECD/NEA, IAEA/NDS, etc.

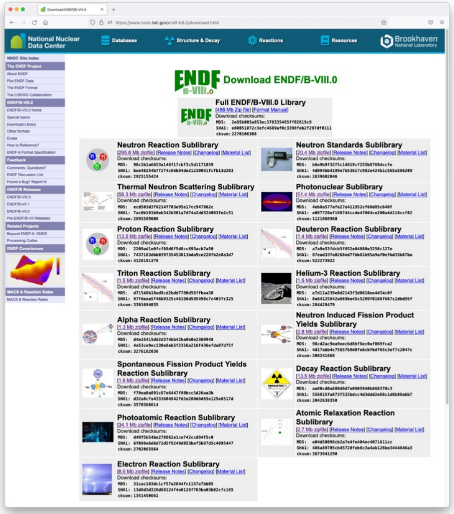
What is ENDF?

```
 9.223500+4 2.330248+2          0          0          0          09228 3 18    1
 1.934054+8 1.934054+8          0          0          1        8399228 3 18    2
        839          2                                            9228 3 18    3
 1.000000-5 0.00000+00 2.250000+3 0.00000+00 2.250000+3 2.634378+09228 3 18    4
 2.250014+3 2.668097+0 2.250056+3 2.769988+0 2.250112+3 2.907176+09228 3 18    5
 2.250251+3 3.252747+0 2.250307+3 3.389935+0 2.250363+3 3.525523+09228 3 18    6
 2.250419+3 3.658711+0 2.250470+3 3.778100+0 2.250517+3 3.884190+09228 3 18    7
 2.250563+3 3.987480+0 2.250598+3 4.063173+0 2.250633+3 4.136965+09228 3 18    8
 2.250668+3 4.209058+0 2.250703+3 4.279151+0 2.250738+3 4.347343+09228 3 18    9
 2.250772+3 4.413436+0 2.250807+3 4.477529+0 2.250842+3 4.539621+09228 3 18   10
```

If you can read this, you are ready to join the nuclear data team. Send an email to nucldata@lanl.gov to apply.

Overview of the ENDF format and structure

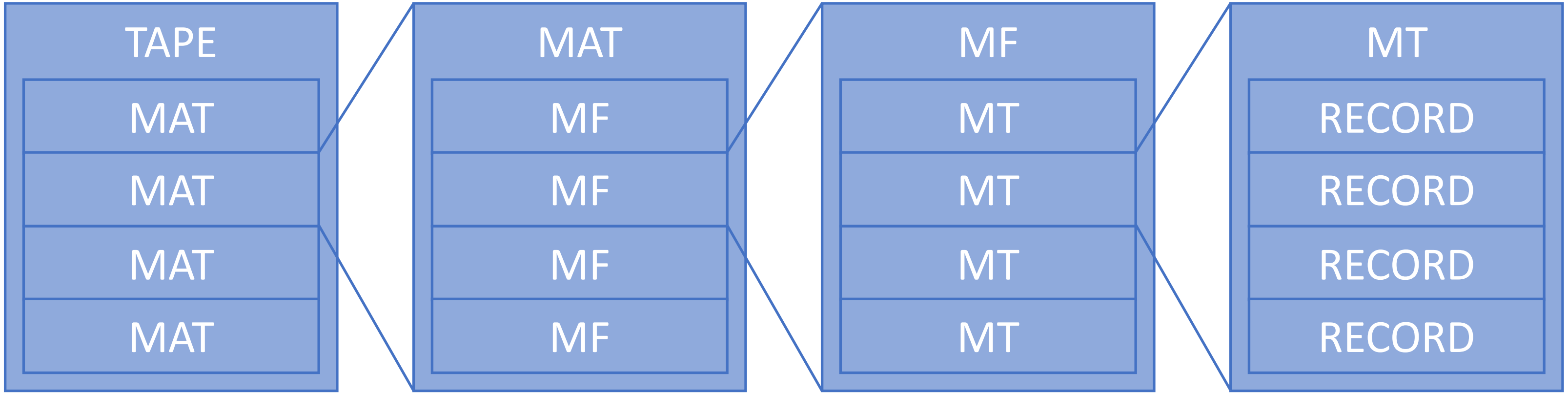
- An ENDF library has multiple sub-libraries
  - Incident particle data: n, p, d, t, ^{3}He, α
  - Photonuclear and photoatomic data
  - Thermal scattering data for crystals and molecules
  - Radioactive decay data
  - Neutron induced and spontaneous fission yields
  - Atomic relaxation data
  - Electron interaction data



