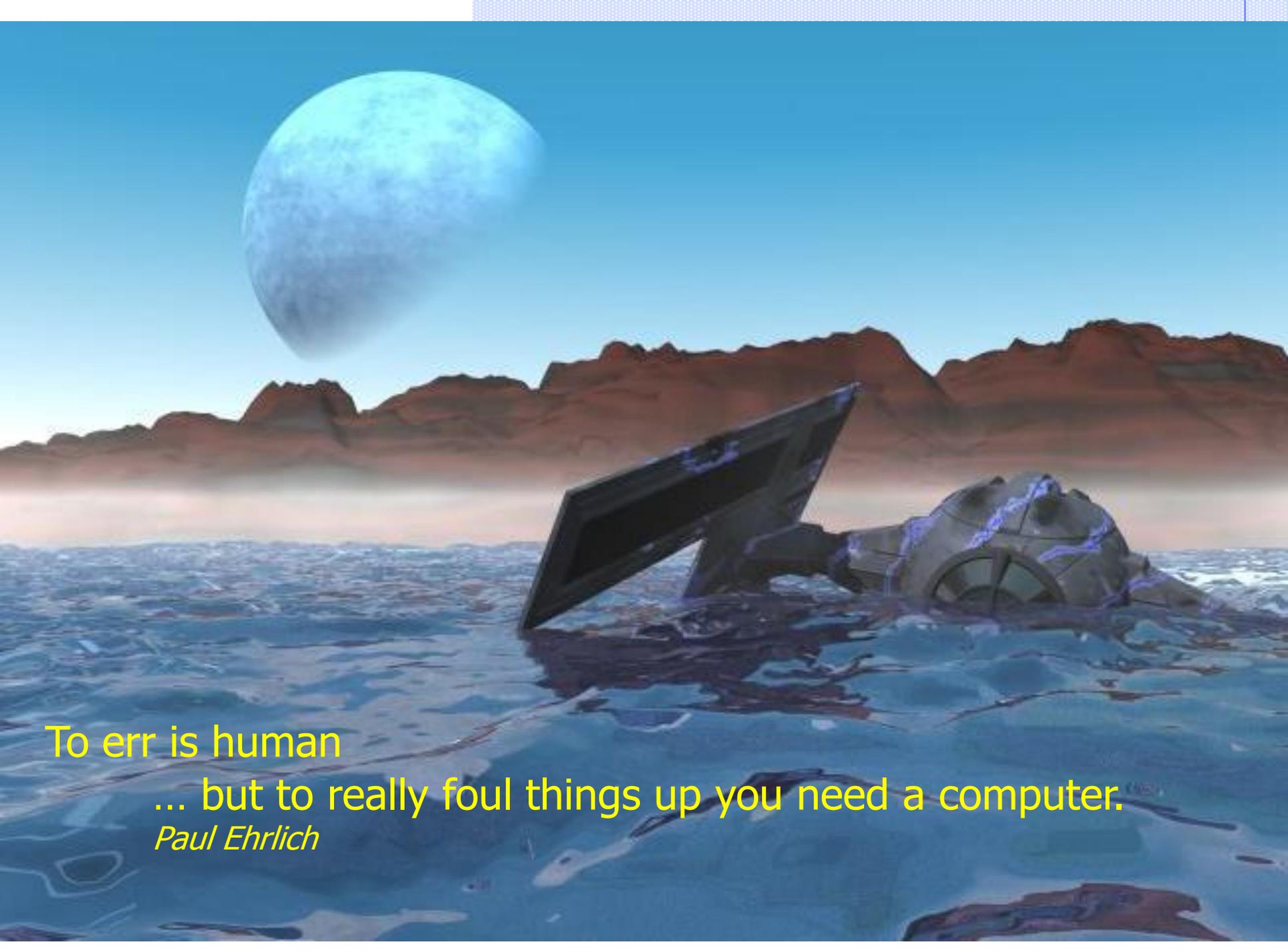




# Handling of errors and crashes

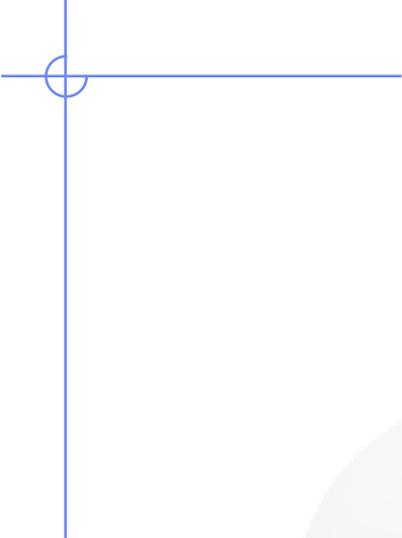
Beginners' FLUKA Course



To err is human

... but to really foul things up you need a computer.

*Paul Ehrlich*



**DON'T PANIC**

# Overview

Sometimes ( or many-times) the users get crashes..

Sometimes results look weird..

- many of times these are simple problems that the user can locate himself: the first purpose of this lecture is to help you how to spot an error which depends on the user.
- In a few cases a real FLUKA problem is found. To locate this kind of error is slightly more complicated, but the second purpose of this lecture is exactly concerning this.
- In the other cases, the debate is on the meaningfulness of results, and on the best approach to simulations. This has nothing to do with this lecture...

# Nothing happens, or “executable not found” message in FLAIR

## Verify your installation

- Please ensure you have downloaded the right version of FLUKA for your platform and you have all packages needed for compiling and executing fortran programs
- Typically for linux distributions you will need the
  - g77 or compat-gcc-34-g77 **64-bit machines: check FAQ**
  - libg2c
- Test if everything compiles ok by setting the correct \$FLUPRO variable and
  - `cd $FLUPRO`
  - `make`
- If everything is Ok your make will compile all fluka tools and the default executable \$FLUPRO/flukahp

The FLUPRO variable must always be set : do it in your shell initialization script (.bashrc, .tcshrc, .cshrc...).

