

# Burnup Zr D Sp

# Depletion & Inventory

**MrOxe**

# FISPACT-II

# Getting Started

# Thomas Stainer United Kingdom Atomic Energy Authority

## **FISPACT-II workshop**

**June 13-15, 2018, OECD/NEA, Paris**



This work was funded by the RCUK Energy Programme  
[Grant number EP/P012450/1]



# FISPACT-II Resources

There are a number of valuable resources for FISPACT-II

- **Wiki** - <https://fispact.ukaea.uk/wiki>
- **Manual** - <http://www.ccfc.ac.uk/FISPACT-II/documentation/UKAEA-R18001.pdf>
- **Forum** - <https://fispact.ukaea.uk/forum/index.php>
- **Nuclear Data** - [http://www.ccfc.ac.uk/FISPACT-II/Release-4.0/nuclear\\_data/](http://www.ccfc.ac.uk/FISPACT-II/Release-4.0/nuclear_data/)
- **GitHub** - <https://github.com/fispact>
- **DockerHub** - <https://hub.docker.com/u/fispact/>

# FISPACT-II Architecture

- Command Line Tool
- Single executable program
- Takes **input** file and produces **output** files



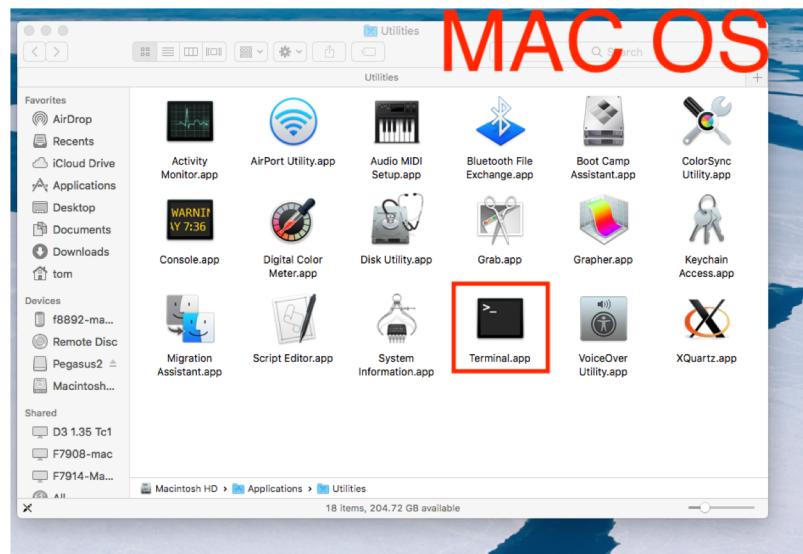
# Running FISPACT-II

No GUI

Command Line Program only!

Mac and Linux -> Terminal

Windows -> cmd



# A note on the Terminal

We will use **Ubuntu** – Linux (hopefully)

But, if you are using Windows outside of this course, a few exceptions:

- Fispact has '.exe' extension
- Use '\' not '/' for paths
- However, you can use the same files file for Linux and Windows!
- Input and output files are the same format

# Running FISPACT-II

Requires inputs as arguments  
Error if no arguments added

```
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact
FATAL ERROR: ...
```

Correct way

```
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact inputFile
```

# Running FISPACT-II

First argument always required – **input** file

Second argument is optional – **files** file

First argument is the input file minus the extension (.i)

```
[tstainer@mac:~/Dev/fispact]$ ls  
run1.i      run2.i      run3.i      files
```

```
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact run1
```

or

```
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact run1 files
```

# Exercise 1

Running a simple example

# Exercise 1

Did it run?

What are the outputs?

Not very exciting, but what did we do?