- Each sub-library is physically separated and stored in one or more “tapes”
  - ENDF jargon dating back to the time of magnetic tapes and punch cards

Overview of the ENDF format and structure

- Each tape is structured as a sequence
  - Materials designated by the `MAT` number
  - Files designated by the `MF` number
  - Sections designated by the `MT` number
  - A section is a sequence of records

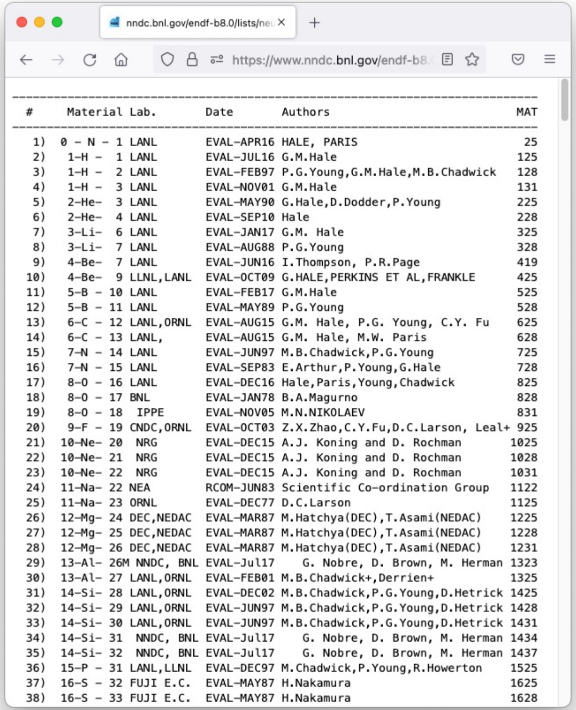


Overview of the ENDF format and structure

- A material is identified by its `MAT` number
  - A specific nuclide, an element, a molecule, etc.
  - Between `1` and `9999`

- Some sublibraries impose rules for isotopes
  - `Z * 100 + 25` for the first stable isotope
  - Decremented/incremented for the previous/next isotope
  - Numbers in between for metastable states

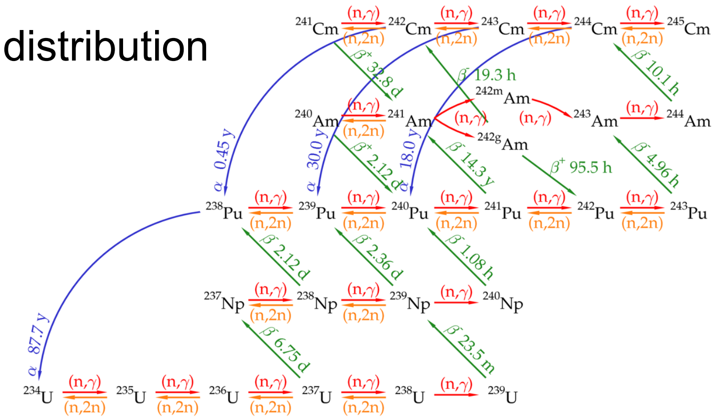
  - For example:
    - `125` for H1, `9228` for U235
    - `9546` for Am242, `9547` for Am242m