# Crashes at run initialization

IF the following message appears on the screen ( or in the **nohup.out** file when running FLAIR) and no results are produced

```
Removing links
```

```
Removing temporary files
```

```
Saving output and random number seed  
No ranexample16_2002 generated!
```

# Crashes at run initialization

## Look into the .log and .out files

<input>001.log <input>001.out

where?

- either in your working directory
- or in the temporary subdirectory fluka\_xxxx
- or in the Output Files window in FLAIR

what?

- the last lines of the output file
- the first line of the log file

A few examples follow

# FLUKA version is expired...

You get the following message:

```
**** This version is obsolete and/or ****
**** you are not allowed to use FLUKA ****
**** Please contact Alfredo Ferrari ****
**** CERN-AB/ATB, tel.+41-22-76-76119 ****
**** or look for an updated version ****
**** at http://www.fluka.org ****
```

You have to download an updated version:

Download, recompile (if you have user routines) and relink.

# Formatting errors in the input file

- This is easily recognizable, since in the \*.out file the echo of all line of input file is reported. In case of problem, the output stops with the last card correctly interpreted DURING the last pass.
- In case the problem is in the geometry please look the file fort.16 in the fluka\_#### directory
- FLUKA reads the input file in many passages. Therefore the order of the cards do not correspond in some cases to the order that FLUKA will process the card!
- Be careful: sometimes non visible control characters may appear in a file sent via-mail (not for all mail clients!). In this case use "dos2unix" (a specific rpm in linux exists) They can be eliminated also using simple perl, or editor commands)
- Remember to check always the first cycle you requested (typically *input\_001.out*).

# Typical examples of this case:

```
***** Next control card ***** BEAM      -3.500      -8.2425E-02  -1.700      0.000      0.000      1.000      PROTON
Abort called from FLUKAM reason UNKNOWN PROJECTILE OR "OLD" HEAVY ION OPTION (NO LONGER SUPPORTED) Run stopped!
STOP UNKNOWN PROJECTILE OR "OLD" HEAVY ION OPTION (NO LONGER SUPPORTED)
```

**You are using fixed format, but the SDUM identifying the particle is not well aligned**

```
*** Unable to resolve name element BLCKHOL in card ***
ASSIGNMA      BLCKHOLE      BLKHOLE
*** run stopped ***
```

**Another misalignment in fixed format the finale E of BLACKHOLE is lost**

This was the main error source before the existence of FLAIR

# Scoring: USRBIN/EVENTBIN

\*\*\* Activity/fission/neutron balance binnings cannot be track-length!!!

USRBIN scoring method:

- Track-length quantities: that can be distributed along a track, e.g. fluence, energy deposition...  
**WHAT(1) ≥ 10**
- Point-wise quantities: that have to be scored on a point, or in the middle of the step!  
e.g Activity, Fission, Neutron balance...  
**WHAT(1) < 10**

# Other errors in scoring definitions

- Example: badly defined userbin limits

```
***** Fluka stopped in Usrbin: "usr/eventbin" n. 1 *****  
***** with zero width 0.000 for axis R *****
```

- Never use unit numbers smaller than 20 or higher than 99
  - <20 They are reserved by FLUKA
  - >99 Depends on fortran
- **Never mix** the output of different scoring cards in the same unit

automatically checked by FLAIR

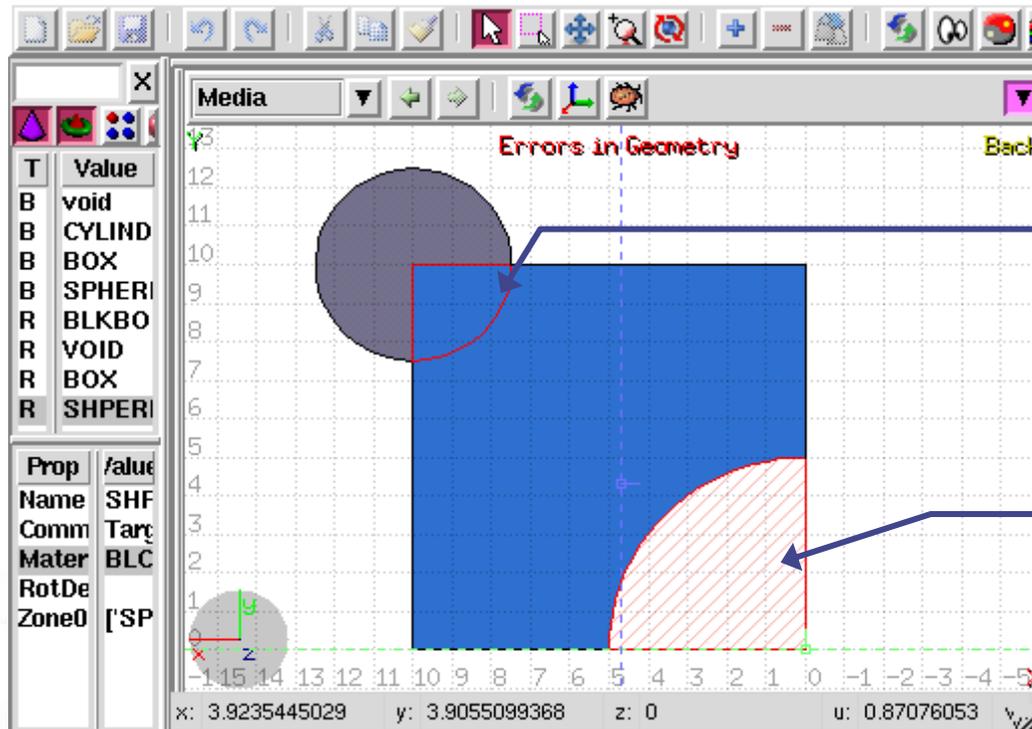
# Have you debugged your geometry?

