Consultancy Meeting on Model Code Output & Application Nuclear Data Form Structure
15 to 17 March 2021

output of the CCONE code and creation of ENDF-6 format file

O. Iwamoto

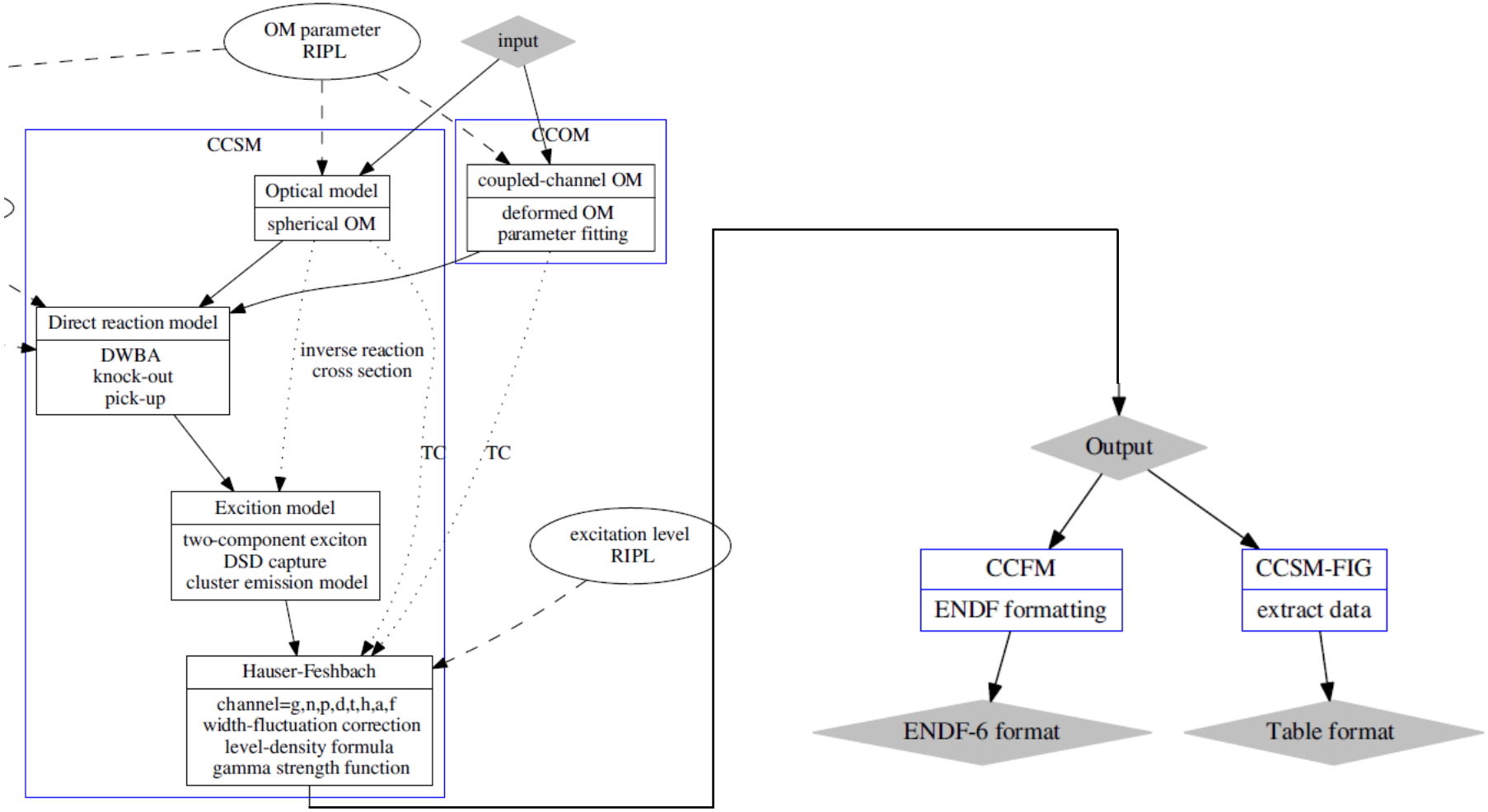
Japan Atomic Energy Agency

CCONE code

O. Iwamoto, JNST 44, 687 (2007)
O. Iwamoto, JNST 50, 409 (2013)
O. Iwamoto, NDS 118, 204 (2014)
O. Iwamoto et al., NDS 131, 259 (2016)