#	Material	Lab.	Date	Authors	MAT
1)	0 - N - 1	LANL	EVAL-APR16	HALE, PARIS	25
2)	1-H - 1	LANL	EVAL-JUL16	G.M.Hale	125
3)	1-H - 2	LANL	EVAL-FEB97	P.G.Young,G.M.Hale,M.B.Chadwick	128
4)	1-H - 3	LANL	EVAL-NOV01	G.M.Hale	131
5)	2-He- 3	LANL	EVAL-MAY90	G.Hale,D.Dodder,P.Young	225
6)	2-He- 4	LANL	EVAL-SEP10	Hale	228
7)	3-Li- 6	LANL	EVAL-JAN17	G.M. Hale	325
8)	3-Li- 7	LANL	EVAL-AUG88	P.G.Young	328
9)	4-Be- 7	LANL	EVAL-JUN16	I.Thompson, P.R.Page	419
10)	4-Be- 9	LLNL,LANL	EVAL-OCT09	G.HALE,PERKINS ET AL,FRANKLE	425
11)	5-B - 10	LANL	EVAL-FEB17	G.M.Hale	525
12)	5-B - 11	LANL	EVAL-MAY89	P.G.Young	528
13)	6-C - 12	LANL,ORNL	EVAL-AUG15	G.M. Hale, P.G. Young, C.Y. Fu	625
14)	6-C - 13	LANL,	EVAL-AUG15	G.M. Hale, M.W. Paris	628
15)	7-N - 14	LANL	EVAL-JUN97	M.B.Chadwick,P.G.Young	725
16)	7-N - 15	LANL	EVAL-SEP83	E.Arthur,P.Young,G.Hale	728
17)	8-O - 16	LANL	EVAL-DEC16	Hale,Paris,Young,Chadwick	825
18)	8-O - 17	BNL	EVAL-JAN78	B.A.Magurno	828
19)	8-O - 18	IPPE	EVAL-NOV05	M.N.NIKOLAEV	831
20)	9-F - 19	CNDC,ORNL	EVAL-OCT03	Z.X.Zhao,C.Y.Fu,D.C.Larson, Leal+	925
21)	10-Ne- 20	NRG	EVAL-DEC15	A.J. Koning and D. Rochman	1025
22)	10-Ne- 21	NRG	EVAL-DEC15	A.J. Koning and D. Rochman	1028
23)	10-Ne- 22	NRG	EVAL-DEC15	A.J. Koning and D. Rochman	1031
24)	11-Na- 22	NEA	RCOM-JUN83	Scientific Co-ordination Group	1122
25)	11-Na- 23	ORNL	EVAL-DEC77	D.C.Larson	1125
26)	12-Mg- 24	DEC,NEDAC	EVAL-MAR87	M.Hatchya(DEC),T.Asami(NEDAC)	1225
27)	12-Mg- 25	DEC,NEDAC	EVAL-MAR87	M.Hatchya(DEC),T.Asami(NEDAC)	1228
28)	12-Mg- 26	DEC,NEDAC	EVAL-MAR87	M.Hatchya(DEC),T.Asami(NEDAC)	1231
29)	13-Al- 26M	NNDC, BNL	EVAL-Jul17	G. Nobre, D. Brown, M. Herman	1323
30)	13-Al- 27	LANL,ORNL	EVAL-FEB01	M.B.Chadwick+,Derrien+	1325
31)	14-Si- 28	LANL,ORNL	EVAL-DEC02	M.B.Chadwick,P.G.Young,D.Hetrick	1425
32)	14-Si- 29	LANL,ORNL	EVAL-JUN97	M.B.Chadwick,P.G.Young,D.Hetrick	1428
33)	14-Si- 30	LANL,ORNL	EVAL-JUN97	M.B.Chadwick,P.G.Young,D.Hetrick	1431
34)	14-Si- 31	NNDC, BNL	EVAL-Jul17	G. Nobre, D. Brown, M. Herman	1434
35)	14-Si- 32	NNDC, BNL	EVAL-Jul17	G. Nobre, D. Brown, M. Herman	1437
36)	15-P - 31	LANL,LLNL	EVAL-DEC97	M.Chadwick,P.Young,R.Howerton	1525
37)	16-S - 32	FUJI E.C.	EVAL-MAY87	H.Nakamura	1625
38)	16-S - 33	FUJI E.C.	EVAL-MAY87	H.Nakamura	1628