Let's start with the files file

# What is the files file?

Declares the nuclear data libraries  
to be used in the run

- **Text file**
- **Typically named ‘files’, ‘Files’ or ‘FILES’**
- **But can be named anything e.g files.tendl2015.run1**
- **14 labels required + some optional**
- **Order does not matter**
- **Use linux style ‘/’ for paths**

# What is the files file?

```

# index of nuclides to be included
ind_nuc      /nuclear_data/TENDL2015data/tendl15_decay12_index

# Library cross section data
xs_endf     /nuclear_data/TENDL2015data/tal2015-n/gxs-709

# Library probability tables for self-shielding
prob_tab    /nuclear_data/TENDL2015data/tal2015-n/tp-709-294

# Library fission data
fy_endf     /nuclear_data/GEFY61data/gefy61_nfy

# Library spontaneous fission data
sf_endf     /nuclear_data/GEFY61data/gefy61_sf

# Library decay data
dk_endf     /nuclear_data/decay/decay_2012

# Library regulatory data
hazards      /nuclear_data/decay/hazards_2012
clear        /nuclear_data/decay/clear_2012
a2data       /nuclear_data/decay/a2_2012

# gamma attenuation data
absorp       /nuclear_data/decay/abs_2012

# fluxes file
fluxes       fluxes

# collapsed cross section data (in and out)
collapxi    COLLAPX
collapxo    COLLAPX

# condensed decay and fission data (in and out)
arrayx      ARRAYX

```

# Nuclear Data

FISPACT-II can process a lot of different nuclear data libraries

Full list of libraries can be found here:

<http://fispact.ukaea.uk/nuclear-data/downloads>

Not going to focus on nuclear data, but one small note...

## TENDL recommended

**FISPACT-II development has run in parallel with that of the  
TALYS-based Evaluated Nuclear Data Library – TENDL**

**Complete package tailored for all application needs:**

- **Nuclear fission and fusion, nuclear fuel cycle,**
- **Accelerator physics, isotope production,**
- **Material characterization, storage and life cycle,**
- **earth exploration, astrophysics, homeland security**

# The FISPACT-II input file

The input file governs everything for the run

- Output types
- Inventory
- Solver settings
- Pathways
- Shelf Shielding
- ...

# The FISPACT-II input file

- Text file
- Order does matter, unlike the files file
- No default name, unlike the files file
- Input file must have .i extension
- Use of keywords to perform different actions
- Allows comments with '<< comment here >>'

97 keywords in all!

# The FISPACT-II input file

3 sections to the input file

- **Control** – 32 possible keywords
- **Initialisation** – 64 possible keywords
- **Inventory** – 41 possible keywords

$$32 + 64 + 41 \neq 97$$

Many keywords can be used in each input section

Full list: [https://fispact.ukaea.uk/wiki/FISPACT-II\\_keywords](https://fispact.ukaea.uk/wiki/FISPACT-II_keywords)

# Input file example

The input from exercise 1

```
<< CLOBBER tells fispact to overwrite existing files of same name >>
CLOBBER
<< GETXS 1 tells fispact to get the cross section data and do collapse >>
GETXS 1 709
<< GETDECAY 1 tells fispact to get the decay data and do condense >>
GETDECAY 1
<< FISPACT signals end of control section >>
FISPACT
* example 1
<< END is required to signal end of input >>
END
* end of example
```

No initialization or inventory sections – no output

# Exercise 2

Get the collapsed cross section data



# Exercise 2

Check the output file (ex2.out)

