Standard output, common errors and mistakes

Exploring the standard output | Handling errors | Common mistakes

Introduction

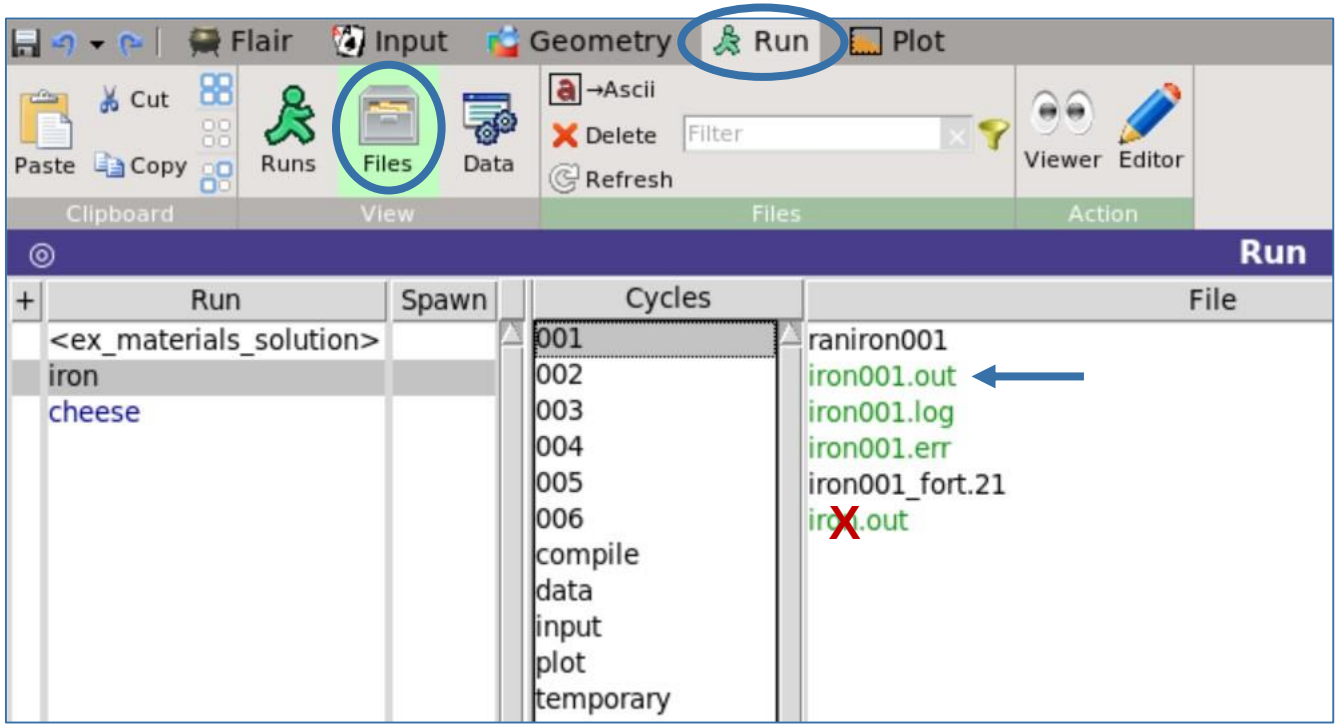
- We have already encountered most of the basic functions of FLUKA and introduced Flair as the main interface for setting up FLUKA simulations (building the geometry, defining materials and beams etc.)
- In the background, the code generates various output files that provide useful information
- The first objective of this lecture is to present the information that can be found in these files, focusing on the standard output file (.out) and, more briefly, on the .err and .log files
- In the second part, we shall present a non-exhaustive list of possible errors and, most importantly, how to use the available information in order to identify their cause

The standard output file

and the *.err* and *.log* files

The standard output file (.out)

