



Handling of errors and crashes

7th FLUKA Course
NEA Paris, Sept.29-Oct.3, 2008





DON'T PANIC

(Florian: +33450FLORIAN)
[your favorite teacher]

Overview

We often receive request of help from users

- 90% of times these requests are simple problems that the user could have located himself: the first purpose of this lecture is to help you how to spot an error which depends on the user.
- In the 5% of 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 5% of cases, the debate is on the meaningfulness of results. This has nothing to do with this lecture....

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
 - libg2c
- Test if everything compiles ok by setting the correct **\$FLUPRO** variable and
 - `cd $FLUPRO`
 - `make`
- If everything is Ok you make will compile all fluka tools and the default executable **\$FLUPRO/flukahp**

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 not accessible from the "Output Files" but only from the "File Browser"

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.

Remember!!

ALWAYS look the content of the

<input>###.out

<input>###.log

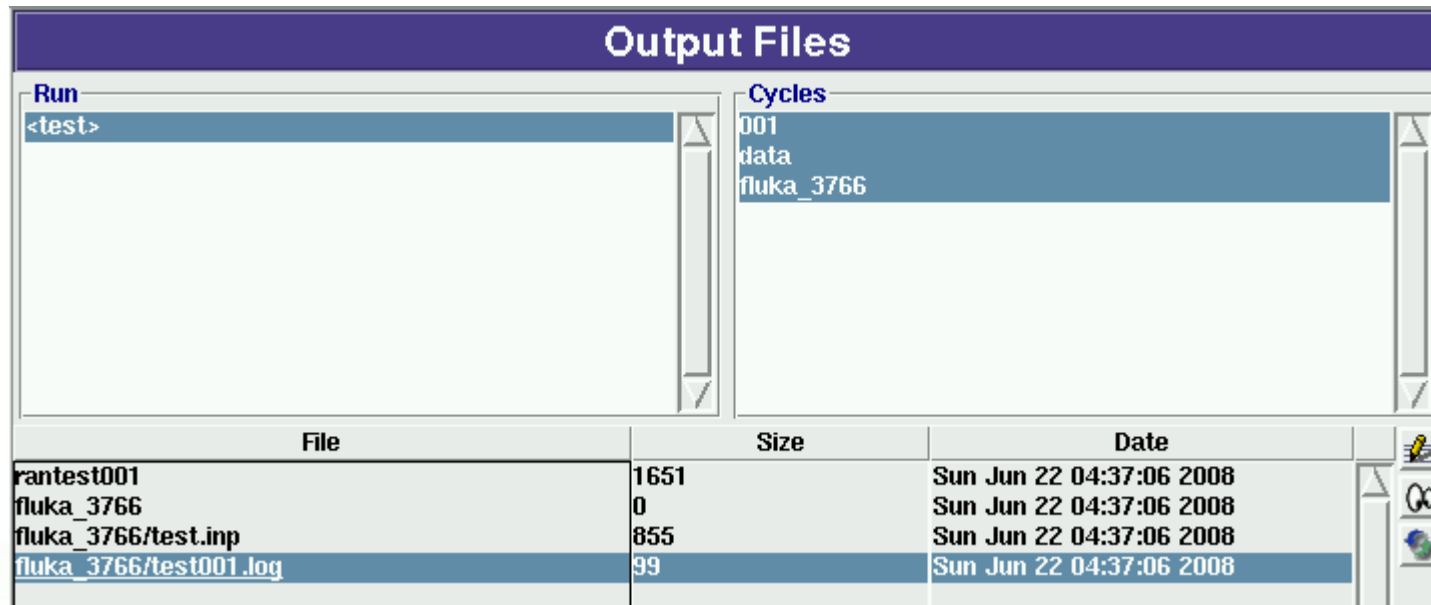
<input>###.err

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

or

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

Waiting to attach



The screenshot shows a window titled "Output Files" with two panes at the top. The left pane, labeled "Run", contains the text "<test>". The right pane, labeled "Cycles", contains the text "001", "data", and "fluka_3766". Below these panes is a table with columns for "File", "Size", and "Date".

File	Size	Date
rantest001	1651	Sun Jun 22 04:37:06 2008
fluka_3766	0	Sun Jun 22 04:37:06 2008
fluka_3766/test.inp	855	Sun Jun 22 04:37:06 2008
fluka_3766/test001.log	99	Sun Jun 22 04:37:06 2008

- Double clicking on the .log file we see the message

```
/usr/local/fluka/flutil/rfluka: line 279: /ramdisk/home/knoppix/brw/tmp/myfluka: Permission denied
```

there is an error accessing the executable file. Either it doesn't exist or you have no permission

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.