The cross sections should be printed for each reaction

```
| THIS RUN
| timestamp:          09:52:10 5 June 2018
| fileroot :          ex2
| name of FILES file: files
| FISPACT title:     * example 1

| See the ex2.log file
| and summary details at the end of this file for further information on files used by this run
=====
=====
```

```
1          C O L L A P S E D   C R O S S   S E C T I O N S   I N   B A R N S

NOTE: The diagnostic cross-sections for dpa (n,D--) and for kerma (n,K--) reactions are given in barns-ev.

The cross section for the specified reaction is given in barns, followed by the error in percent.

parent reaction daughter cross section    +-error      parent reaction daughter cross section    +-error
H  1 (n,Dtot )  1.73063E+02+- 0.00000E+00  H  1 (n,Ddiss)  1.73063E+02+- 0.00000E+00
H  1 (n,Ktot )  7.32927E+05+- 0.00000E+00  H  1 (n,Kel )   3.84884E-01+- 0.00000E+00
H  1 (n,Knone)  7.32926E+05+- 0.00000E+00  H  1 (n,Kphot)  7.32493E+05+- 0.00000E+00
H  1 (n,Kktot)  7.32927E+05+- 0.00000E+00  H  1 (n,Xd )   3.29462E-01+- 0.00000E+00
H  1 (n,total)  3.02774E+01+- 0.00000E+00  H  1 (n,E )    H  1 2.99480E+01+- 2.96233E-01
H  1 (n,g )     H  2 3.29417E-01+- 2.55297E+00  H  2 (n,Dtot )  5.18112E-01+- 0.00000E+00
H  2 (n,Ddiss)  5.18112E-01+- 0.00000E+00  H  2 (n,Ktot )  3.14013E+03+- 0.00000E+00
H  2 (n,Kel )   4.82672E-02+- 0.00000E+00  H  2 (n,Knone)  3.14009E+03+- 0.00000E+00
H  2 (n,Kktot)  3.53743E+00+- 0.00000E+00  H  2 (n,Xt )   5.01819E-04+- 0.00000E+00
H  2 (n,total)  4.22333E+00+- 1.50000E+00  H  2 (n,nonel)  5.01748E-04+- 0.00000E+00
H  2 (n,E )     H  2 4.22283E+00+- 2.00000E+00  H  2 (n,g )    H  3 5.01748E-04+- 3.00000E+00
H  3 (n,Ktot )  1.87166E-02+- 0.00000E+00  H  3 (n,Kel )   1.87166E-02+- 0.00000E+00
H  3 (n,Kktot)  1.87166E-02+- 0.00000E+00  H  3 (n,total)  1.93838E+00+- 0.00000E+00
H  3 (n,E )     H  3 1.93838E+00+- 0.00000E+00  H  3 (n,Dtot )  1.05921E+07+- 0.00000E+00
He 3 (n,Ddiss)  1.05921E+07+- 0.00000E+00  He 3 (n,Ktot )  4.02870E+09+- 0.00000E+00
He 3 (n,Kel )   3.72152E-02+- 0.00000E+00  He 3 (n,Knone)  4.02870E+09+- 0.00000E+00
He 3 (n,Kktot)  4.02870E+09+- 0.00000E+00  He 3 (n,Xp )   5.27488E+03+- 0.00000E+00
He 3 (n,Xt )   5.27488E+03+- 0.00000E+00  He 3 (n,Xa )   5.45646E-05+- 0.00000E+00
He 3 (n,total)  5.27801E+03+- 0.00000E+00  He 3 (n,p )    H  3 5.27415E+03+- 0.00000E+00
He 3 (n,E )     He 3 3.85381E+00+- 0.00000E+00  He 3 (n,g )    He 4 5.45572E-05+- 0.00000E+00
He 4 (n,Ktot )  7.12495E-03+- 0.00000E+00  He 4 (n,Kel )   7.12495E-03+- 0.00000E+00
He 4 (n,Kktot)  7.12495E-03+- 0.00000E+00  He 4 (n,total)  8.62086E-01+- 0.00000E+00
He 4 (n,E )     He 4 8.62086E-01+- 1.10339E+00  Li 6 (n,Dtot )  5.61809E+06+- 0.00000E+00
Li 6 (n,Ddiss)  5.61809E+06+- 0.00000E+00  Li 6 (n,Ktot )  4.45511E+09+- 0.00000E+00
Li 6 (n,Kel )   4.91128E-03+- 0.00000E+00  Li 6 (n,Knone)  4.45511E+09+- 0.00000E+00
Li 6 (n,Kktot)  4.45483E+09+- 0.00000E+00  Li 6 (n,Xt )   9.31233E+02+- 0.00000E+00
Li 6 (n,Xa )   9.31233E+02+- 0.00000E+00  Li 6 (n,total)  9.31920E+02+- 0.00000E+00
Li 6 (n,nonel)  9.31143E+02+- 0.00000E+00  Li 6 (n,t )    He 4 9.31105E+02+- 1.35019E-01
Li 6 (n,E )     Li 6 7.76787E-01+- 2.00988E+00  Li 6 (n,q )    Li 7 3.81990E-02+- 8.00000E+00
```

# Collapse and condense

Common practice is to separate out collapse and condense

**collapse.i** input file – GETXS 1 <group>

**condese.i** input file – GETDECAY 1

Computationally expensive to do these processes

If flux is not changing, then no need to redo collapse

We will do this in exercise 3

```
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact collapse
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact condense
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact input1
[tstainer@mac:~/Dev/fispact]$ ./bin/fispact input2
```