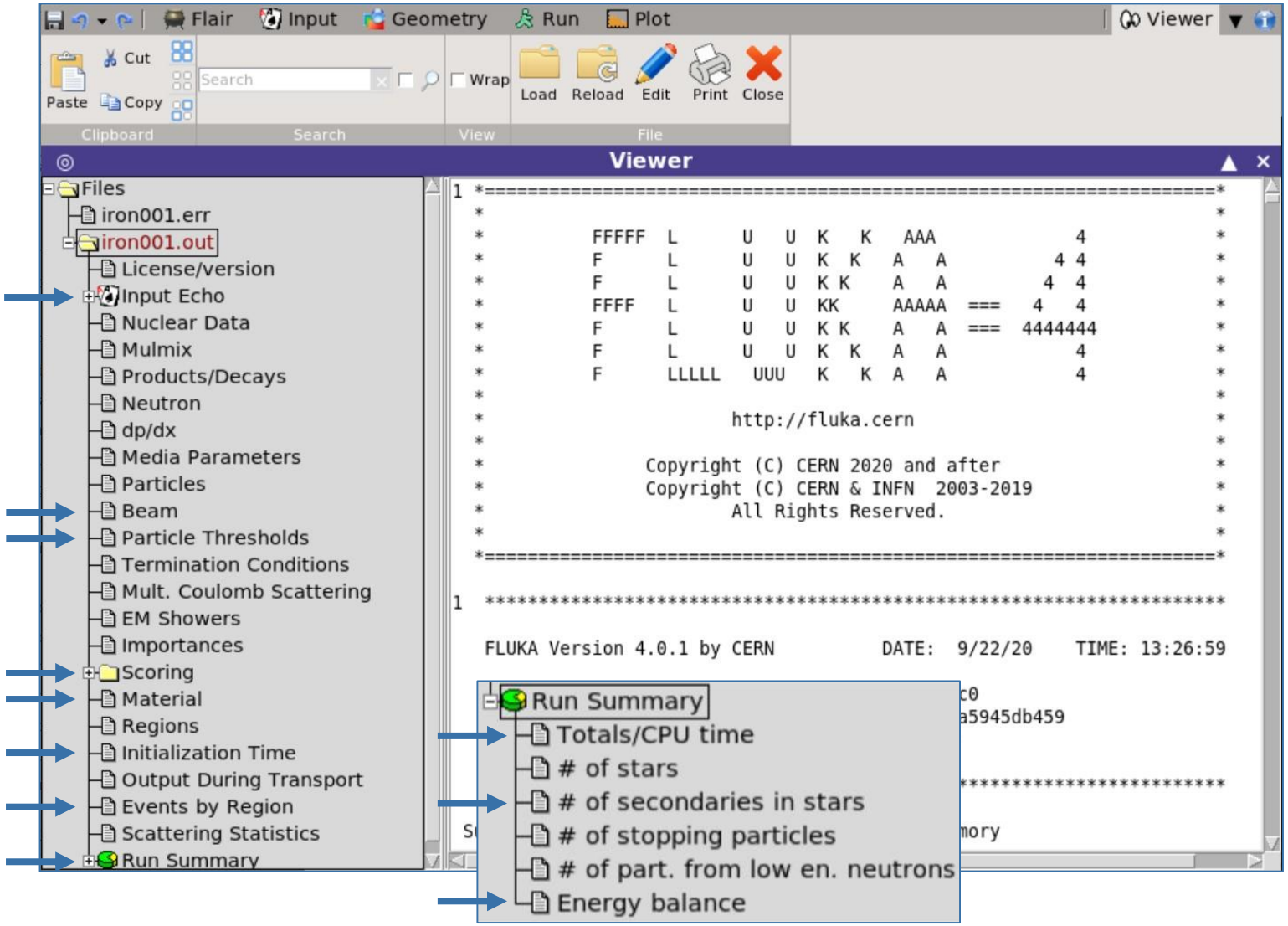
- The standard output file (and other generated files) can be listed and viewed inside Flair via the Files menu under the Run tab
- It may be found in the working directory (i.e. the directory in which you launched the run) or in the temporary `fluka_*` subdirectory if the run is still ongoing or if it ended with errors
- Name: `<inputname>###.out`, e.g. `myrun001.out`



- One output file is produced per cycle (`*001.out`, `*002.out` etc.)
- **Note:** Not to be confused with the `<inputname>.out` file

The standard output file (.out)

- The .out file can be quite large
- The Flair viewer organises it into sections for easier navigation
- It contains a trove of useful information concerning the simulation
- Extremely useful for investigating strange results and errors
- Let's look at some of the contents!



The standard output file (.out)

- **Beam information**
  - Particle type, energy, source position and spatial profile
  - (as long as you are not generating arbitrary distributions via source routine)

```
=== Output before the actual run - Beam properties ===


Fluka incident beam properties:


  Beam particle: PROTON   Id:   1 (Fluka)  2212 (PDG) Charge:   1 Baryon n.:   1
                 Mass: 0.9383     (GeV/c^2) Mean life: 1.0000E+18 (s) Weight:  1.000
  Average beam momentum      :                    0.644445 (GeV/c)
  Average beam kinetic energy:                    0.200000 (GeV)
  Momentum deviation at FWHM (rectangular):         0.0000000 (GeV/c)
  Beam hit position     :   0.00000000       0.00000000      -5.00000000      cm
  Beam direction cosines:   0.00000000       0.00000000       1.00000000
  Beam spot FWHM X-width (Rectangular ):  0.0000      cm
  Beam spot FWHM Y-width (Rectangular ):  0.0000      cm
  Beam FWHM angular divergence (Rectangular ):            0.0000  (mrad)
  (Spatial distribution, polarization, and angular direction and distribution
  are given in the beam frame of reference)

  Beam reference frame (world coordinates):
    Beam X axis:    1.00000000       0.00000000       0.00000000
    Beam Y axis:    0.00000000       1.00000000       0.00000000
    Beam Z axis:    0.00000000       0.00000000       1.00000000

  The nominal beam position belongs to region:       2 (VOID    ),
  lattice cell:       0 (        )
```

The standard output file (.out)

- **Summary of transport thresholds**

```
=== Particle transport thresholds:

Global cut-off kinetic energy for particle transport:  1.000E-04 GeV
The cut-off kinetic energy is superseded by individual particle thresholds if set


  Cut-off kinetic energy for 4-HELIUM transport:  1.000E-04 GeV

  Cut-off kinetic energy for 3-HELIUM transport:  1.000E-04 GeV

  Cut-off kinetic energy for TRITON   transport:  1.000E-04 GeV

  Cut-off kinetic energy for DEUTERON transport:  1.000E-04 GeV

  Cut-off kinetic energy for PROTON   transport:  1.000E-04 GeV

  Cut-off kinetic energy for APROTON  transport:  1.000E-04 GeV

  Cut-off kinetic energy for ELECTRON transport defined in the Emfcut card

  Cut-off kinetic energy for POSITRON transport defined in the Emfcut card

  Cut-off kinetic energy for NEUTRIE  transport:  0.000E+00 GeV

  Cut-off kinetic energy for ANEUTRIE transport:  0.000E+00 GeV

  Cut-off kinetic energy for PHOTON   transport defined in the Emfcut card

  Cut-off kinetic energy for NEUTRON  transport:  1.000E-14 GeV
```