Overview of the ENDF format and structure

- Files identified by their `MF` number store specific types of data:
  - `MF1`: descriptive and miscellaneous data
  - `MF2`: resonance parameters
  - `MF3`: cross section data
  - `MF4`: secondary particle angular distribution
  - `MF5`: secondary particle energy distribution
  - `MF6`: correlated secondary particle angle-energy distribution
  - `MF7`: thermal scattering data
  - `MF8`: radioactive decay data
  - `MF12` to `MF15`: photon data
  - `MF31` to `MF35`: covariance data
  - And there are even more …

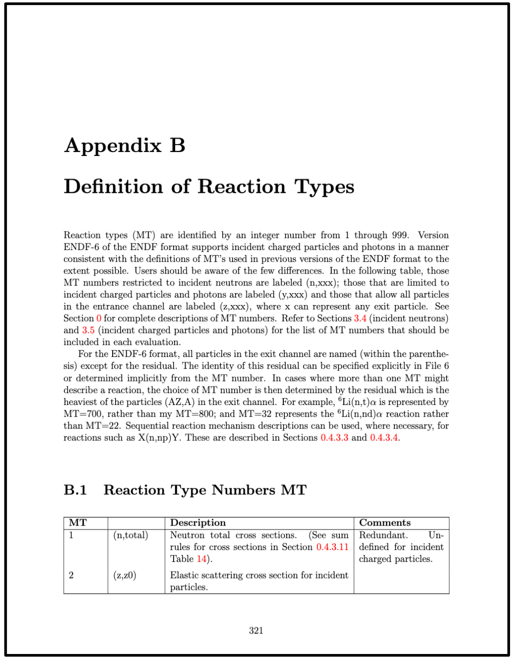


Overview of the ENDF format and structure

- Sections designated by an `MT` number store specific “reaction data”

- These can be “simple” reactions
  - `MT102` (neutron capture), `MT51` to `MT91` (inelastic levels)
- These can be “summation” reactions
  - `MT4` (inelastic scattering, sum of `MT51` to `MT91`)
- These can be “special” sections
  - `MT451` (descriptive data, only in `MF1`)
  - `MT151` (resonance parameters, only in `MF2`)

- `MT` numbers are limited to `1-999`



Appendix B

Definition of Reaction Types

Reaction types (MT) are identified by an integer number from 1 through 999. Version ENDF-6 of the ENDF format supports incident charged particles and photons in a manner consistent with the definitions of MT’s used in previous versions of the ENDF format to the extent possible. Users should be aware of the few differences. In the following table, those MT numbers restricted to incident neutrons are labeled (n,xxx); those that are limited to incident charged particles and photons are labeled (y,xxx) and those that allow all particles in the entrance channel are labeled (z,xxx), where x can represent any exit particle. See Section 0 for complete descriptions of MT numbers. Refer to Sections 3.4 (incident neutrons) and 3.5 (incident charged particles and photons) for the list of MT numbers that should be included in each evaluation.

For the ENDF-6 format, all particles in the exit channel are named (within the parenthesis) except for the residual. The identity of this residual can be specified explicitly in File 6 or determined implicitly from the MT number. In cases where more than one MT might describe a reaction, the choice of MT number is then determined by the residual which is the heaviest of the particles (AZ,A) in the exit channel. For example, ^{6}Li(n,t)α is represented by MT=700, rather than my MT=800; and MT=32 represents the ^{6}Li(n,nd)α reaction rather than MT=22. Sequential reaction mechanism descriptions can be used, where necessary, for reactions such as X(n,np)Y. These are described in Sections 0.4.3.3 and 0.4.3.4.

B.1 Reaction Type Numbers MT

MT		Description	Comments
1	(n,total)	Neutron total cross sections. (See sum rules for cross sections in Section 0.4.3.11 Table 14).	Redundant. Undefined for incident charged particles.
2	(z,z0)	Elastic scattering cross section for incident particles.	