# Collapse

The nuclear data collapse with GETXS takes the extensive reaction cross section data and probability tables for resonance self-shielding and folds (collapses) them with irradiation spectrum in fluxes

Produces a binary COLLAPX file of 1-group data with uncertainties

$$\bar{\sigma} = \frac{\sum_i^N \sigma_i \varphi_i}{\sum_i^N \varphi_i}$$

The nuclear data condense via GETDECAY produces a binary  
ARRAYX file from the extensive decay and  
fission yield data

May also require an irradiation spectrum (fluxes)  
to calculate spectrum-dependent fission yields  
but is not normally the case

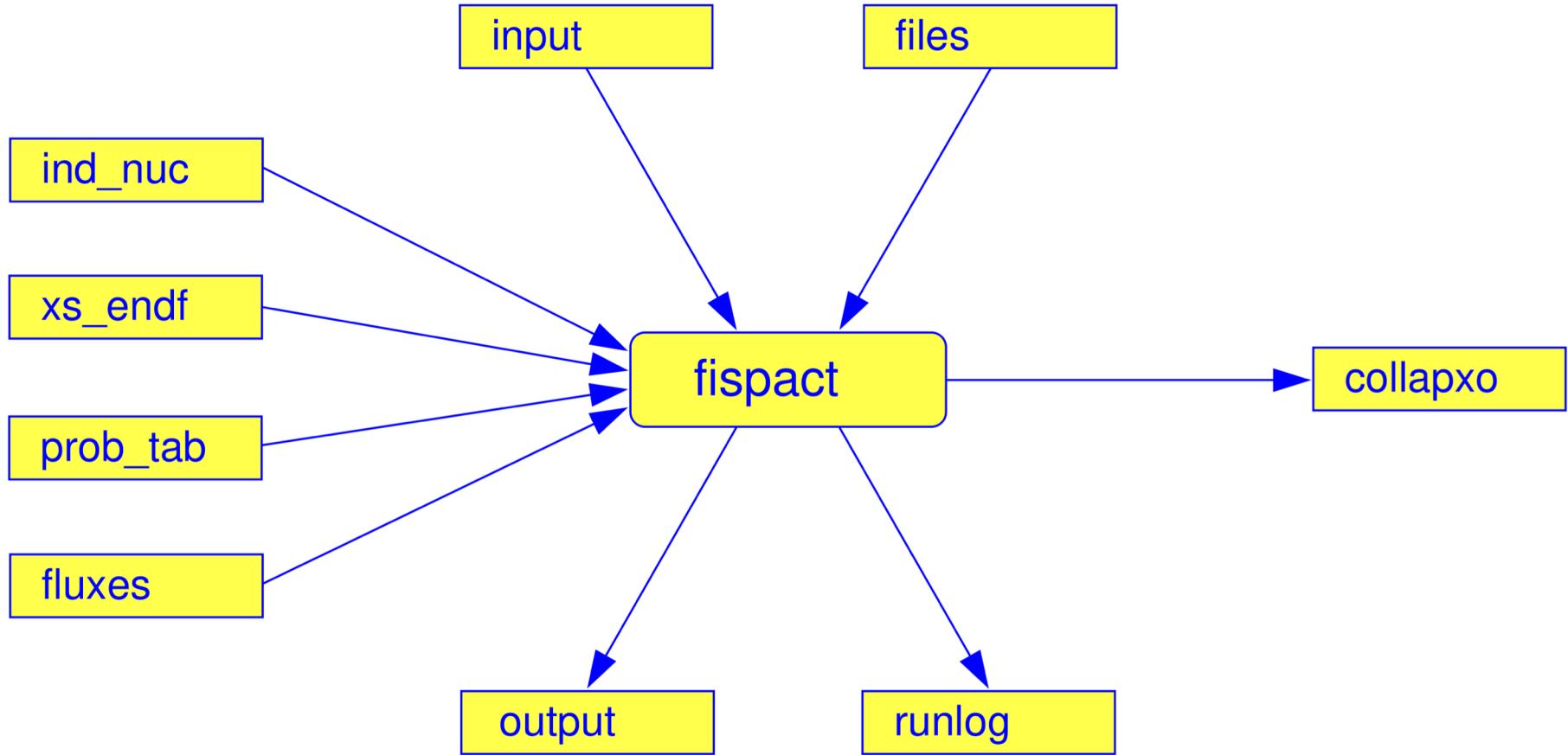
# Intro Summary

We have used the fluxes and nuclear data to calculate all of the effective cross sections and prepare the decay data for time-dependent simulations.