100 keV (**PRECISION** on **DEFAULTS**)

```
Correspondence of regions and EMF-FLUKA material numbers
     Region          EMF                        FLUKA

        1 BLKBODY          0   VACUUM                1  BLCKHOLE
        2 VOID             0   VACUUM                2  VACUUM
        3 TARGET           1   IRON                 11  IRON
Prompt:
   Ecut =  6.1100E-01 MeV,    Pcut =  3.3333E-02 MeV
   BIAS = F,   Ray. = T,   S(q,Z) = T,  Pz(q,Z) = T,  Acolin.= F
Delayed:
   Ecut =  6.1100E-01 MeV,    Pcut =  3.3333E-02 MeV
```

In the absence of an **EMFCUT** card explicitly setting them, **electron/positron** and **γ** transport thresholds are calculated internally

The standard output file (.out)

- **Materials and material properties**
  - Includes FLUKA predefined materials and user-defined materials
  - Basic material properties
  - (In)elastic scattering length, radiation length for selected **BEAM** particle/energy

```
 === Material compositions: ===

Material        Atomic          Atomic          Density         Inelastic       Elastic         Radiation
Number&Name     Number          Weight                          Scattering      Scattering      Length
                                                                Length for      Length for
                                                                PROTON   at     PROTON   at
                                                                Beam energy     Beam energy

                                                  g/cm**3          cm              cm              cm

  1 BLCKHOLE      0.000           0.000           0.000         0.1000E+31      0.1000E+31      0.1000E+31
  2 VACUUM        0.000           0.000           0.000         0.1000E+31      0.1000E+31      0.1000E+31
  3 HYDROGEN      1.000           1.008         0.8370E-04      0.3641E+10      0.8848E+06      0.7532E+06

 30 GRUYERE       3.294           5.975           0.9153          115.2           219.2           41.98

    Material        Number          Atom content    Partial Densities

    HYDROGEN         3              0.62121         0.95916E-01
    OXYGEN           8              0.17083         0.41870
    CARBON           6              0.18469         0.33981
    NITROGEN         7              0.19411E-01     0.41649E-01
    CALCIUM         21              0.15141E-02     0.92953E-02
    PHOSPHO         28              0.11754E-02     0.55772E-02
    SODIUM          19              0.95020E-03     0.33463E-02
    POTASSIU        29              0.11485E-03     0.68786E-03
    MAGNESIU         9              0.99864E-04     0.37181E-03

 31 WATER         3.333           6.005           1.000           109.5           194.4           36.08

    Material        Number          Atom content    Partial Densities

    HYDROGEN         3              0.66667         0.11190
    OXYGEN           8              0.33333         0.88810
```

The standard output file (.out)

- **Summary of requested scoring**
  - Interpreted scoring cards

```
******* "usrbin" option:

  Cartesian binning n.   1  "Protons    " , generalized particle n.    1
    X coordinate: from -1.0000E+01 to  1.0000E+01 cm,   201 bins ( 9.9502E-02 cm wide)
    Y coordinate: from -1.0000E+01 to  1.0000E+01 cm,   201 bins ( 9.9502E-02 cm wide)
    Z coordinate: from -5.0000E+00 to  1.5000E+01 cm,   200 bins ( 1.0000E-01 cm wide)
    data will be printed on unit  -21 (unformatted if < 0)
    accurate deposition along the tracks requested
    normalized (per unit volume) data will be printed at the end of the run
    this is a track-length binning


******* "USRBDX" option:


No user bdrx defined


******* "USRTRACK" option:


No user track-length estimator defined
```

Proton

The standard output file (.out)

- **Input echo**
  - Interpreted body and region definitions
  - Very useful in case of complex inputs controlled by many different preprocessor directives, where the options may not be activated as intended due to a mistake
  - The input echo will only contain the parts which were really active in the run

- **Initialisation time**
  - Run initialisation happens at the beginning of each cycle
  - It is generally quite fast (especially relative to the total duration of the run) but can reach up to several minutes for complex geometries, LATTICEs, scoring, preprocessor directives etc.

```
Total time used for initialization:   0.539      s
```

Basic input template

```
Total time used for initialization:   82.1       s
```

Complex LHC simulation, 34000 line input file

The standard output file (.out)