321

Overview of the ENDF format and structure

- Only 6 record types to store information:
  - `TEXT`: stores just text
  - `CONT`: 2 floating point numbers and 4 integers
  - `LIST`: a list of values
  - `TAB1`: a one dimensional function y = f(x)
  - `TAB2`: a two dimensional function z = f(x,y), used in combination with other records
  - `INTG`: a correlation matrix (used for covariance data)

- Special cases:
  - `HEAD`: a `CONT` record at the beginning of each section
  - `TEND`, `MEND`, `FEND`, `SEND`: records to signal the end of a tape, material, file or section

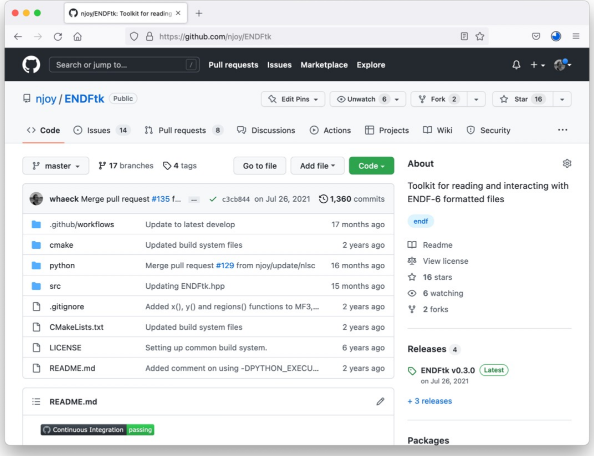
Overview of the ENDF format and structure

```
 9.223500+4 2.330248+2          0          0          0          09228 3 18    1
 1.934054+8 1.934054+8          0          0          1        8399228 3 18    2
        839          2                                            9228 3 18    3
 1.000000-5 0.00000+00 2.250000+3 0.00000+00 2.250000+3 2.634378+09228 3 18    4
 2.250014+3 2.668097+0 2.250056+3 2.769988+0 2.250112+3 2.907176+09228 3 18    5
 2.250251+3 3.252747+0 2.250307+3 3.389935+0 2.250363+3 3.525523+09228 3 18    6
 2.250419+3 3.658711+0 2.250470+3 3.778100+0 2.250517+3 3.884190+09228 3 18    7
 2.250563+3 3.987480+0 2.250598+3 4.063173+0 2.250633+3 4.136965+09228 3 18    8
 2.250668+3 4.209058+0 2.250703+3 4.279151+0 2.250738+3 4.347343+09228 3 18    9
 2.250772+3 4.413436+0 2.250807+3 4.477529+0 2.250842+3 4.539621+09228 3 18   10
```

- Lines consisting of 80 characters
  - Six 11 characters columns for floats and integers
  - Four columns for the MAT, MF, MT and sequence number

The ENDFtk toolkit

- ENDFtk: https://github.com/njoy/ENDFtk
  - A format component developed in the NJOY modernisation project
  - Reading, writing and manipulate ENDF files
  - Using a C++ and Python API at the same time



MF	Description	ENDFtk support	Python support
1	General information	Full	Full
2	Resonance parameters	Full	Full
3	Reaction cross sections	Full	Full
4	Angular distributions	Full	Full
5	Energy distributions	Full	Full
6	Product energy-angle distributions	Full	Full
7	Thermal neutron scattering law data	Full	Full
8	Decay and fission product yields	Partial	Full
9	Multiplicities of radioactive products	Full	Full
10	Radioactive nuclide production	Full	Full
12	Photon production yield data	Full	Full
13	Photon production cross sections	Full	Full
14	Photon angular distributions	Full	Full
15	Continuous photon energy spectra	Full	Full
23	Photon interaction cross sections	Full	Full
26	Photo-atomic distributions	Full	Full
27	Atomic form factor functions	Full	Full
28	Atomic relaxation data	Full	Full
30	Covariance of model parameters	None	None
31	Covariances of fission	Soon	Soon
32	Covariances of resonance parameters	Soon	Soon
33	Covariances of cross sections	Full	Full
34	Covariances of angular distributions	Full	Full
35	Covariances of energy distributions	Soon	Soon
40	Covariances for nuclide production	Soon	Soon

The ENDFtk toolkit

- Prerequisites:
  - git
  - cmake 3.15 or higher
  - a C++-17 compliant compiler such as gcc-7 or higher
  - Python 3.5 or higher

- Installation instructions:

```
git clone https://github.com/njoy/ENDFtk
cd ENDFtk
git checkout develop
mkdir build
cd build
cmake -DCMAKE_BUILD_TYPE=Release ../
make ENDFtk.python -j8
```