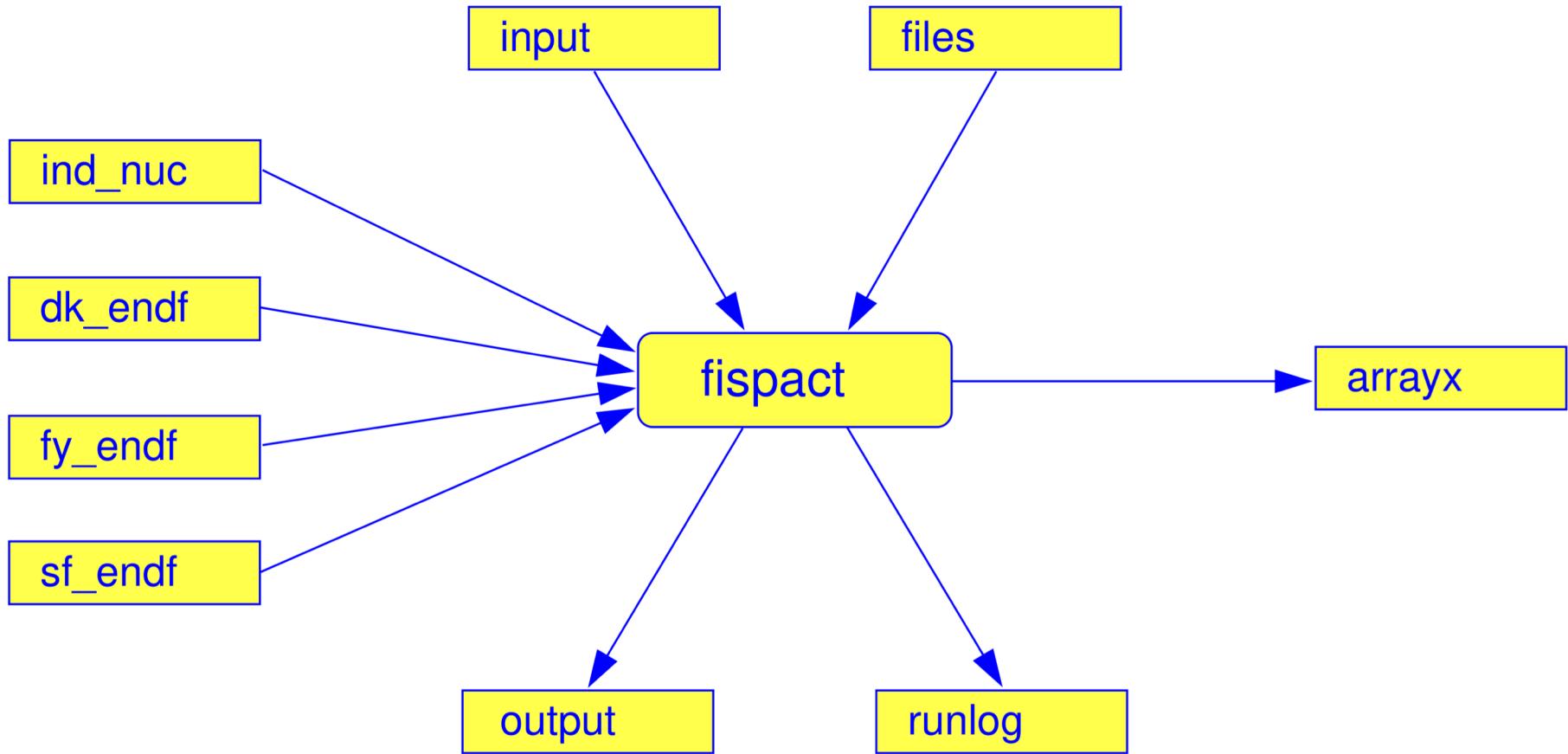
Next we must provide the initial conditions and then simulate some irradiation and cooling phases.

We will then use FISPACT-II to return various quantities such as activity, (spectroscopic) heat, dose rates, emitted spectra and various other quantities.