- The latest version of FLAIR with the geometry editor allows to spot geometry errors (see dedicated lecture!)



- A warning window notifies that are errors in the geometry;
- Non-strictly geometrical errors (i.e. missing Material Assignment to a region, non recognized cards) are also notified;

# Debugging Geometry Errors...



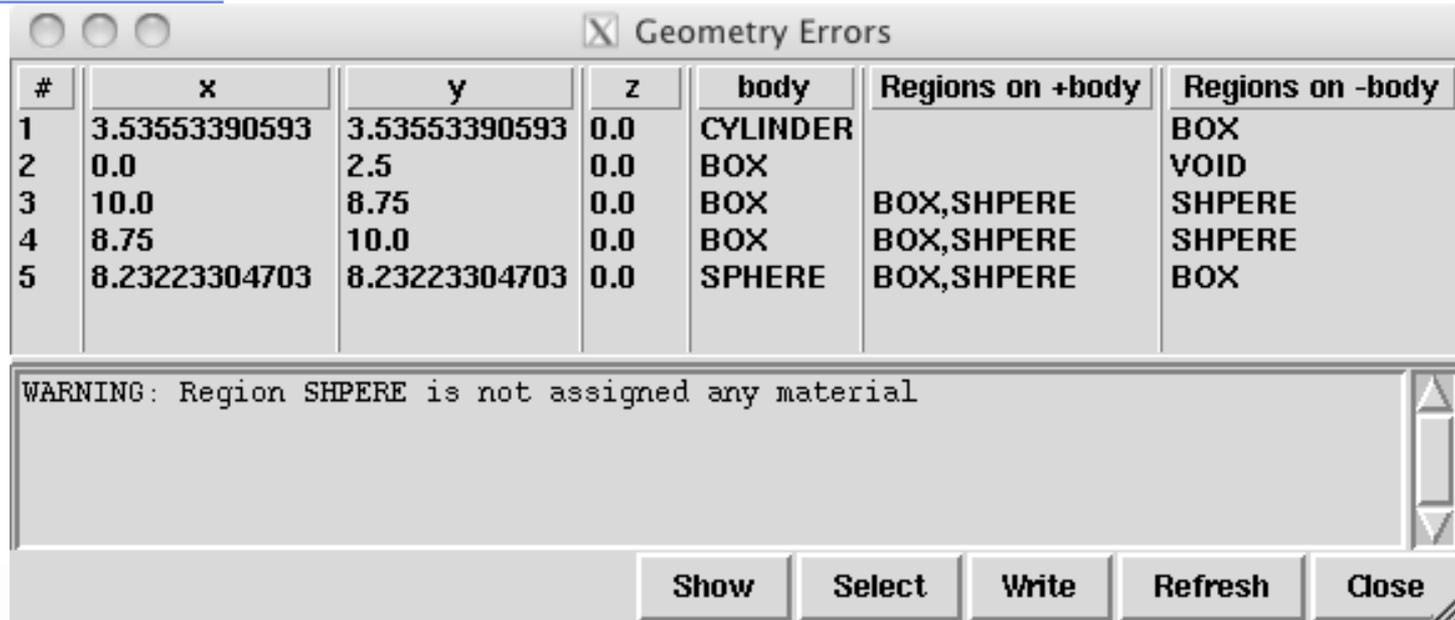
Overlapping regions

Missing region definition

**Errors in Geometry** notifies that are errors in the geometry:

- The areas affected by the errors are outlined with a **Red** stroke:
  - Areas filled with a full color correspond to overlapping regions;
  - Areas filled with red line correspond to a missing region definition;
- Clicking the  icon displays the dialog with the errors.
- Touching surfaces are checked against 10 significant digits

# Debugging Geometry Errors...



The screenshot shows a window titled "Geometry Errors" with a table of error data and a warning message below it. The table has columns for error number, x, y, z coordinates, body name, and regions on both sides of the body. The warning message states: "WARNING: Region SHPERE is not assigned any material".

#	x	y	z	body	Regions on +body	Regions on -body
1	3.53553390593	3.53553390593	0.0	CYLINDER		BOX
2	0.0	2.5	0.0	BOX		VOID
3	10.0	8.75	0.0	BOX	BOX,SHPERE	SHPERE
4	8.75	10.0	0.0	BOX	BOX,SHPERE	SHPERE
5	8.23223304703	8.23223304703	0.0	SPHERE	BOX,SHPERE	BOX

WARNING: Region SHPERE is not assigned any material

Buttons: Show, Select, Write, Refresh, Close

**x, y, z** Coordinates of the error (on the surface of **body**)

**body** Body with the **x,y,z** point on surface generating the error

**+body** Regions that are on the **+** side of the **body**.

Regions where the body should be **subtracted** to remove the error

**-body** Regions that are on the **-** side of the **body**.

Regions that the **body** should be **intersected** to remove the error

**+/-** are defined according to the normal on the surface, **+** refers to outside, **-** to inside

# Geometry: Parenthesis expansion

- Starting the run you get the following message in the first line of **xxx.log** file:

Subscript out of range on file line 56, procedure rpnorm.f/rpnorm.  
Attempt to access the 114705-th element of variable tx.

- You have exceeded the maximum expansion limit for parenthesis. Which is 100 000.
- **WARNING** even with a simple region expression you could easily reach this limit especially if you are using infinite bodies!