- **Run summary**
  - Energy balance

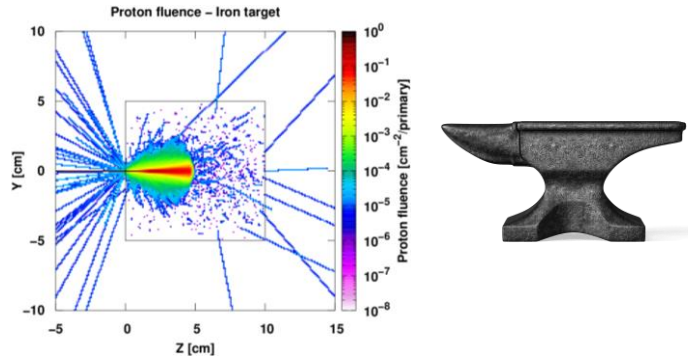
```
  2.0000E-01  (100.%) GeV available per beam particle divided into
Prompt radiation      Radioactive decays
  1.7460E-01  (87.3%)   0.0000E+00 ( 0.0%) GeV hadron and muon dE/dx
  9.8263E-03  ( 4.9%)   0.0000E+00 ( 0.0%) GeV electro-magnetic showers
  4.4650E-04  ( 0.2%)   0.0000E+00 ( 0.0%) GeV nuclear recoils and heavy fragments
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles below threshold
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV residual excitation energy
  8.4339E-05  ( 0.0%)   0.0000E+00 ( 0.0%) GeV low energy neutrons
  6.1338E-03  ( 3.1%)   0.0000E+00 ( 0.0%) GeV particles escaping the system
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles discarded
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles out of time limit
  8.9106E-03  ( 4.5%)                       GeV missing

  2.0000E-01  (100.%) GeV available per beam particle divided into
Prompt radiation      Radioactive decays
  4.5129E-02  (22.6%)   0.0000E+00 ( 0.0%) GeV hadron and muon dE/dx
  3.1495E-03  ( 1.6%)   0.0000E+00 ( 0.0%) GeV electro-magnetic showers
  2.6012E-04  ( 0.1%)   0.0000E+00 ( 0.0%) GeV nuclear recoils and heavy fragments
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles below threshold
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV residual excitation energy
  5.9178E-06  ( 0.0%)   0.0000E+00 ( 0.0%) GeV low energy neutrons
  1.4916E-01  (74.6%)   0.0000E+00 ( 0.0%) GeV particles escaping the system
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles discarded
  0.0000E+00  ( 0.0%)   0.0000E+00 ( 0.0%) GeV particles out of time limit
  2.2925E-03  ( 1.1%)                       GeV missing
```

Hadronic / EM break-down

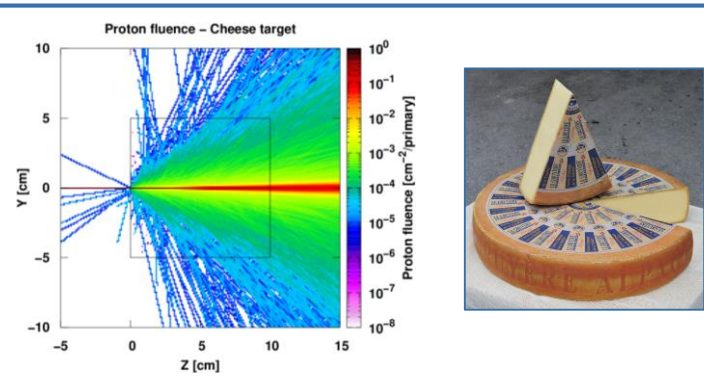
Exiting the geometry

Nuclear reactions, particle production

Recall the materials exercise:
200 MeV protons on iron & cheese



More reactions, less escaping energy

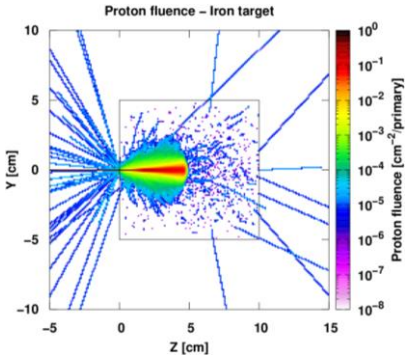


Fewer reactions, more escaping energy

The standard output file (.out)

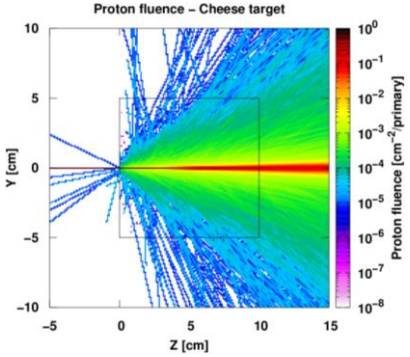
- **Information per region**
  - Energy deposition

1Region #	name	volume in cubic cm	ALL-PART Star Density Stars/cm**3 /one beam particle	BEAMPART Star Density Stars/cm**3 /one beam particle	ENERGY Density GeV/cm**3 /one beam particle	EM-ENRGY Density GeV/cm**3 /one beam particle
1	BLKBODY	1.000000000D+00	0.000000000D+00	0.000000000D+00	6.133758472D-03	7.633542712D-04
2	VOID	1.000000000D+00	0.000000000D+00	0.000000000D+00	0.000000000D+00	0.000000000D+00
3	TARGET	1.000000000D+00	3.153000000D-01	2.530000000D-01	1.849556851D-01	9.826343461D-03
Total (integrated over volume):			3.153000000D-01	2.530000000D-01	1.910894436D-01	1.058969773D-02



≈ 185.0 MeV/pr.

1Region #	name	volume in cubic cm	ALL-PART Star Density Stars/cm**3 /one beam particle	BEAMPART Star Density Stars/cm**3 /one beam particle	ENERGY Density GeV/cm**3 /one beam particle	EM-ENRGY Density GeV/cm**3 /one beam particle
1	BLKBODY	1.000000000D+00	0.000000000D+00	0.000000000D+00	1.491629305D-01	1.177755188D-04
2	VOID	1.000000000D+00	0.000000000D+00	0.000000000D+00	0.000000000D+00	0.000000000D+00
3	TARGET	1.000000000D+00	9.630000000D-02	8.840000000D-02	4.854454810D-02	3.149463521D-03
Total (integrated over volume):			9.630000000D-02	8.840000000D-02	1.977074786D-01	3.267239040D-03



≈ 48.5 MeV/pr.