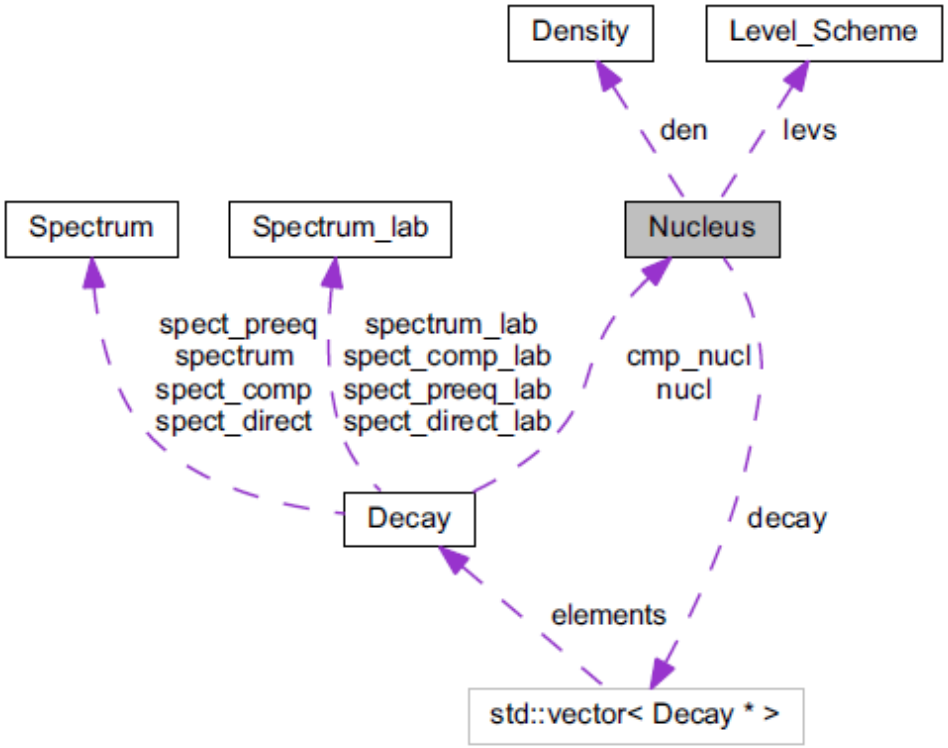
- Optical model
  - spherical optical model
  - coupled channel optical model (rotational band)
  - RIPL OMP data base
- DWBA
- cluster emission
  - pickup and knockout reaction systematics by Kalbach
  - Iwamoto-Harada model
- pre-equilibrium two component exciton model
- Hauser-Feshbach statistical model
  - channels: g, n, p, d, t, h, a, f
  - width fluctuation correction
- C++ object oriented programming