## LOW-MAT errors [1/2]

```
*** (n,p)  proton production activated for Xsec mat. #    3 ***
**** Low energy neutron xsec not found for some media 12 13 ****
POTASS
```

There a material has been defined using a name which has no correspondence with the low energy neutron cross sections available in FLUKA (see chap. 10 of the manual). In this case POTASS instead of POTASSIU

Check the names of the materials.

Check the use of **LOW-MAT** cards: check if are using the updated parameters listed in the manual.

Do you really need **LOW-MAT**? This is necessary only in a limited number of cases and it is easy to misinterpret this command if you are not an expert user.

*[Read carefully the manual about this]*

## LOW-MAT errors [2/2]

- WARNING on the change of the temperature
  - New library with 260 groups uses: **296 K**
  - Old library with 72 groups uses: **293 K**
- When an error is printed FLUKA prints the SDUM of the LOW-MAT card. If the SDUM is empty you will see nothing in the output

# Geometry: Errors during tracking

- The run crashes during tracking, the .out file says

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

***Did you debug your geometry????????***

To better direct your debugging

**LOOK AT THE <input>00n.err file in the temporary subdirectory**

See also the use of **RAY** to trace and analyze your geometry

# Geometry: Errors during tracking

Geofar: Particle in **region 3** (cell # 0) in **position** 1.000000000E+00 0.000000000E+00 1.000000000E+00 is now causing trouble, requesting a **step** of 6.258867675E-07 cm to **direction** -2.285059979E-01 -9.412338141E-01 2.487245789E-01, error count: 0 [...skipped...] Particle index 3 total energy 5.189748600E-04 GeV Nsurf 0 We succeeded in saving the particle: current region is n. 2 (cell # 0

- Getting GEOFAR errors during tracking is pointing to:
  - errors in the geometry
  - Numerical precision errors
  - Perpendicularity of the **REC** and **BOX** axes (Use maximum precision)
- FLUKA will try to “save” the problematic particles. But if they are too many it will stop the run.
- G1 errors means a problem in your geometry. **Please check your geometry** for possible errors. The problematic point is given by **position + step \* direction**
- **LATTICE** are VERY sensitive to numerical precision. Use as many digits as possible to ensure proper description of the lattice, transformation and prototype.

# It runs..but

something seems wrong...you do not understand the results

**LOOK in the output file!**

what?  
everything!

a few examples follow, you'll find out more....

# General: RANDOMize

You run several cycles and the statistical errors appears to be **ZERO**.

- Most probably you are starting the same histories on every cycle. This is a typical error if the **RANDOMize** card does not exist!
- Verify the existence of **RANDOMize** card and that **WHAT(1)=1.0**
- You get the following message in the fluka\_### dir

```
**** No Random file available !!!!!! ****
```

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

```
STOP NO RANDOM FILE
```

**Most probably the error is on the previous cycle which didn't generate a random number seed**

# Primary particle cards

The total / partial energy balance is not what expected

- Remember that for HEAVY IONS the Energy or Momentum in the beam card is PER NUCLEON, while in all other cards in FLUKA the energy is the total kinetic one. Verify in the beam section of the output
- Check if you selected Energy or Momentum in BEAM
- Verify that your primary source **BEAMPOS** is **NOT defined on a surface boundary**, but is always inside a region.
- If you define a **SOURCE** routine the standard FLUKA source is NOT CALLED.
  - You have to define all distributions.
  - You need a **BEAM** card to define the maximum energy of your problem

# Media: MATERIAL errors

The total / partial energy balance is not what expected or the particle fluences have discontinuities

verify

- You specified the density for the MATERIALS
- You **supplied the Z,A** when specific **isotopes** are requested
- Compounds: Mixing fractions (signs)
- Recursion
- Numbers out of order when copying/pasting (Use always the NAMES format)

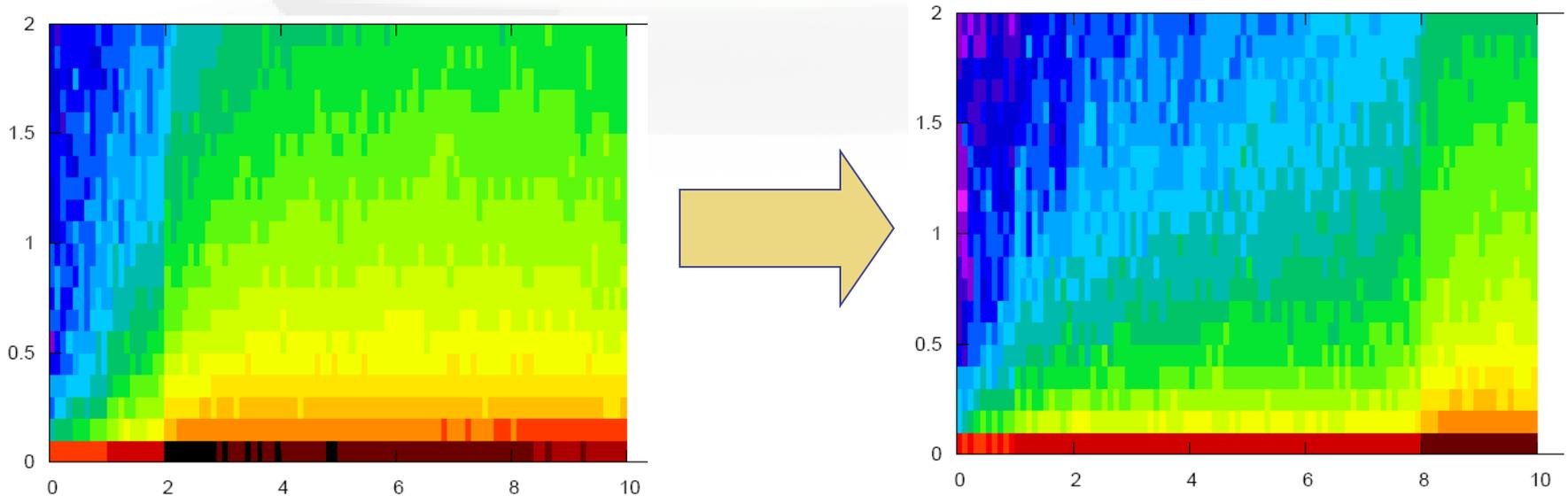
# GEOMETRY errors again

possible symptoms : the energy/fluence sharing among regions is not what expected. Problems with boundary-crossing estimators (low fluence, empty..)