Something wrong in the input file!

- By far this is the most common problem. 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
- 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: [1/3]

```
***** 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

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  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

General: Files and Unit numbers

- 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
- When programming prefer to use the routine **OAUXFI** that will correctly determine the location of the file you want to read from the following locations:
 - Running temporary directory fluka_#####
 - Current directory where .inp is located
 - FLUKA directory pointed by \$FLUPRO

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

- 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 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

Geometry: Have you debugged it?

- Sometimes input errors are in the geometry
- 3 types of mistakes:
 1. trivial (wrong data cards)
 2. essential (non defined space points, multiple definition of space points): see the lectures on geometry and the use of **DEBUG** in the **GEOEND** card
 3. Precision errors due to insufficient digits or huge geometries

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

Geometry: Trivial mistake...

There a ZCC is defined with only 2 numbers instead than 3.

```
* Lead target  
ZCC TARG      0.0 0.0
```

You will find:

```
For the body = TARG      type = ZCC      2 numeric parameters were given instead of 3
```

The important geometry errors may appear as a lot of messages in the *.out and *.err file, complaining that a point in space is not defined, cannot be reached, etc.

Remember the need for proper normalization of direction cosines! (magfld, source etc.) You will get a lot of (warning) messages also in those cases, but this is not a true error...

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!

Geometry: Errors during tracking [1/2]

```
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
- 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.
- **LATTICE** are VERY sensitive to numerical precision. Use as many digits as possible to ensure proper description of the lattice, transformation and prototype.

Geometry: Errors during tracking [2/2]

- The error position is given by the following vector operation

position + step * direction

The logo for FLUKA, featuring the word "FLUKA" in a stylized, 3D, metallic font. The letters are arranged in a slightly curved line, and the entire logo is set against a light, circular, glowing background that resembles a lens flare or a soft light effect.

Media: MATERIAL errors

- That you have specified a material for every region
- You specified the density for the MATERIALS
- You **supplied the Z,A** when specific **isotopes** are requested
- You **DON'T** supply any **Z** or **A** when a **compound** is requested
- When you want to override an existing FLUKA material **ALWAYS** specify the index **WHAT(4)**

Media: COMPOUND

- Mixing fractions (signs)
- Recursion
- Numbers out of order when copying/pasting (Use always the NAMES format)

The image shows the word "FLUKA" in a stylized, 3D, metallic font. The letters are arranged in a slightly curved line, and the entire word is set against a light, circular, glowing background that resembles a lens or a lens flare. The letters are a golden-yellow color with a metallic sheen and shadows, giving them a three-dimensional appearance.

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

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

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

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.

What is left after these problems

These errors are not generating exceptions. The temporary directory is erased and you remain with this message on the standard output:

```
Removing links
```

```
Removing temporary files
```

```
Saving output and random number seed
```

```
No ranexample16_2002 generated!
```

FLUKA users and the manual

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



FLUKA



NOW PANIC

(Alfredo: +33450ALFREDO)

Run time errors with exceptions...

- These are caused in 90% of cases by problems in user's code or user's configuration

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!

You can easily spot the problem using the GNU debugger (gdb).

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
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
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
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
BFD: BFD 20040223 20040223 assertion fail /usr/src/build/392707-i386/BUILD/gdb+dejagnum-20040223/bfd/libbfd.c:551
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

Using gdb [3/5]

4) type: frame 6

```
(gdb) frame 6
#6  0x080496f4 in source_ (nomore=0x1) at source.f:123
123          PMOFLK (NPFLKA) = SQRT ( TKEFLK (NPFLKA) * ( TKEFLK (NPFLKA)
Current language:  auto; currently fortran
```

5) you can inspect variable content, for instance:

```
(gdb) p TKEFLK (NPFLKA)
$1 = -0.1541494786490023
```

A negative kinetic energy!!!!