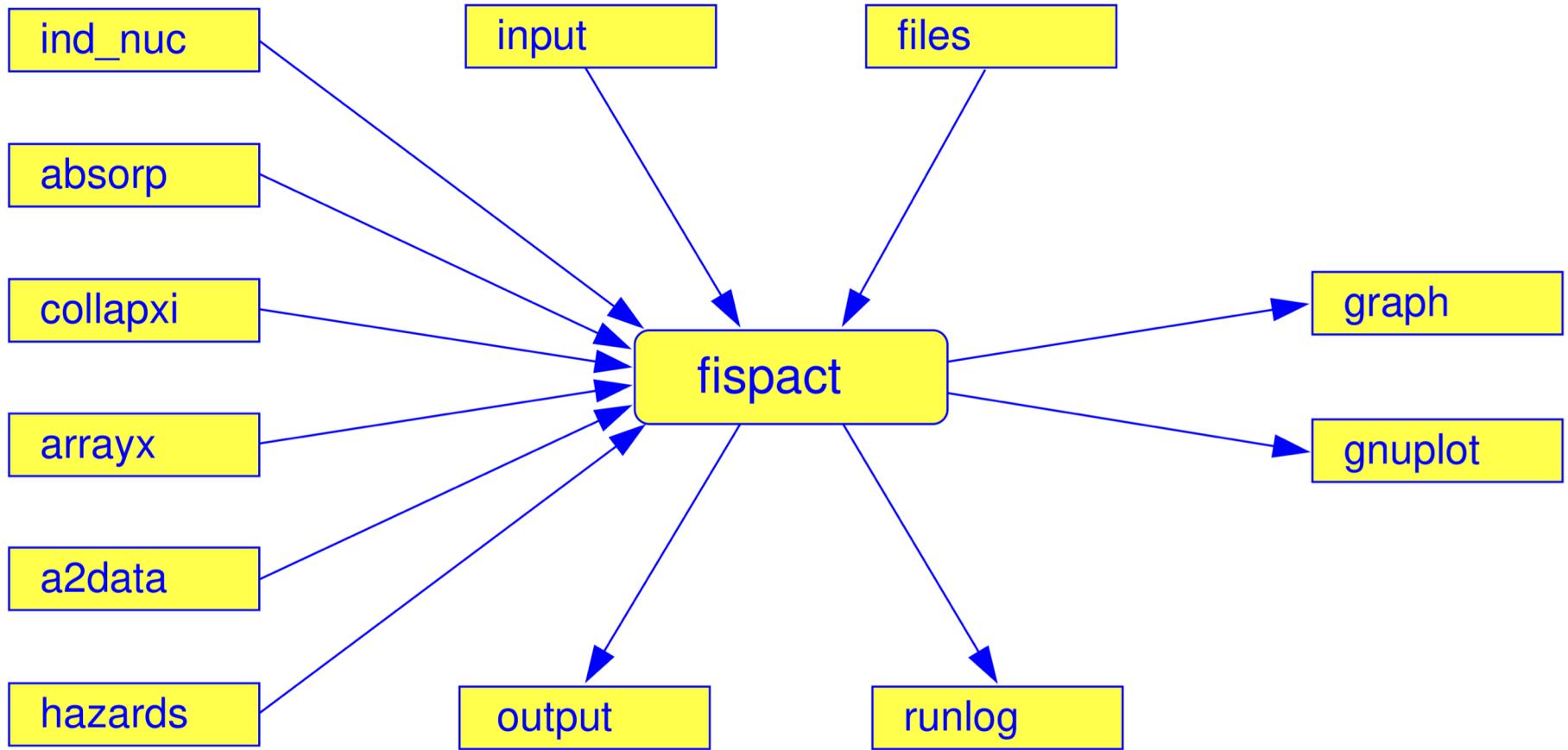
# Collapse Summary



# Condense Summary



# Inventory Summary



# FNS Inconel example

Contains 4 inputs

- Collapse
- Condense
- Printlib
- Inventory

The first 3 are similar to exercise 2

The inventory input is then used to simulate our irradiation and cooling phases

# FNS Inconel example

Read the collapsed and condensed data

```
<< -----set initial switches and get nuclear data----- >>  
CLOBBER  
JSON  
GETXS 0  
GETDECAY 0  
FISPACT  
* FNS 5 Minutes Inconel-600
```