The standard output file (.out)

- **Run summary**
  - Primary particles, number of interactions, CPU-time

```
Total number of primaries run:          10000 for a weight of: 1.000000E+04
!!! Please remember that all results are normalized  per unit weight !!!
The main stack maximum occupancy was         12 out of      70000 available


Total number of inelastic interactions (stars):              963
Total weight of the inelastic interactions (stars):  9.630000E+02


Total number of elastic interactions:                649
Total weight of the elastic interactions:  6.490000E+02


Total number of low energy neutron interactions:               1047
Total weight of the low energy neutron interactions: 1.047000E+03

Total CPU time used to follow all primary particles:   6.594E+00 seconds of:
Average CPU time used to follow a primary particle:    6.594E-04 seconds of:
Maximum CPU time used to follow a primary particle:    5.707E-03 seconds of:
Residual CPU time left:                                1.000E+30 seconds of:
```

From fractions of a millisecond to tens of minutes or more

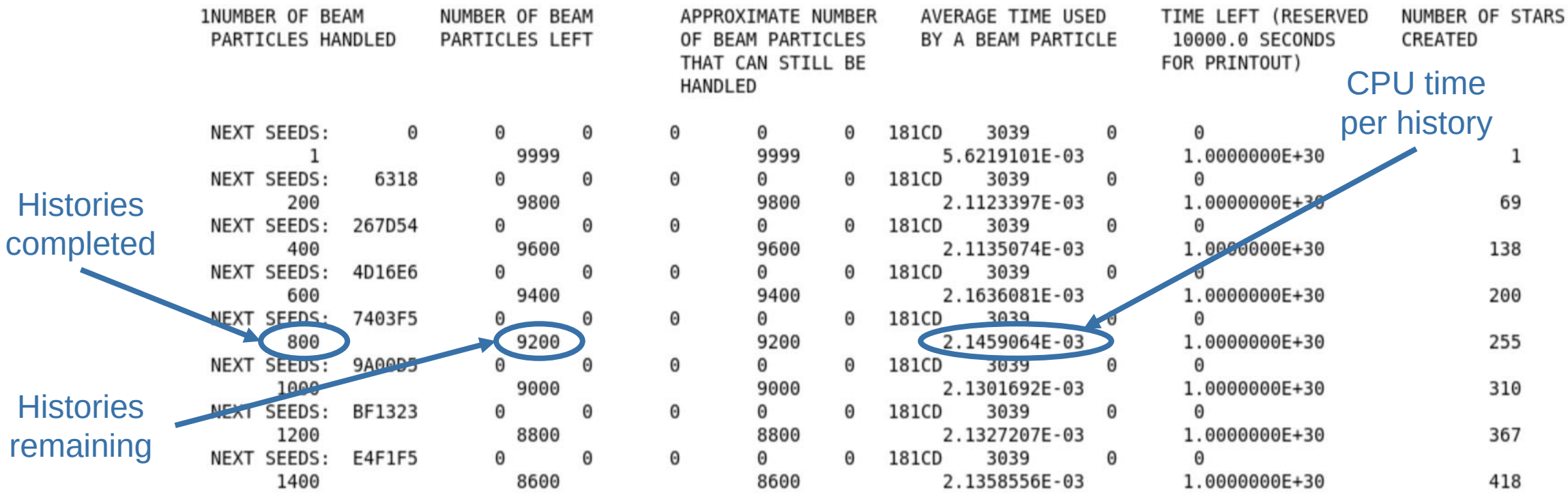
The standard output file (.out)

- **Run summary**
  - Number and type of produced secondary particles

```
Number of secondaries generated in inelastic interactions per beam particle:
Prompt radiation      Radioactive decays
  1.9023E+00 (100.%)   0.0000E+00 (100.%)
  6.4300E-02 ( 3.4%)   0.0000E+00 ( 0.0%) 4-HELIUM
  1.4000E-03 ( 0.1%)   0.0000E+00 ( 0.0%) 3-HELIUM
  2.4000E-03 ( 0.1%)   0.0000E+00 ( 0.0%) TRITON
  1.3900E-02 ( 0.7%)   0.0000E+00 ( 0.0%) DEUTERON
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) HEAVYION
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) OPTIPHOT
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) RAY
  5.1690E-01 (27.2%)   0.0000E+00 ( 0.0%) PROTON
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) APROTON
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) ELECTRON
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) POSITRON
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) NEUTRIE
  0.0000E+00 ( 0.0%)   0.0000E+00 ( 0.0%) ANEUTRIE
  7.0840E-01 (37.2%)   0.0000E+00 ( 0.0%) PHOTON
  5.9500E-01 (31.3%)   0.0000E+00 ( 0.0%) NEUTRON
```

The .err file

- **Note:** The .err file is generated even for successful runs, its presence does not necessarily imply than any errors were encountered
- It contains information concerning the progress of the run
- Run-time error messages and warnings will also be written to this file



The .log file

- Critical errors, usually following a code abort will be written to the .log file, as in the following examples