***Did you debug your geometry????????***

Overlapping regions are not detected at initialization, and do not cause run-time errors.

Example: in "our" target, make targs2 overlap with targs3



# Preprocessor

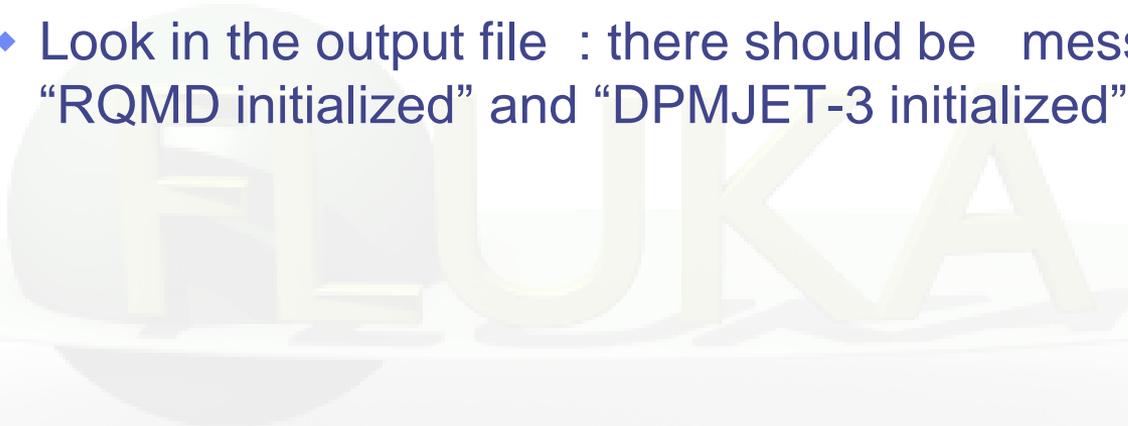
Preprocessor directives are very powerful, but very quickly can make your input unreadable. Please use with care

- Check every #if should have a corresponding #endif
- Up to 10 nesting are allowed
- Up to 40 definitions

# Heavy Ions

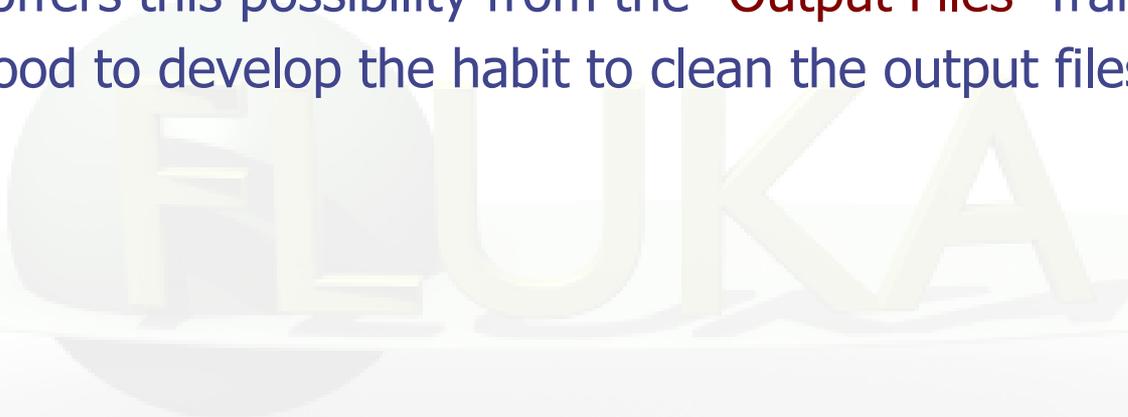
You have an heavy ion beam and no reaction products

- Did you link the interaction generators with `ldpmqmd` ?
  - ◆ Look in the output file : there should be messages like “RQMD initialized” and “DPMJET-3 initialized”



# Merging cycles from different inputs

- Verify that you didn't merge cycles from different runs that the input has been modified.
- It's a good habit to clean the files before starting a run.
- Flair offers this possibility from the "Output Files" frame.
- It's good to develop the habit to clean the output files from test runs.

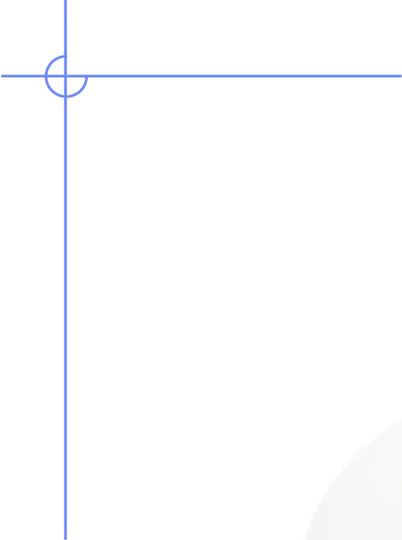


# FLUKA users and the manual

- All reported here so far implies that a FLUKA user is living symbiotically with the Manual....