# FNS Inconel example

## Initial conditions

DENSITY 8.42

MASS 1.0E-3 4

NI 75.82

MN 0.39

FE 7.82

CR 15.97

1 gram of **Inconel 600** with density of 8.42 g/cc  
Inconel 600 – 75.82% Ni, 15.97% Cr, 7.82% Fe and 0.39% Mn

# FNS Inconel example

## Initial conditions part 2

```
MIND 1E3
GRAPH 1 2 1 3
UNCERTAINTY 2
HALF
HAZARDS
```

**MIND** – ignore nuclides with inventory < 1000 atoms

**GRAPH** – create gnuplot data and file

**UNCERTAINTY** – pathways analysis

**HALF** – output half lives

**HAZARDS** – output ingestion and inhalation

# FNS Inconel example

## Irradiation phase

```
<< -----irradiation phase----- >>  
FLUX 1.116E+10  
ATOMS  
TIME 5.0 MINS  
ATOMS
```

**FLUX** – specifies the energy integrated flux ( $\text{cm}^{-2}\text{s}^{-1}$ )

**ATOMS** – outputs initial inventory

**TIME** – specify **time interval**, not elapsed time

**ATOMS** – output inventory for time interval specified (5 mins)

# FNS Inconel example

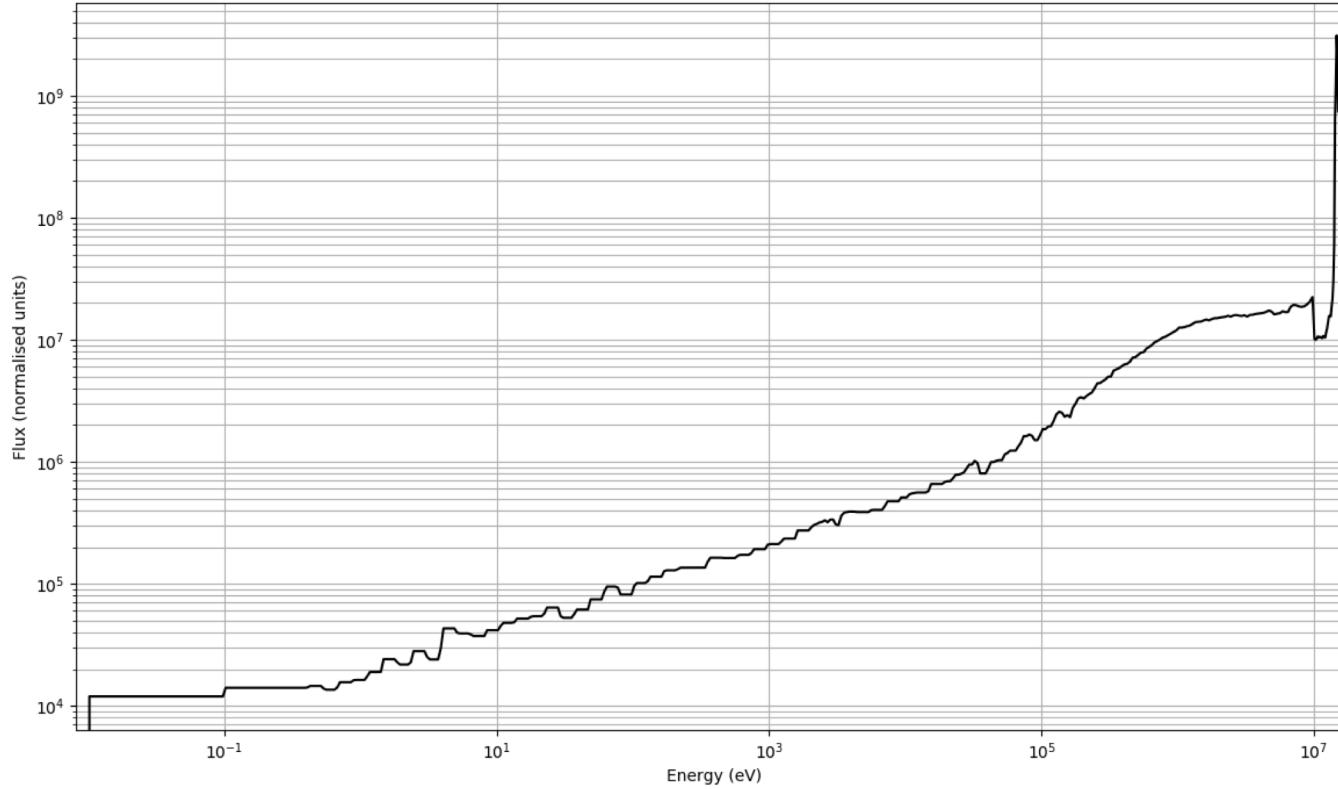
## Cooling phase

```
<< -----cooling phase----- >>
FLUX 0.
ZERO
TIME 36 ATOMS
TIME 15 ATOMS
TIME 16 ATOMS
...
END
* END
```