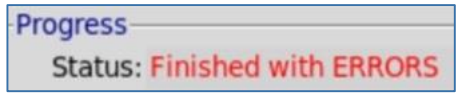
```
Program received signal SIGFPE: Floating-point exception - erroneous arithmetic operation.

Backtrace for this error:
#0  0x7f60e3aae3ff in ???
#1  0x7f60e3ab84fe in ???
#2  0x7f60e47b13a8 in read_real
        at /GCC/build/contrib/gcc-8.2.0/src/gcc/8.2.0/libgfortran/io/list_read.c:1872
#3  0x7f60e47b2fcd in list_formatted_read_scalar
        at /GCC/build/contrib/gcc-8.2.0/src/gcc/8.2.0/libgfortran/io/list_read.c:2180
#4  0x6b1859 in flkcgi_
        at comlat/flkcgi.f:556
#5  0x56d96e in geoinp_
        at geolat/geoinp.f:241
#6  0x40388e in flukam_
        at main/flukam.f:2442
#7  0x402100 in fluka
        at main/fluka.f:77
#8  0x402100 in main
        at /shared/src/usflmd.inc:15
```

```
 STOP TOO MANY ERRORS IN GEOMETRY
Note: The following floating-point exceptions are signalling: IEEE_UNDERFLOW_FLAG IEEE_DENORMAL
STOP STOP: FLUKA ABORTED
```

User-defined messages

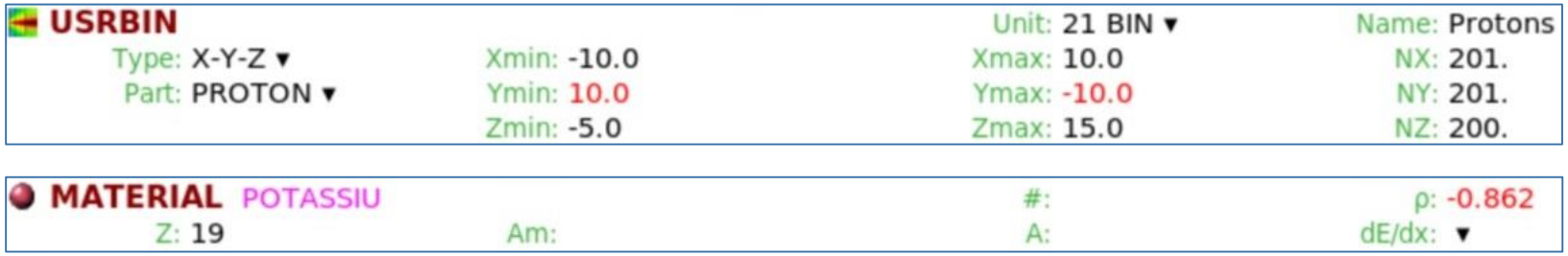
- Users may print information, warning or error messages to the .out and .err files from within their user routines, e.g.:
  - `WRITE ( LUNOUT,* ) 'This is a message'`
  - `WRITE ( LUNERR,* ) 'This is a message'`
- This can be useful when debugging user routines

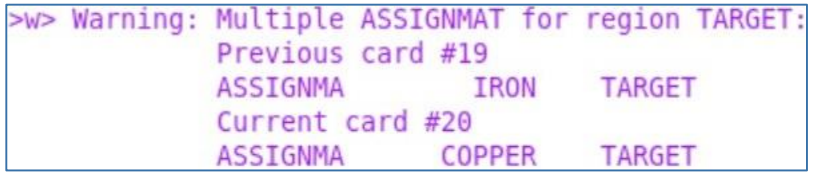
Error handling and common mistakes



Before running

- Have the latest FLUKA and Flair versions correctly installed, downloading the appropriate version for your system and following the instructions
- Try running a provided example to confirm that there are no installation errors
- Do not run inside the FLUKA directory!
- Avoid using file and directory names with spaces
- Flair will already detect and highlight many mistakes, such as nonsense inputs in certain fields, multiple **`ASSIGNMAT`** cards for a region etc.:





- Build the geometry carefully, use the Geometry Editor to search for geometry errors

Click “Start”…

- Runs may fail during initialisation; this usually indicates the existence of a blatantly wrong setting in a card, some missing necessary setting, a mistyped expression or value, a missing file, `#if` directive without corresponding `#endif`, …
- Crashes occurring once particle tracking has started are often attributable to geometry errors, numerical exceptions (often associated with user routines), …

Crashes at initialisation

- **Low-energy neutron cross-sections not found**
  - As mentioned in the Materials lecture, low-energy neutron cross sections are associated to materials based on their names
  - If a user-defined elemental material is named differently **(1)** than in the relevant library **(2)** (see section 10.4.1.2 of the FLUKA manual), then FLUKA will not be able to assign any neutron cross-sections

(1)

(2)

  - This leads to a stop of the execution with the following message (found in the .out file)

```
 **** Low energy neutron xsec not found for some media
F
```

Crashes at initialisation

- **Missing random number file**
  - This error usually indicates that the previous cycle did not end successfully and therefore the random number file **`ran<inputname>###`** was not generated
  - The relevant error messages are found in the .out, .err and .log files:

```
 **** No Random file available !!!!!! ****
Abort called from FLRM64 reason NO RANDOM FILE Run stopped!
STOP NO RANDOM FILE
```

`<inputname>###.out`

```
 STOP NO RANDOM FILE
STOP STOP: FLUKA ABORTED
```

`<inputname>###.log`

```
Abort called from FLRM64 reason NO RANDOM FILE Run stopped!
STOP NO RANDOM FILE
```

`<inputname>###.err`

  - When a run fails, it is generally advisable to look at the output files of the first cycle, because all further cycles will fail with this error after a failed first cycle