FLUKA



**NOW PANIC**

# Run time errors with exceptions...

## You will see a message of this kind:

```
===== Running FLUKA for cycle # 1 =====  
luka: line 309: 30048 Aborted (core dumped) ${EXE} <${INPF} 2>${LOGF} >${LOGF}
```

The temporary directory remains there. You will find the \*.log, \*.out, \*.err files, a core.\* file and the last random seed

The last random seed allows to restart your run from a configuration which was the one occurring at maximum 5 minutes of CPU before the error!

Could be easy to spot the problem using the GNU debugger (gdb).

# Run time errors with exceptions...

Also in this case, have a look at

the output file

the log file

the error file

- Maybe still a geometry problem, even after many histories if it's well hidden
- The code has many internal checks, some error conditions are recorded in the out/err files

The 1<sup>st</sup> lines of the .log file may contain important information for you or for the experts

# Run time errors with exceptions...

- If you use your own user routines please check them first!
- with the hints from the log file
- with the help of the gdb debugger

# Using gdb [1/5]

- 1) Go inside the temporary directory
- 2) type: `gdb ../flukahp core.nnnn`
  - From flair you can **double click** on the core file from the "Output Files" Frame



```
GNU gdb Red Hat Linux (6.0post-0.20040223.19rh)
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "i386-redhat-linux-gnu"...Using host libthread_db library "/lib/tls/

Core was generated by `/home/battist/flukacourse/Pavia/examples/errors/flukamy'.
Program terminated with signal 6, Aborted.
Reading symbols from /usr/lib/libg2c.so.0...done.
Loaded symbols for /usr/lib/libg2c.so.0
Reading symbols from /lib/tls/libm.so.6...done.
Loaded symbols for /lib/tls/libm.so.6
Reading symbols from /lib/libgcc_s.so.1...done.
Loaded symbols for /lib/libgcc_s.so.1
Reading symbols from /lib/tls/libc.so.6...done.
Loaded symbols for /lib/tls/libc.so.6
Reading symbols from /lib/ld-linux.so.2...done.
Loaded symbols for /lib/ld-linux.so.2
#0  0x009c37a2 in _dl_sysinfo_int80 () from /lib/ld-linux.so.2
```

# Using gdb [2/5]

3) give the command bt (or where)



```
(gdb) bt
#0 0x009c37a2 in _dl_sysinfo_int80 () from /lib/ld-linux.so.2
#1 0x00a02e59 in raise () from /lib/tls/libc.so.6
#2 0x00a04882 in abort () from /lib/tls/libc.so.6
#3 0x005f0baf in sig_die () from /usr/lib/libg2c.so.0
#4 0x005f0c4b in f_setarg () from /usr/lib/libg2c.so.0
#5 <signal handler called>
#6 0x080496f4 in source_ (nomore=0x1) at source.f:123
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
#7 0x0806cb04 in feeder_ (kendcn=0x91520e4) at feeder.FOR:186
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
#8 0x0804d2cb in flukam_ (iflgeo=0x864e948) at flukam.FOR:3228
#9 0x080492f2 in MAIN__ () at fluka.FOR:278
#10 0x0864e836 in main ()
```

**Here it is!**  
**frame #6 in source.f**  
**at line 123**

# FLUKA bug?

- If you have no user code, and you do not understand what happens:
- In that case, prepare a report with ( possibly ) the gdb results and send it to [fluka-discuss@fluka.org](mailto:fluka-discuss@fluka.org) together with a tar file containing:
  - input, output, log, err files
  - user routines (if any),
  - additional auxiliary files (if any),
  - last random seed, (the one in fluka\_####)
  - and any other possible useful information.

## Other cases [2/3]

- An error in reading Unit 14 refers to nuclear.bin: your nuclear.bin file might be corrupted or missing (check all your binary files in \$FLUPRO) or doesn't match the FLUKA distribution
- Error in reading Unit 1: a problem in the random seed! Often in case when you start in with rfluka using `-N n` with  $n > 0$  : check for the existence of `ran***n` in the launch directory!

## Other cases [3/3]

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

- The above message occurs whenever the seed for the run is not available (or is corrupted).
- **Example:** if, for instance you ask for  $n$  cycles, and cycle  $\#n-k$  ends with an error, then the seed for cycle  $\#n-k+1$  is not produced and therefore all runs from  $n-k+1$  to  $n$  will crash in this way...

# The \*.err file

The \*.err file will report errors, but also a lot of warning which have a meaning mostly for the developers.

For instance: the following messages are not errors!

```
*** Frmbrk: we are dealing with a bag of 8 164.306992
*** Frmbrk: we are dealing with a bag of 7 186.47261
*** Frmbrk: a bag of 10 identical nucleons, cannot be managed ***
```