Flow of CCONE calculation

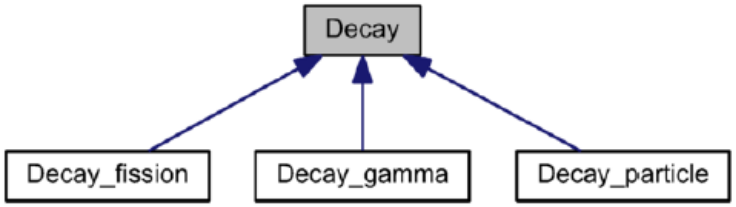


Example the class scructures

Relation of classes around the Nucleus class



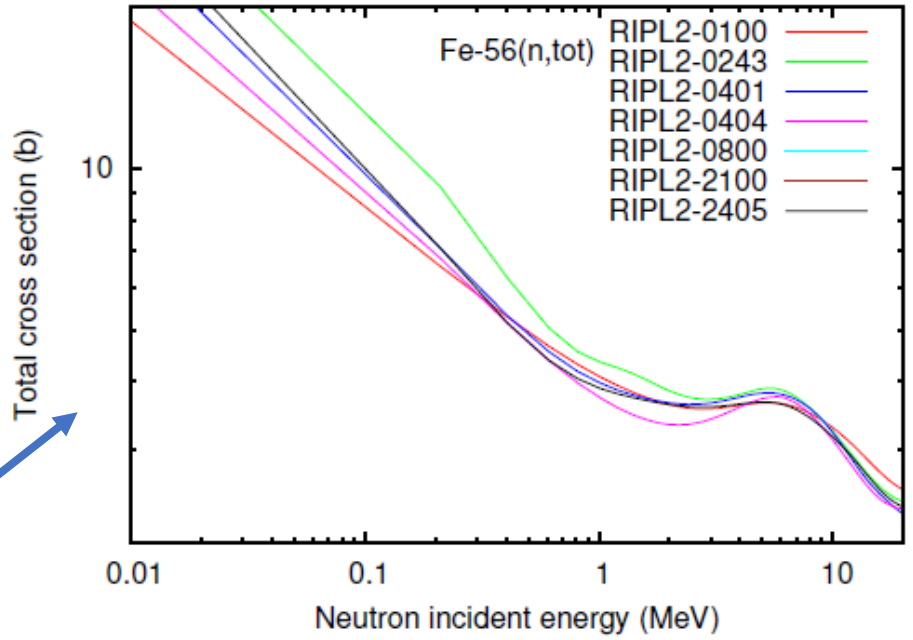
Class structure of the Decay class

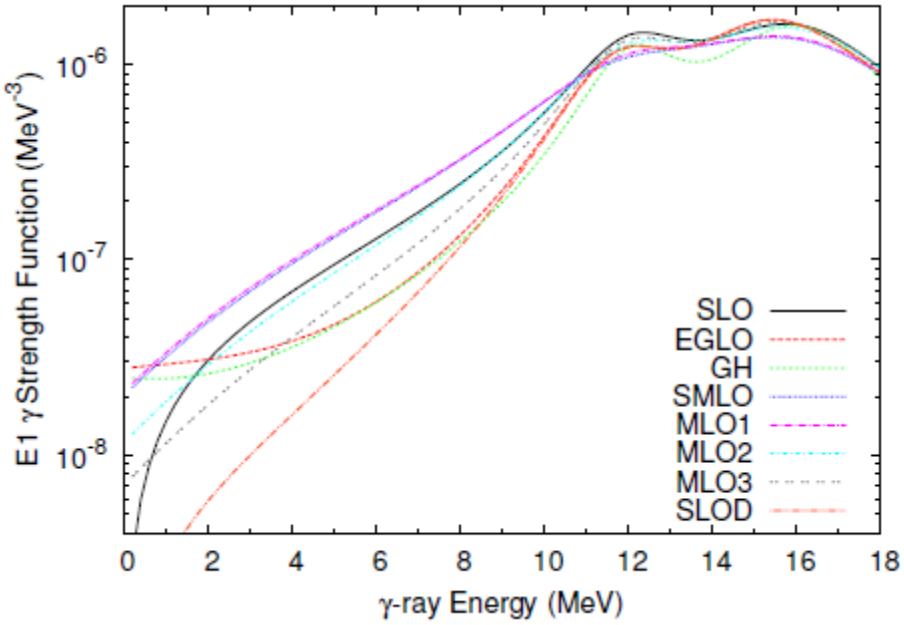


Example of input

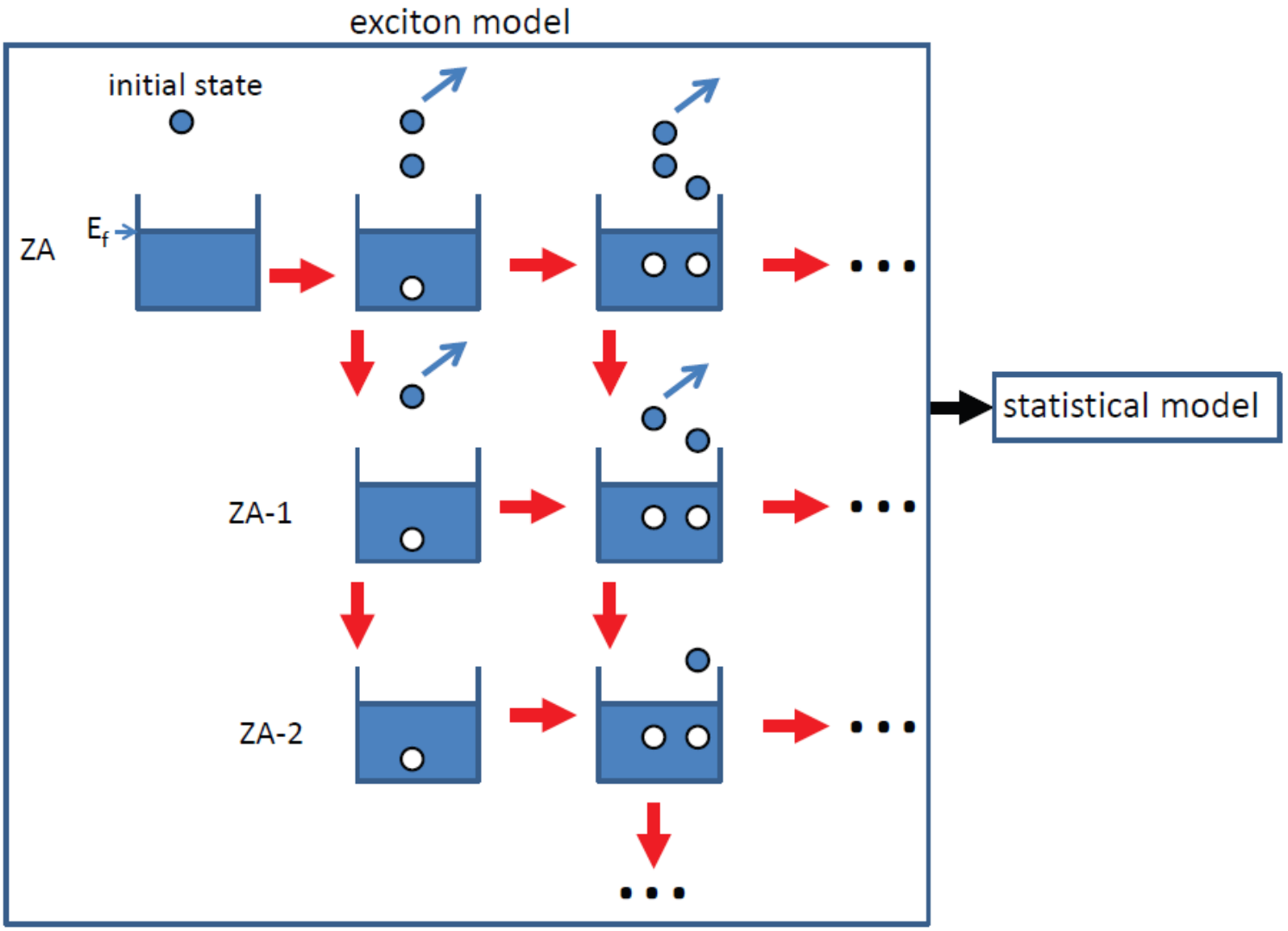
```
target fe-56
projectile n
energy ( 1 2 3 )
include common1.in
nthread 10
nucleus fe-57 {
    decay +( n ripl2-2405
           p ripl2-5405 )
    density +( ... )
    gamma +( type 1 ... )
}
nucleus fe-56 {
...
}
```