**FLUX** – sets the flux to zero (cooling)  
**ZERO** – reset time to zero, start of cooling  
**TIME** – output the inventory at time intervals  
**END** – finish simulation

# FNS Inconel example

Neutron Flux

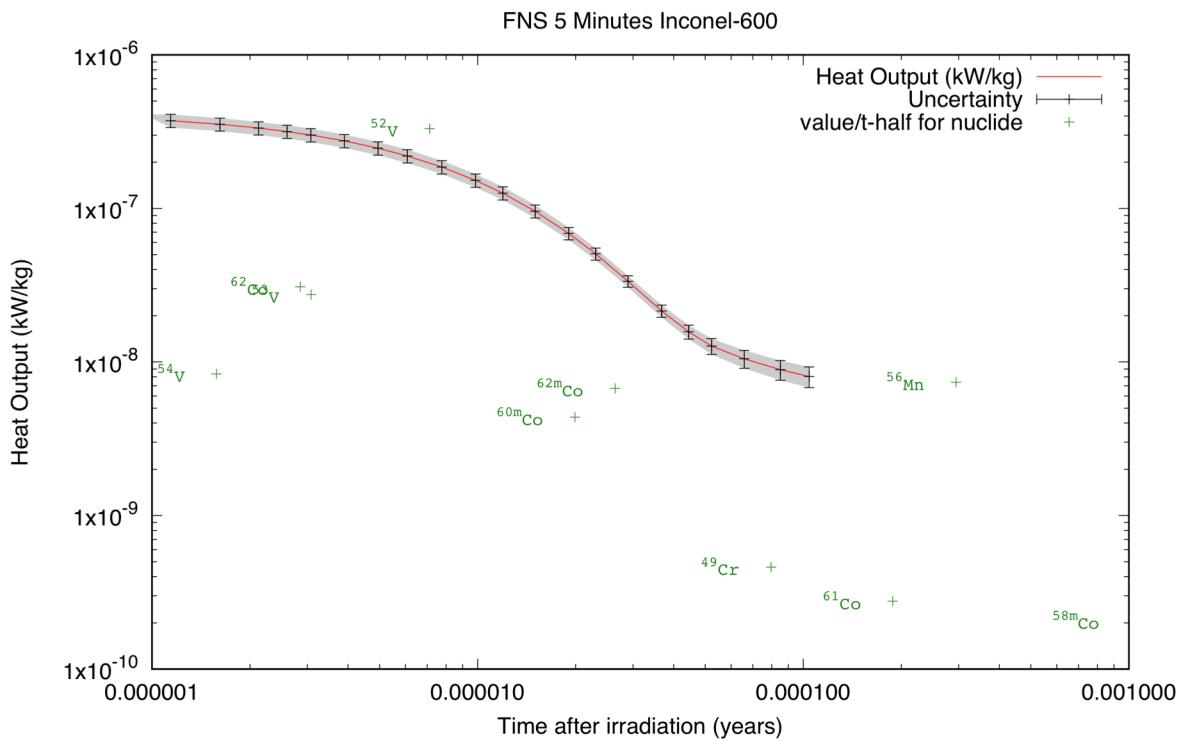


# Exercise 3

Running FNS Inconel 600

# Exercise 3

## Heat Output



file name = inventory.gra run timestamp = 10:18:37 6 June 2018

# Summary

We should now know

- How to construct an files file and input file
- How to run fispact
- How to get some basic outputs

Advanced fispact, up next

- Uncertainties
- JSON
- Pypact + other tools
- Running in Docker