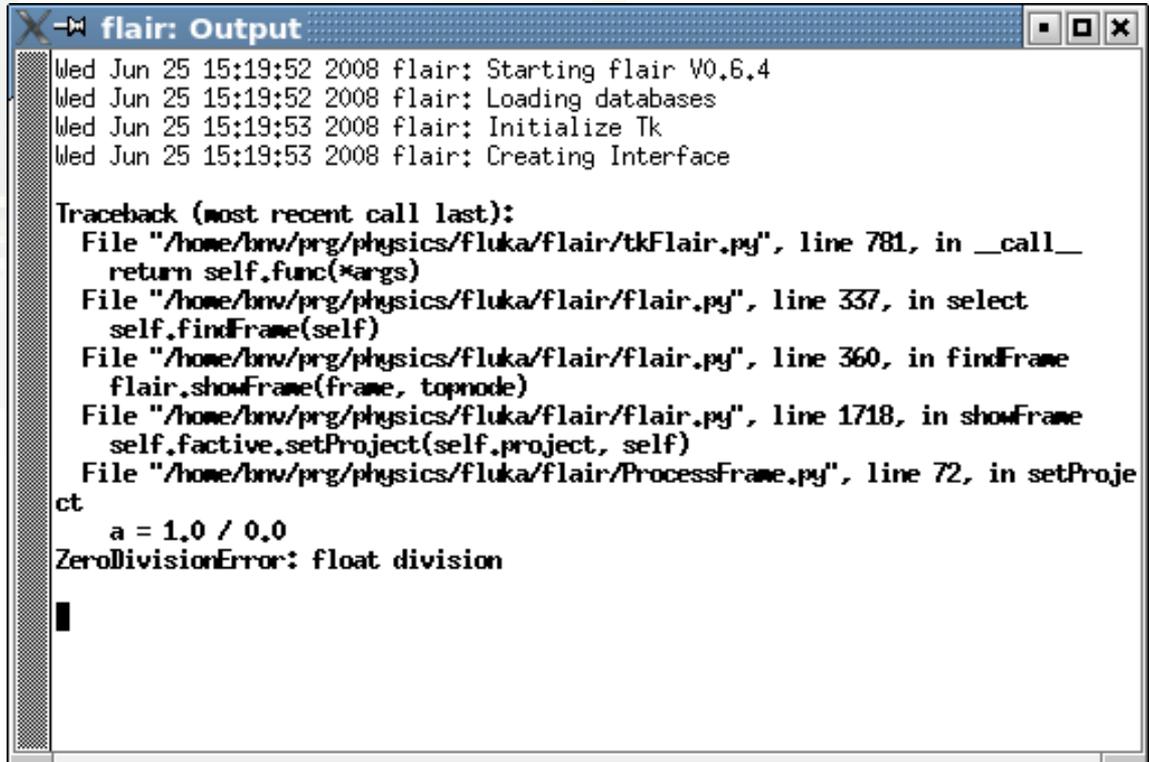
```
NEXT SEEDS:171B5708      1      0      0      0      0 33B49B1      0      0      0
220000      780000      780000      1.6305137E-02      1.00
NEXT SEEDS:1E5C731D      1      0      0      0      0 33B49B1      0      0      0
*** EVENTD: IJ, IBAR(IJ), ICH(IJ), IBTAR, ICTAR, PPERNU -6 4 2 12 6 67.0629729
ECKDPM, PKDPM, PYKDPM, PZKDPM -2.61119861 0.000579929462 -0.0154642238
-2.52471092
KP, IBRSNC(KP), ICRSNC(KP), TVRSNC(KP), ANRSNC(KP), EKRSNC(KP) 1 9 5 0.0885864878
8.39331037 0.00395470202
```

No. of events simulated so far

No. of events remaining to be simulated

# Flair output window

- The output window of flair contains all messages from flair, fluka, processing and plotting tools.
- Always consult it in case of problem or on doubt that something goes wrong



```
flair: Output
Wed Jun 25 15:19:52 2008 flair: Starting flair V0.6.4
Wed Jun 25 15:19:52 2008 flair: Loading databases
Wed Jun 25 15:19:53 2008 flair: Initialize Tk
Wed Jun 25 15:19:53 2008 flair: Creating Interface

Traceback (most recent call last):
  File "/home/bnv/prg/physics/fluka/flair/tkFlair.py", line 781, in __call__
    return self.func(*args)
  File "/home/bnv/prg/physics/fluka/flair/flair.py", line 337, in select
    self.findFrame(self)
  File "/home/bnv/prg/physics/fluka/flair/flair.py", line 360, in findFrame
    flair.showFrame(frame, totnode)
  File "/home/bnv/prg/physics/fluka/flair/flair.py", line 1718, in showFrame
    self.factive.setProject(self, project, self)
  File "/home/bnv/prg/physics/fluka/flair/ProcessFrame.py", line 72, in setProject
    a = 1,0 / 0,0
ZeroDivisionError: float division
```