O. Iwamoto *et al.* Nuclear Data Sheets 131 (2016) 259

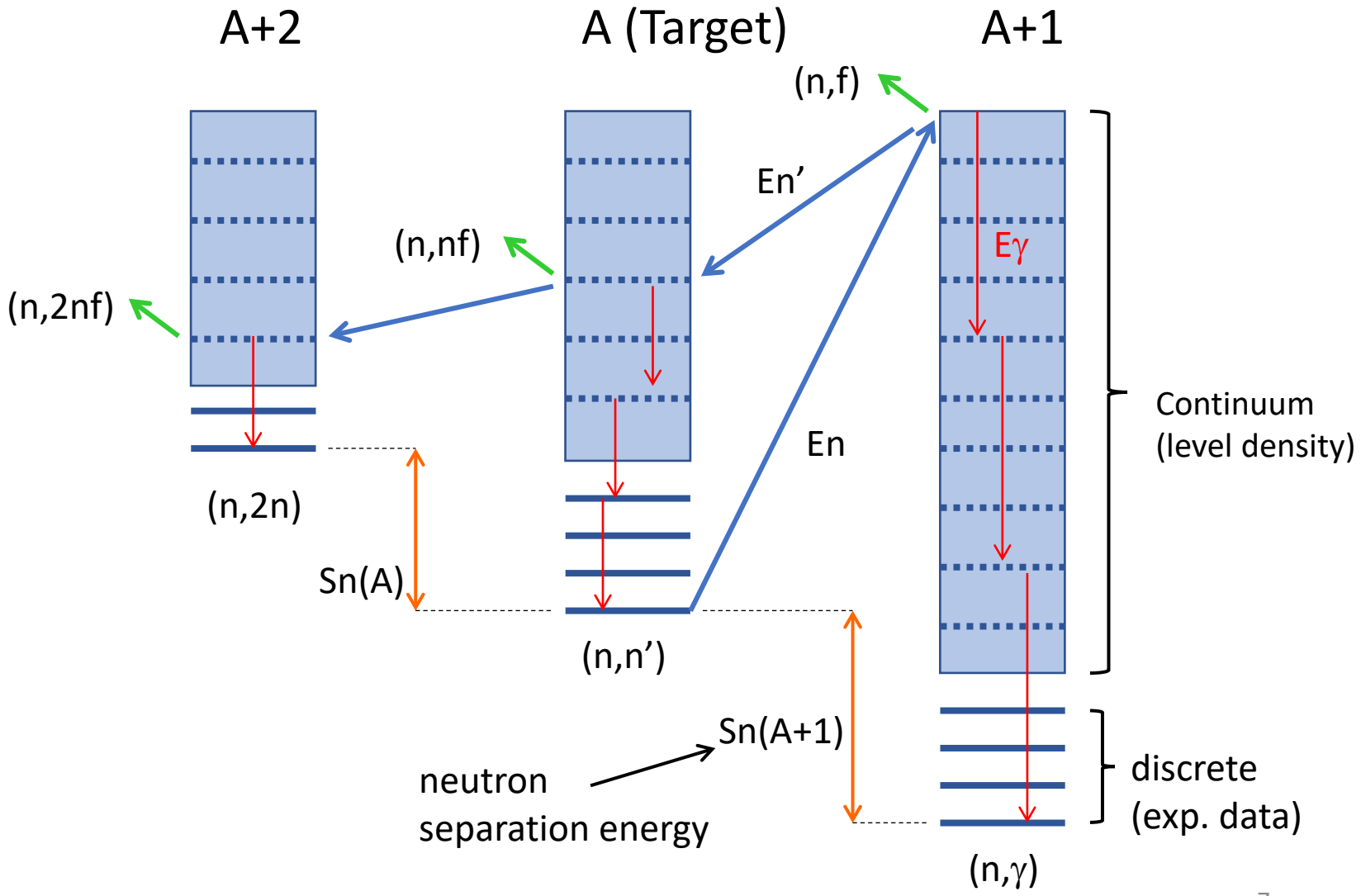




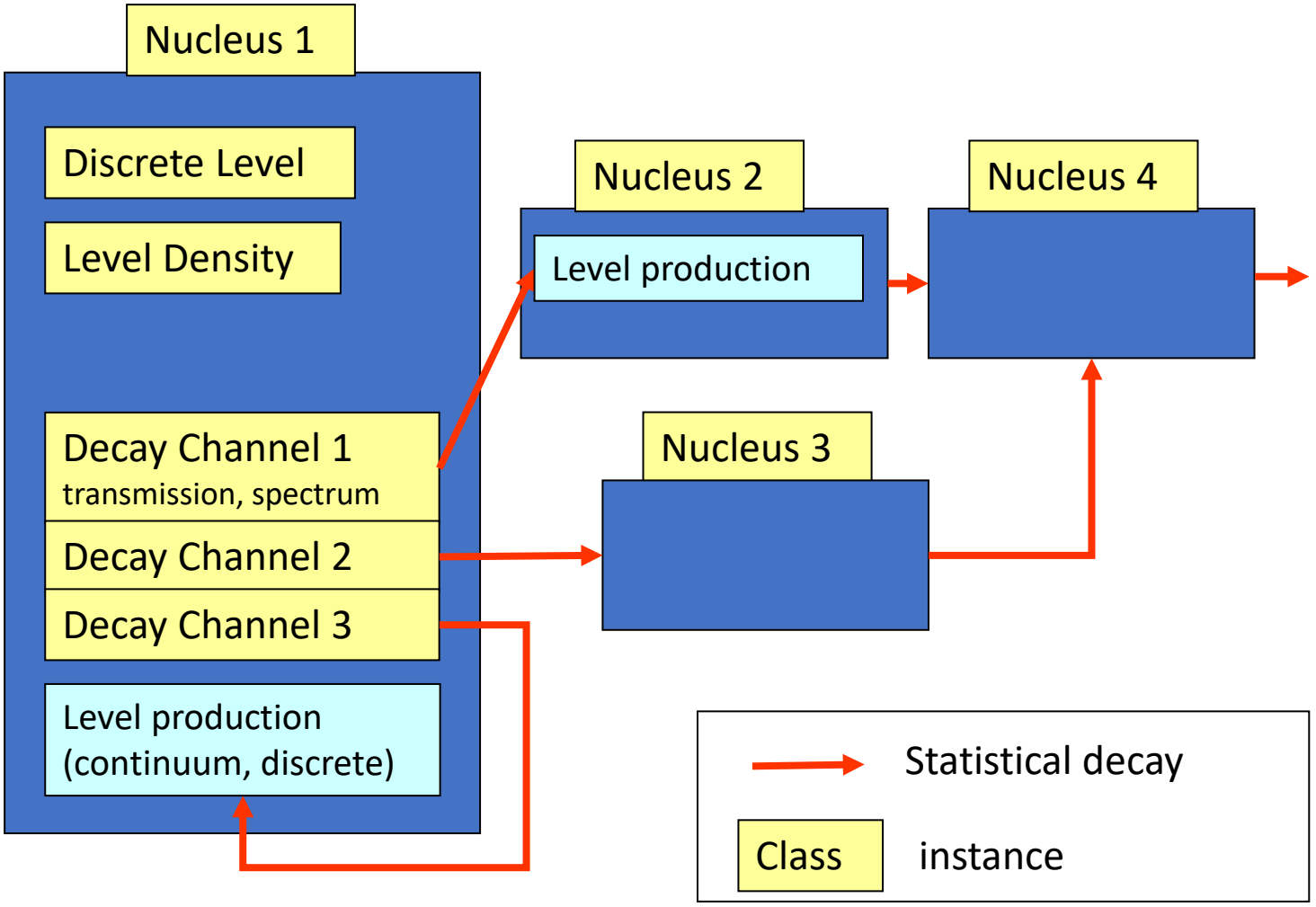
Multiple emission on exciton model



Decay chain on statistical model



Relation of instances of class

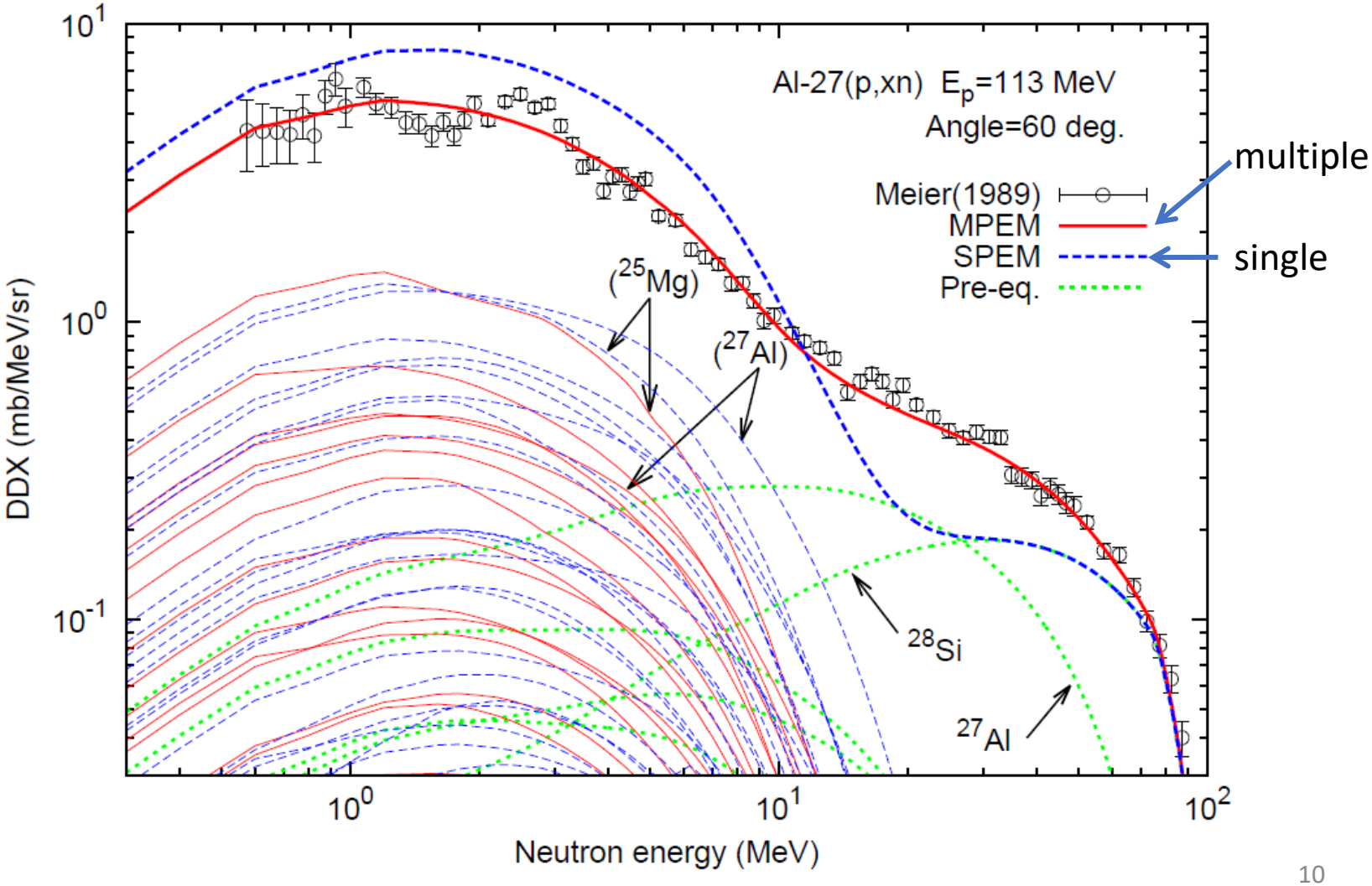


Outputs structure

Outputs is in a single file in the XML format.

```
<?xml version='1.0'?>
<ccsm>
…
<energy e="1">
<nucleus Z="26" A="57" za="26057">
<production final="1.929717e+00" initial="3.894552e+03">
# Fe57->Fe57+g
<spectrum particle="g" sigma="5.674690e+00"/>
# Fe57->Fe56+n
<spectrum particle="n" sigma="2.287526e+03"/>
...
</nucleus>
<nucleus Z="26" A="56" za="26056"/>
...
<inclusive>
<spectrum particle="g" sigma="5.767200e+02" type="dae" da="table" angle="">
…
</inclusive>
</energy>
</ccsm>
```