Using gdb [4/5]

6) you can list the nearby lines:

```
(gdb) list
118         IF ( ABS(RGAUSS) .GT. 3.D+00) GO TO 555
119     *     TKEFLK (NPFLKA) = SQRT ( PBEAM**2 + AM (IONID)**2 ) - AM (IONID)
120     TKEFLK (NPFLKA) = WHASOU(1) + WHASOU(2)*RGAUSS
121     * Particle momentum
122     *     PMOFLK (NPFLKA) = PBEAM
123         PMOFLK (NPFLKA) = SQRT ( TKEFLK (NPFLKA) * ( TKEFLK (NPFLKA)
124     &         + TWOTWO * AM (ILOFLK(NPFLKA)) ) )
125         WRITE(*,*) TKEFLK (NPFLKA), PMOFLK (NPFLKA)
126     * Cosines (tx,ty,tz)
127     TXFLK (NPFLKA) = UBEAM
```

It's quite obvious that this was not a protected statement!

Using gdb [5/5]

- In the remaining 5% of cases there will be an error inside a FLUKA routine.
- In that case, if you have not the source, prepare a report with the `gdb` results and send it to fluka-discuss@fluka.org together with a tar file containing: input, user routines (if any), additional auxiliary files, last random seed, and any other possible useful information.

Other cases [1/3]

- In case you are using **SOURCE**, and by mistake you sample an energy (or momentum) larger than the **WHAT(1)** value of the **BEAM** card (maximum of **dE/dx** tabulation), then an **ABORT** will be generated at run time.
- The ***.out** file will contain the following message:

```
Abort called from STEPOP reason dp/dx=<0 Run stopped!  
STOP dp/dx=<0
```


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      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, ICHTAR, PPERNU -6 4 2 12 6 67.0629729
ECKDPM, PPKDPM, 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
```

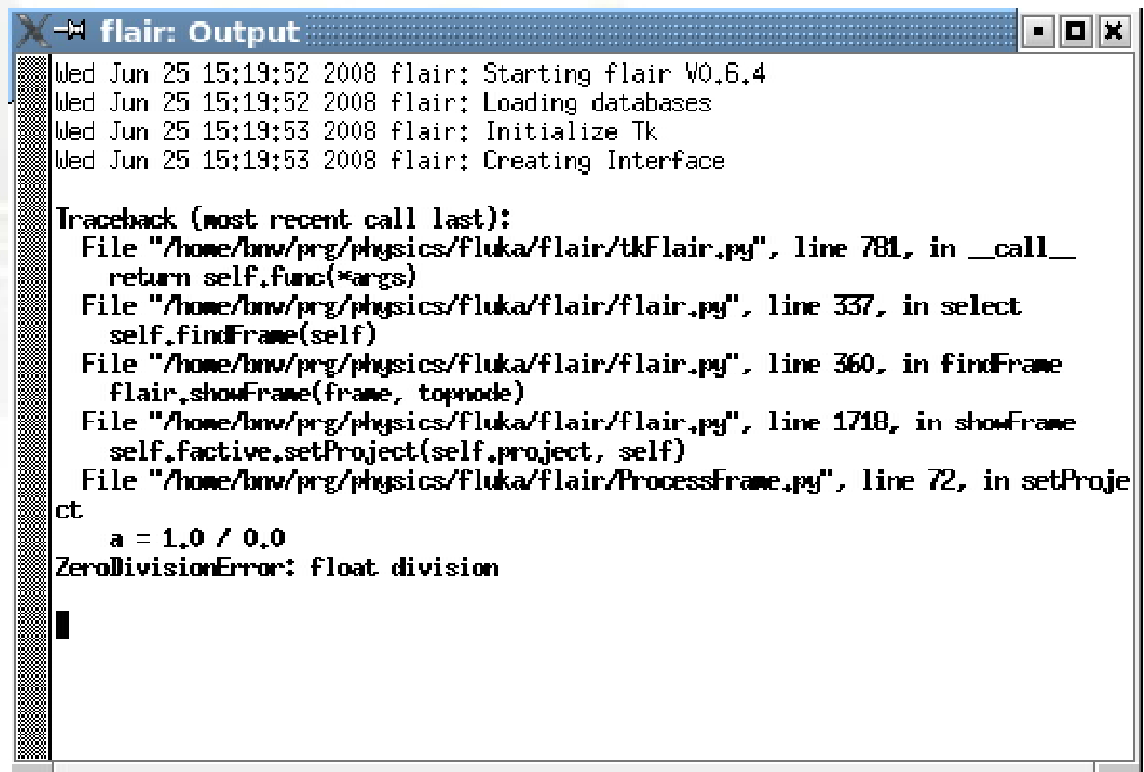
No. of events simulated so far

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

A Traceback means that something went wrong with flair