A Traceback means that something went wrong with flair

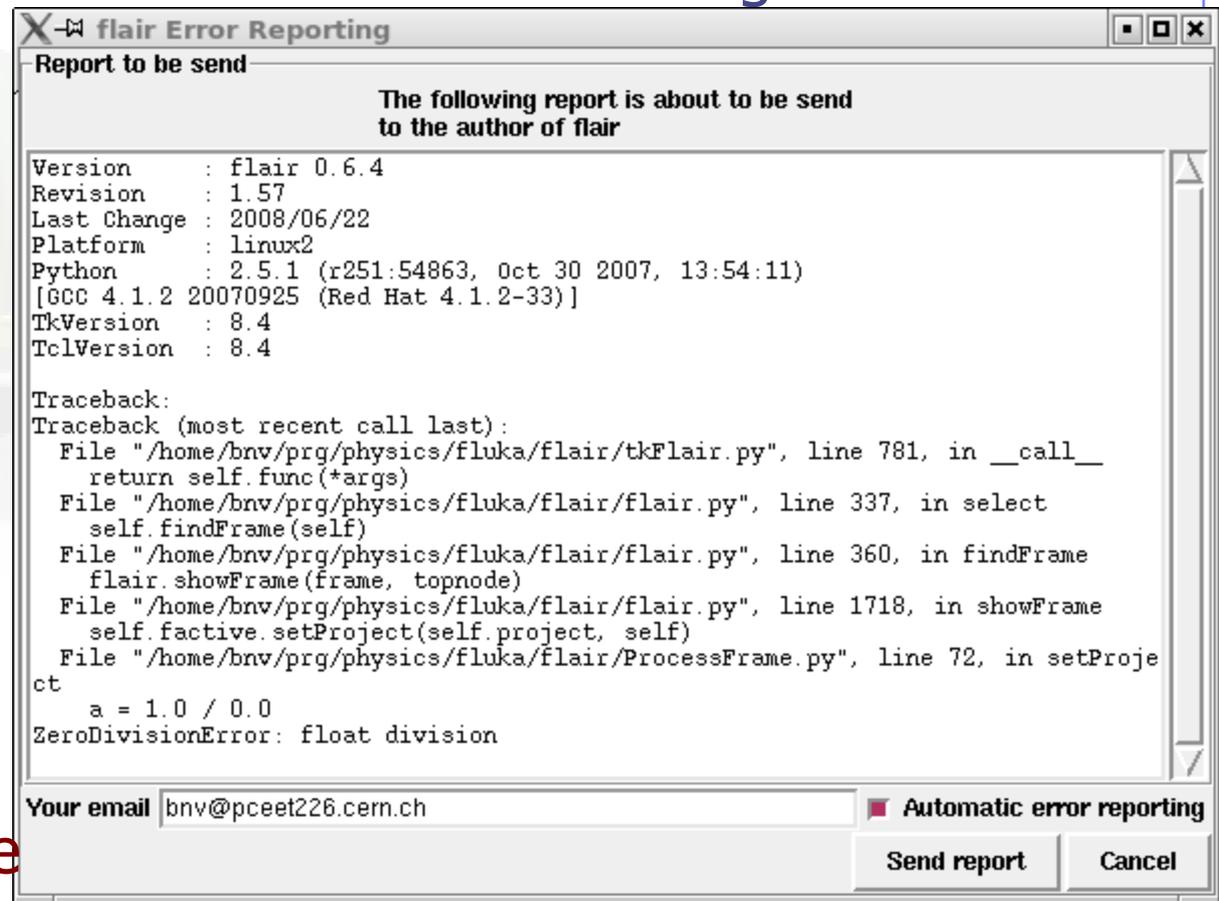
# Flair Error Report

- In case of Traceback, when closing flair or after too many messages flair will propose to send the traceback to the author for further investigation

Please send this report unless if you are sure that is due to your fault e.g. opening a problematic input, directory non existant etc.

It will be good to provide also a small description on how to reproduce the problem.

The text box is editable



# Doubts on the results?

- Firstly, (and not only in case of doubts) read the **FAQ** of FLUKA
- Secondly, make a **search in the FLUKA discussion list** for a similar problem
- Thirdly, if you really cannot understand, or if you need to ask about physics related problems, once again write to [fluka-discuss@fluka.org](mailto:fluka-discuss@fluka.org)

● END



# Attaching Mechanism in flair

- flair is using a **file based mechanism** to identify the status of a run.
  - Increases **portability across various platforms and batch systems**.
  - However **if the files are not generated**, and you get the message **“Waiting to attach”** is impossible for flair to understand if there is a real problem or the run takes a long time to start
    - Please look at the **“Output Files”** to find out the problem
- The **“Output Files”** offers the possibility to delete all files and temporary directories generated by FLUKA is a good habit to clean them before starting a run.
- When creating additional runs in the same project don't forget to give a **different input name**

# Typical examples of this case: [2/3]

```
*** Unknown control code: ROT-DEF , ignored ***
```

```
***** Next control card ***** ROT-DEF 0.000 0.000 0.000 0.000 0.000 0.000 NEW-DEFA
```

```
**** Unknown Input card !!!!!!!!!!! ****
```

**You mistyped a FLUKA command ROT-DEF instead of ROT-DEFI**



```
*** The 3th field -50 of the following input card ***
USRBDX 99.0 218.0 -50 TARGS1 TARGS2 78.5398 Sp1ChH
*** does not contain a valid formatted fortran real number!!! ***
*** It is ambiguous and it could be read differently on different compilers ***
*** depending whether it defaults or not to the blank=0 formatted input rule ***
```

**You gave a numerical value without the "."**

FLUKA expects all numbers with a floating point representation even the integers ONLY if they are not aligned to the right

## Typical examples of this case: [3/3]

```
*** The 1th field -.9256E-3 of the following input card ***  
COMPOUND -.9256E-3 NITROGEN-.2837E-03 OXYGEN-.01572E-3 ARGON AIR  
*** does not contain a valid formatted fortran real number!!! ***  
*** It is ambiguous and it could be read differently on different compilers ***  
*** depending whether it defaults or not to the blank=0 formatted input rule ***
```

### Another misalignment in fixed format: the E-03

#### Remember

The total line length should not be greater than 132 characters

# FLUKA Running procedure

- When **rfluka** is executed the following procedure takes place
  - Check the existence of **\$FLUPRO**
  - Check the existence of the executable  
default: **\$FLUPRO/flukahp**
  - Create a temporary directory named **fluka\_####** where **####** is the process id given by the system to the rfluka job
  - Run the executable which in turn it will create a **<input>.log / .out / .err** files containing the output of fortran file units 6, 11, 15, see the **(IUNIT)** include file
  - Check the existence of the **ran<input>###** the random number seed from the **previous cycle** if any otherwise create one based on the **RANDOMize** card

# Running through flair

- Don't forget that flair is using by default the "nohup" program to run (See the submit command in the Preferences dialog)
- **nohup is used to detach the execution of rfluka from the current terminal.** This is important otherwise when you exit flair or your terminal the run will stop.
- nohup is generating by default a "nohup.out" file located in your project directory containing a copy of all output of rfluka printed on the screen.
- The nohup.out is accessible from the "Output Files"

# Remember!!

ALWAYS look the content of the

*<input>###.out*

*<input>###.log*

*<input>###.err*

The error message most probably will be either  
on the **LAST** line of the *<input>###.out*

**WARNING:** echo order of input cards is not the same as  
in the input

or

on the **FIRST** line of *<input>###.log*