Neutron emission spectrum via various nuclei

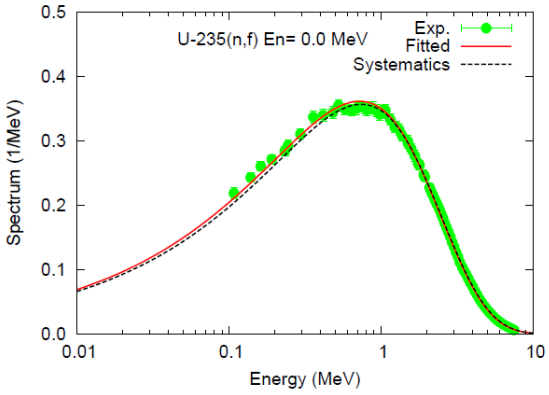


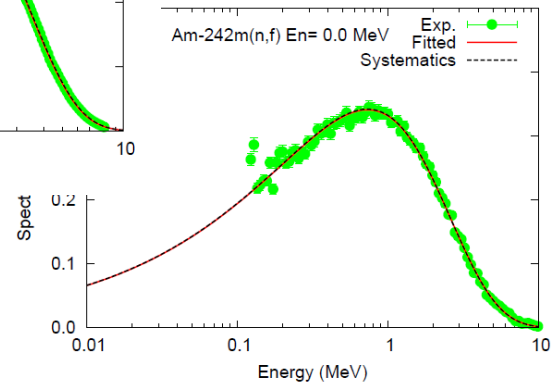
Fission spectrum

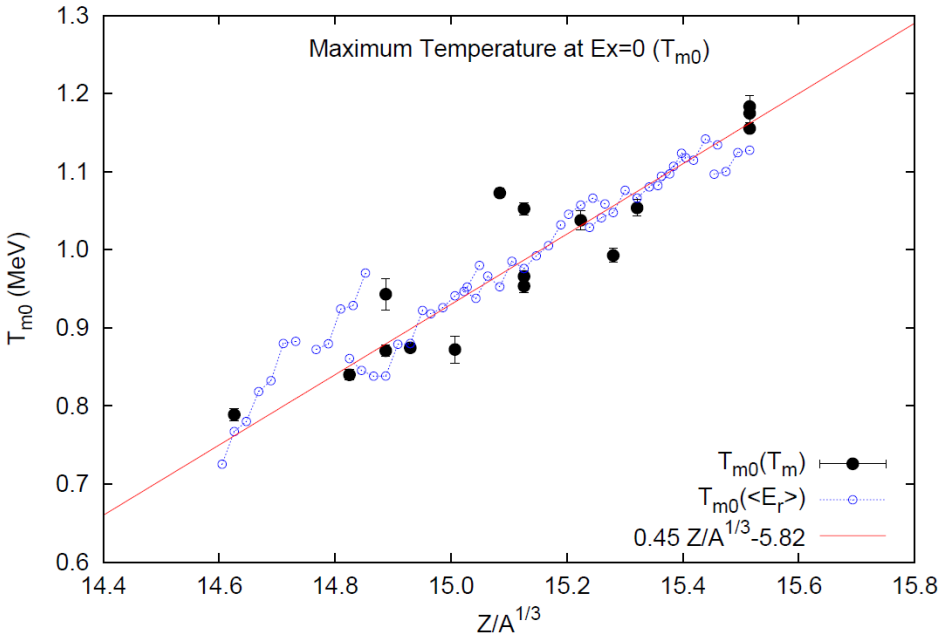
- Los Alamos (Madland-Nix '82)
- Parameters were derived by fitting to the experimental data

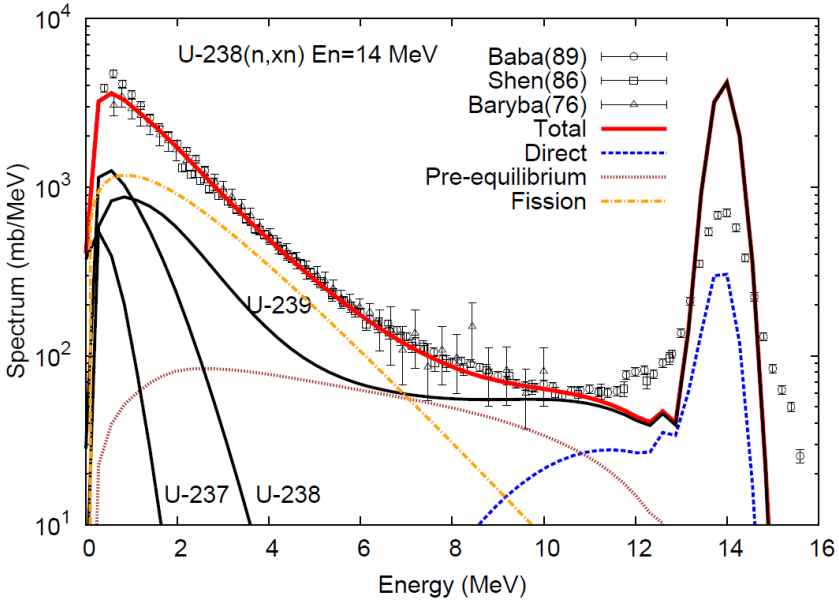
$$\phi(\epsilon) = k(T)\sigma_c(\epsilon)\epsilon e^{-\epsilon/T}$$