```
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/bnw/prg/physics/fluka/flair/tkFlair.py", line 781, in __call__
    return self.func(*args)
  File "/home/bnw/prg/physics/fluka/flair/flair.py", line 337, in select
    self.findFrame(self)
  File "/home/bnw/prg/physics/fluka/flair/flair.py", line 360, in findFrame
    flair.showFrame(frame, totnode)
  File "/home/bnw/prg/physics/fluka/flair/flair.py", line 1718, in showFrame
    self.factive.setProject(self, project, self)
  File "/home/bnw/prg/physics/fluka/flair/ProcessFrame.py", line 72, in setProject
    a = 1.0 / 0.0
ZeroDivisionError: float division
```

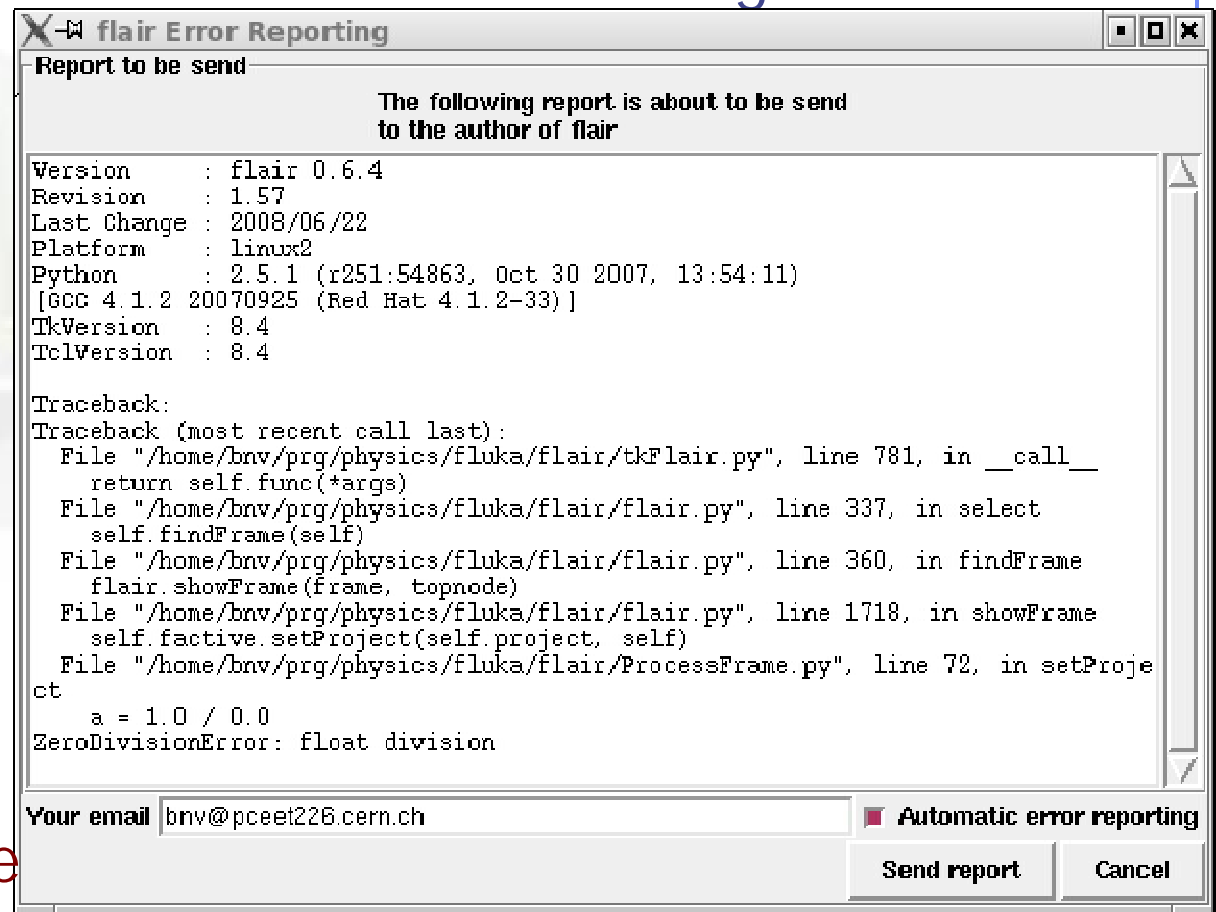
Flair Error Report

- 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?

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!

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