Crashes at initialisation

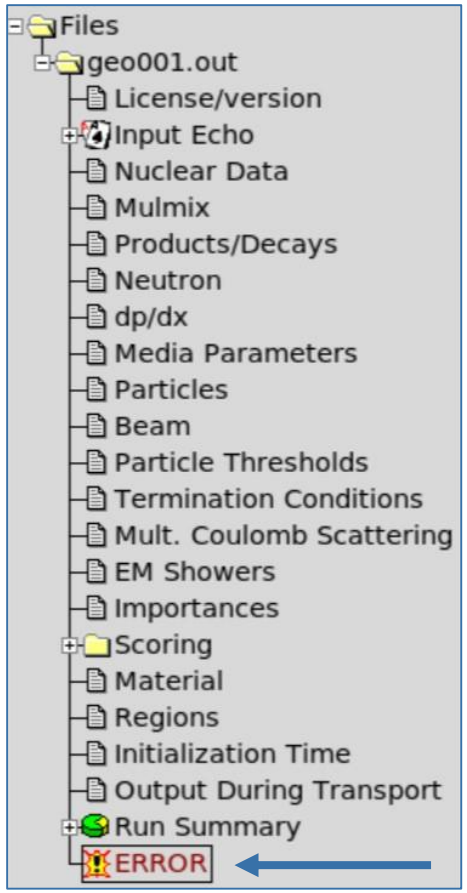
- **Missing executable or external file**
  - If a custom executable or external auxiliary file (e.g. to be used for sampling by a source routine) is not found in the working directory, the code will stop
  - An example of the error message in the **`<inputname>.out`** file:

```
*-* Running: errors
Dir: /errors
Cmd: /usr/bin/nohup /soft/fluka4-0.1/bin/rfluka -e /errors/myexe -M 5 errors
/usr/bin/nohup: ignoring input

                                F L U K A

Dir:   /soft/fluka4-0.1
Data:  /soft/fluka4-0.1/data
Exec:  /errors/myexe
Input: /errors/errors.inp
Error: /errors/myexe does not exist or it is not executable!
```

- **Input file formatting errors**
  - Syntax/alignment errors in manually edited input files
  - Presence of hidden DOS characters in files edited in Windows (clean with **`dos2unix`** command)

Run-time errors

- Where to look?
  - The bottom of the .out and .err files, and the top of the .log file



- When you open the standard output file in Flair, the navigation menu will point to the error message
  - …or, at least, to the bottom of the file; in case of an uncontrolled crash, the final information printed in the standard output file is less likely to be informative
  - Instead, more meaningful information may be found in the .log and .err files

Run-time errors

- **Geometry and tracking accuracy errors**
  - A single geometry error or problem with tracking accuracy (often encountered when using LATTICEs) may not condemn a run, as the code attempts to recover the lost particle
  - These attempts are recorded in the .err file (“geofar” errors)

```
NEXT SEEDS:  F0A31F       0       0       0       0       0   181CD    3039       0       0
      3900                  1100                    1100             2.2694074E-04         1.0000000E+30                23
NEXT SEEDS:  F25D77       0       0       0       0       0   181CD    3039       0       0
Geofar: Particle in region     3 (cell #    0) in position  9.208454725E-01  1.554251376E+00  1.303608751E+00
 is now causing trouble, requesting a step of  5.206776825E+00 cm
 to direction  8.513230560E-02  3.069905258E-01  9.478973086E-01, error count:  0
 R2:  1.806558531E+00 R3:  2.227790273E+00 cm
 X*U   (2D):  5.555341454E-01 X*U   (3D):  1.791221372E+00 cm
 X*UOLD(2D):  8.622110860E-01 X*UOLD(3D):  4.169256137E-01 cm
 Kloop:      32194, Irsave:      3, Irsav2:      3, error code: -33 Nfrom:  5000
 old direction  9.398507183E-01 -2.088589279E-03 -3.415790758E-01, lagain, lstnew, lsense, lsnsct  F  F  F  T
 Particle index    7 total energy   4.547721732E-04 GeV  Nsurf    0
Try again to establish the current region  moving the particle of a  3.887637526E-08 long step
We succeeded in saving the particle:  current region is n.     3 (cell #    0)
```

  - The run will stop if too many errors are encountered and/or the particle cannot be recovered

```
Abort called from FLKAG1 reason TOO MANY ERRORS IN GEOMETRY Run stopped!
STOP TOO MANY ERRORS IN GEOMETRY
```

Run-time errors

- **Geometry errors**
  - These error messages offer valuable information that can help locate the problem

```
Geofar: Particle in region     3 (cell #    0) in position  9.208454725E-01  1.554251376E+00  1.303608751E+00
 is now causing trouble, requesting a step of  5.206776825E+00 cm
 to direction  8.513230560E-02  3.069905258E-01  9.478973086E-01, error count:  0

Particle index     7 total energy   4.547721732E-04 GeV  Nsurf    0
```