$$\Phi(\epsilon) = \int_0^{T_m} \phi(\epsilon)P(T)dT$$







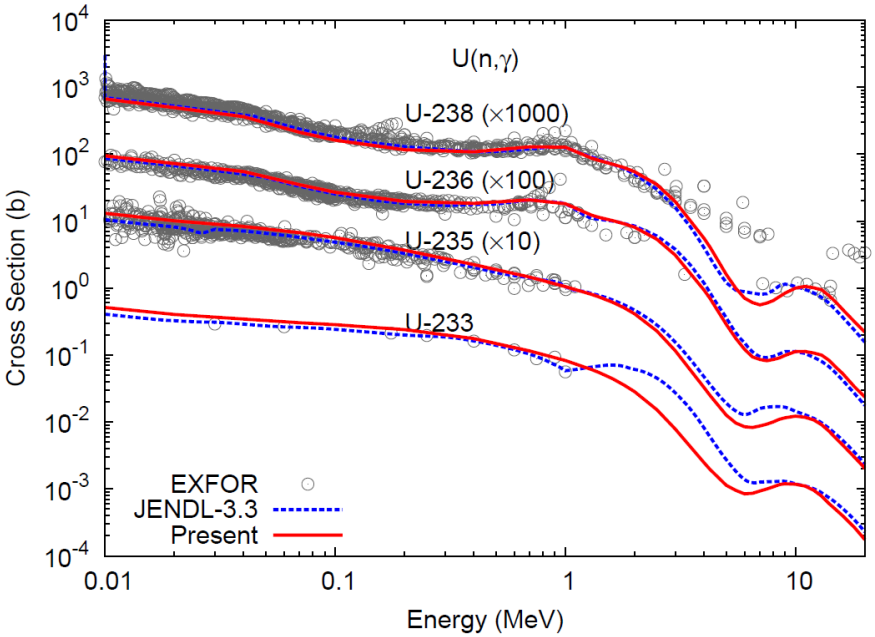


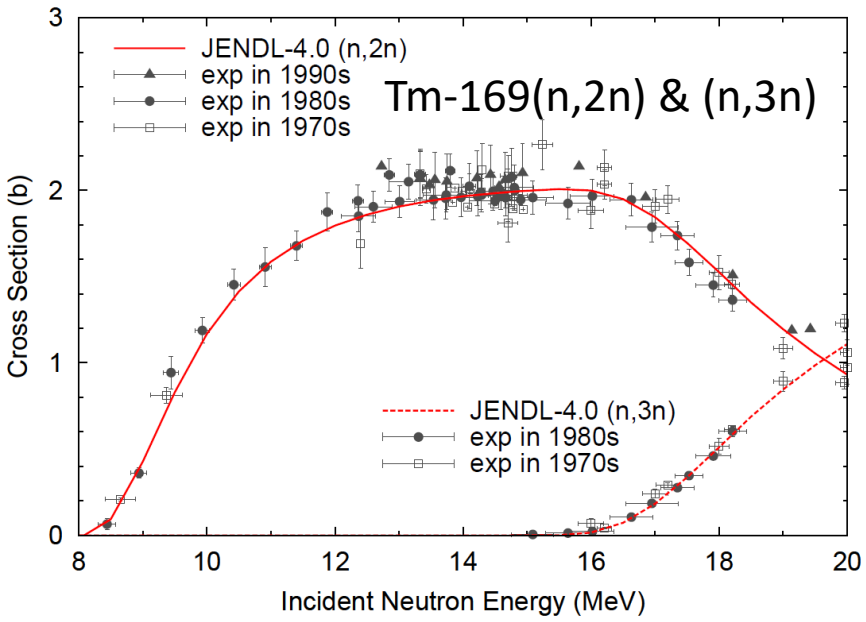
CCSM-FIG

```
<energy e="1">
<nucleus Z="26" A="57" za="26057">
<production final="1.929717e+00"/>
<nucleus Z="26" A="56" za="26056"/>
...
</energy>
<energy e="2">
<nucleus Z="26" A="57" za="26057">
<production final="1.689335e+00" />
...
</energy>
...
</ccsm>
```