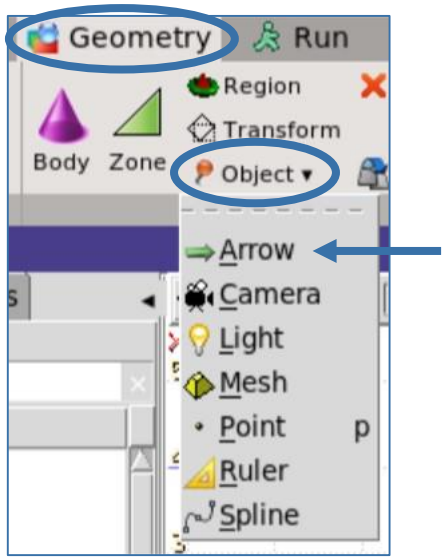
Particle position

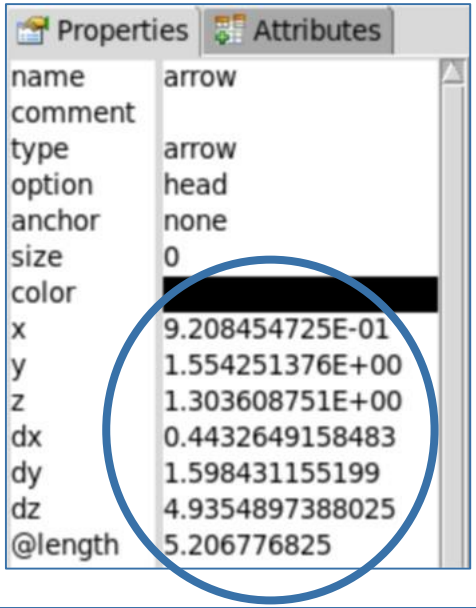
Step length

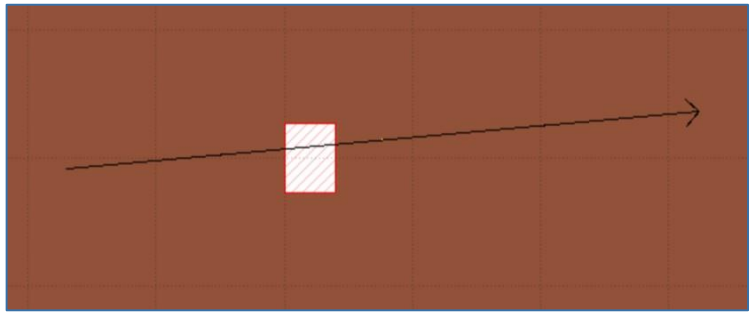
Particle type (Section 5.1 of FLUKA manual)

Particle direction

  - In Flair, you can add a vector (“Arrow”) setting its origin, direction and length according to the error message to help in visualising the geometry problem







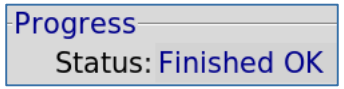
Advanced debugging

- It can get worse: uncontrolled code aborts may lead to the creation of a core dump in the temporary FLUKA directory (**`fluka_*`**), with an error like the following to be found in the **`<inputname>.out`** file

```
======================== Running FLUKA for cycle # 1 ========================
/soft/fluka4-0.1/bin/rfluka: line 368:  1201 Floating point exception(core dumped) "${EXE}" "${INPF}" > "${LOGF}" 2>&1
```

- Such errors are usually caused by numerical issues, variable type discrepancies, attempts to access non-existent array elements etc., and are often associated with the use of user routines
- The **`core.*`** file contains information on the program state at the time of the crash and the relevant back-trace
- It can be accessed via the GNU debugger (gdb)
  - …but this is a story for another course…

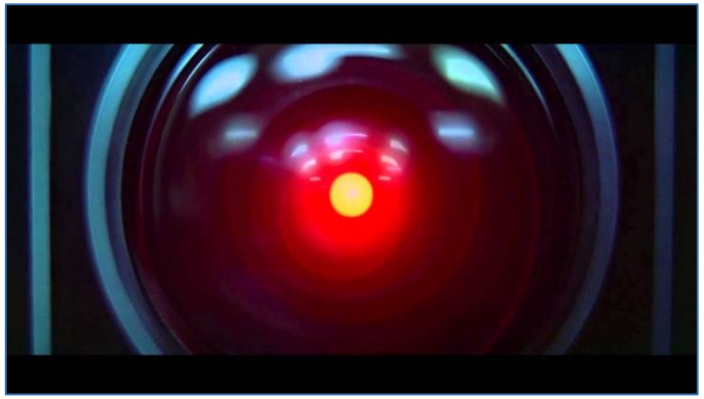
If the run ends without errors…



- Some mistakes will not lead to run-time crashes, but may have an impact on the results of the simulation
  - Overlapping geometry regions will not generate an error; during particle tracking the code will decide in which region the particle is and continue, potentially distorting the scoring of energy deposition, fluence etc.
  - Wrong beam energy assignment (e.g. MeV vs. GeV, energy vs. momentum, total HEAVYION energy in **BEAM** card instead of energy per nucleon, …)
  - Wrong material densities
  - Wrong/multiple material assignments for a region
  - Inappropriate thresholds
  - A missing **RANDOMIZ** card will lead to zero statistical errors, as identical cycles are executed
- Such errors can be discovered upon inspection of the results which may completely deviate from expectations, display strange discontinuities etc.
- You should always critically assess your results based on your knowledge of the problem!

To conclude…

- Errors will always crop up, even for experienced users; do not despair!



“It can only be attributable to human error.”
-- HAL 9000

- Most of the time, the cause will be fairly mundane, and a hint on its origin will be found in the various error messages and output files
- If you are unable to discover the cause of the problem using the suggestions in this lecture, it’s time to explore the FLUKA forum for similar issues and, finally, to post your question on the FLUKA forum
- Remember that meaningful debugging assistance generally requires that you at least provide your .flair file, and other relevant files, user routines etc.


FLUKA