Output in table format for plotting

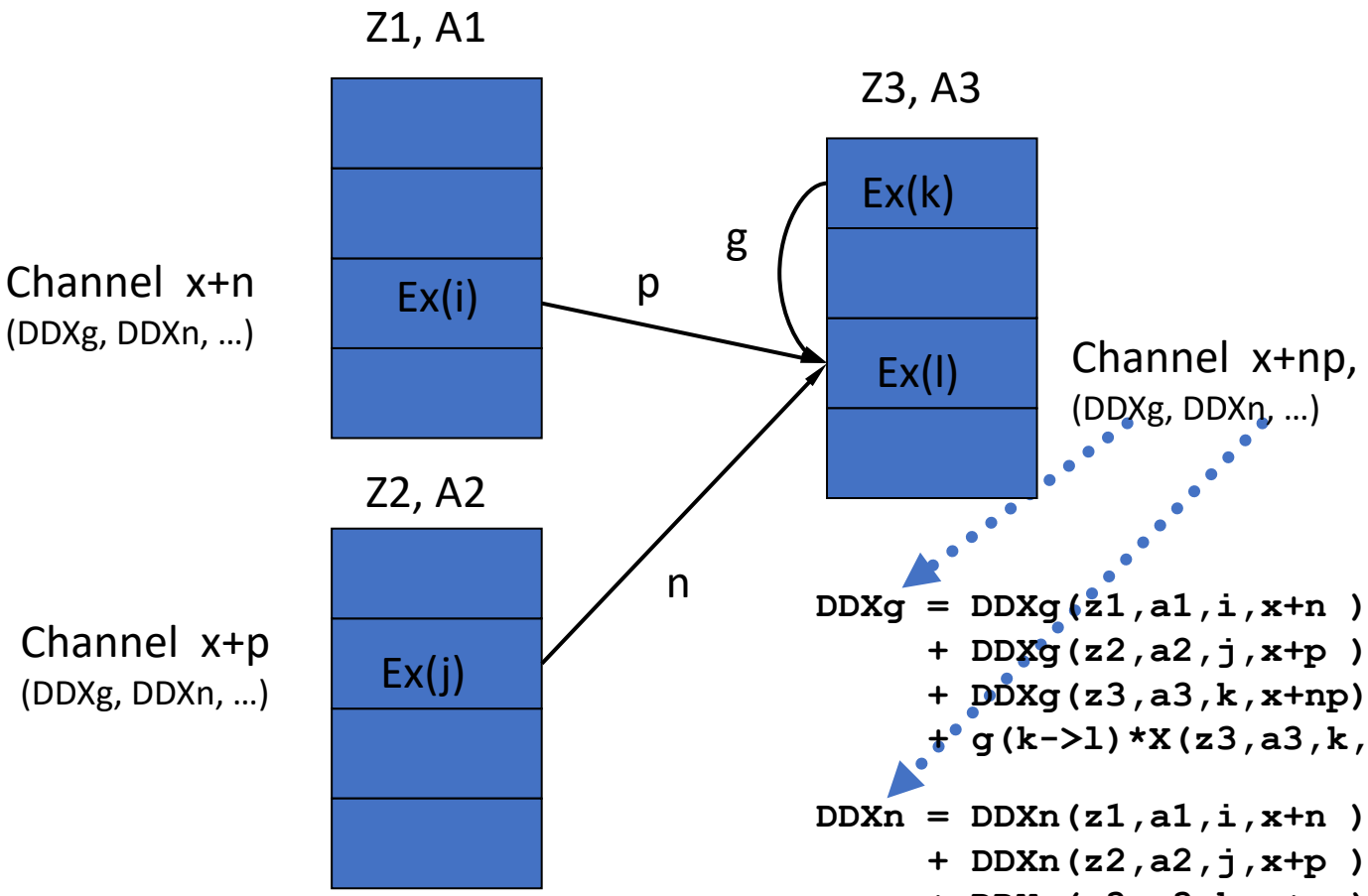
```
#energy/za: (  0)26057       (  1)26056       (  2)25056
 1.000e+00  1.929717e+00  5.712214e+02  0.000000e+00
 2.000e+00  1.689335e+00  9.474797e+02  0.000000e+00
 3.000e+00  1.674177e+00  1.078836e+03  0.000000e+00
...
```





Exclusive cross section and spectrum

Channel: { N_1n, N_2p, N_3d, … } N_i: no. of emitted particles




```
DDXg = DDXg(z1,a1,i,x+n )* Γp(i->l)/Γi
     + DDXg(z2,a2,j,x+p )* Γn(j->l)/Γj
     + DDXg(z3,a3,k,x+np)* Γg(k->l)/Γk
     + g(k->l)*X(z3,a3,k,x+np)/X(z3,a3,k)

DDXn = DDXn(z1,a1,i,x+n )* Γp(i->l)/Γi
     + DDXn(z2,a2,j,x+p )* Γn(j->l)/Γj
     + DDXn(z3,a3,k,x+np)* Γg(k->l)/Γk
     + n(j->l)*X(z2,a2,j,x+p )/X(z2,a2,j)
```

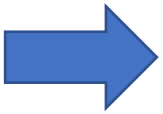
Spin & parity dependences were ignored.

CCFM

- Create ENDF-6 format file
- Ruby script
- Use XML library

CCSM output

```
<energy e='E1'>
<endf>
  <reaction
   channel='2n'
   sigma='X1'>
   <spectrum
    particle='n'>
    .....
   </spectrum>
    .....
   </reaction>
  .....
</endf>
```



```
xml[3][16]=
<section>
<head> ... </head>
<tab1>
   E1 X1
   E2 X2
</tab1>
   .....
xml[3][51]=
<section>
...
</section>
```

```
 2.605600+4 5.545443+1          0          0          0          02631 3 51    1
 0.000000+0-8.467780+5          0          0          1         622631 3 51    2
         62          2                                          2631 3 51    3
 8.620478+5 0.000000+0 1.000000+6 5.712167-1 1.300000+6 7.701051-12631 3 51    4
 1.600000+6 8.874914-1 2.000000+6 9.474352-1 2.122705+6 9.513962-12631 3 51    5
 2.500000+6 8.991819-1 2.705513+6 8.609547-1 2.994544+6 7.147537-12631 3 51    6
 ...
```

C++ class library for manipulation of ENDF-6 format file

- Manipulate the output data to satisfy the consistencies in ENDF-6 format.
- Developed to replace CRECTJ
- Read and write of ENDF-6 format file
- Add and delete of sections (MF/MT)
- Arithmetic operation of Tab1
- Interface for Ruby (Python) by SWIG

- CCFM uses to create total cross sections from partial ones, isomer ratios, etc.