FLUKA

https://fluka.cern

Presentation of the FLUKA particle transport code and tutorial with its graphical user interface Flair

Material prepared by the FLUKA collaboration

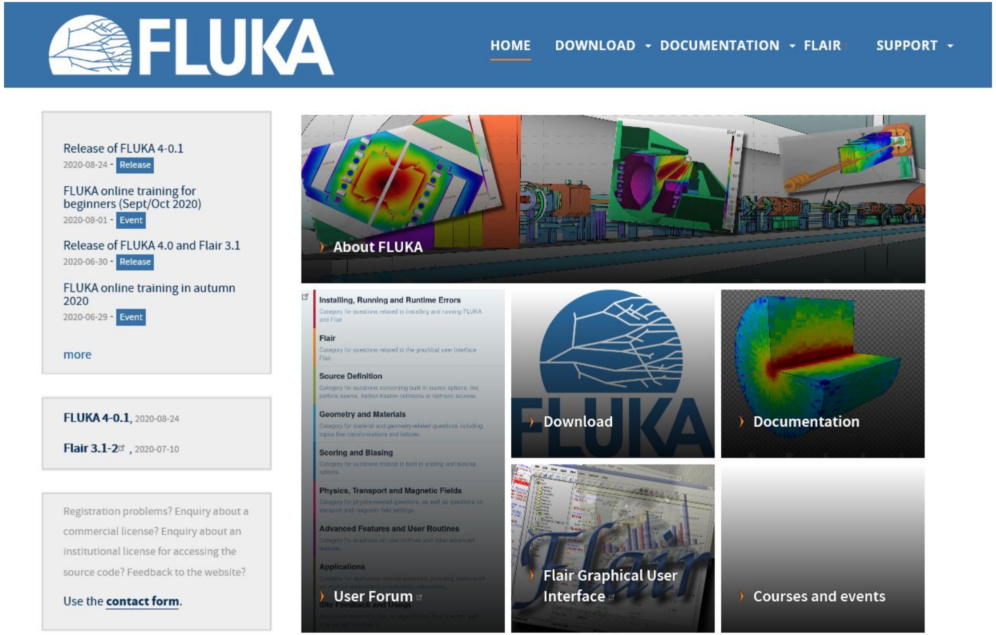
RADMEP workshop 2023

04.12.2023

Introduction

FLUKA.CERN distribution

https://fluka.cern



Code history:
FLUKA was born in the 60's at CERN from Johannes Ranft

It's in active development since then, where several institutes and collaborators have contributed.

Currently the 4th generation of the code is distributed by CERN.

The next release **FLUKA 4-4.0** is scheduled for **early 2024**

Licensing scheme

Registration options	*Includes access to the*
FLUKA **Single User License Agreement**	**source code**
Affiliates of institutes with a FLUKA **Institutional License Agreement**	**source code**
CERN Staff members and Fellows	**source code**
Affiliates of institutes which signed the FLUKA **Memorandum of Understanding**	**development version**
Companies which purchased a FLUKA **Commercial License Agreement**	Binary version only

- **Licenses are free** except for commercial use
- They are granted for **non-military use** only

User support

FLUKA User Forum

https://cern.ch/fluka-forum

Note: an independent one time registration is required to be able to participate

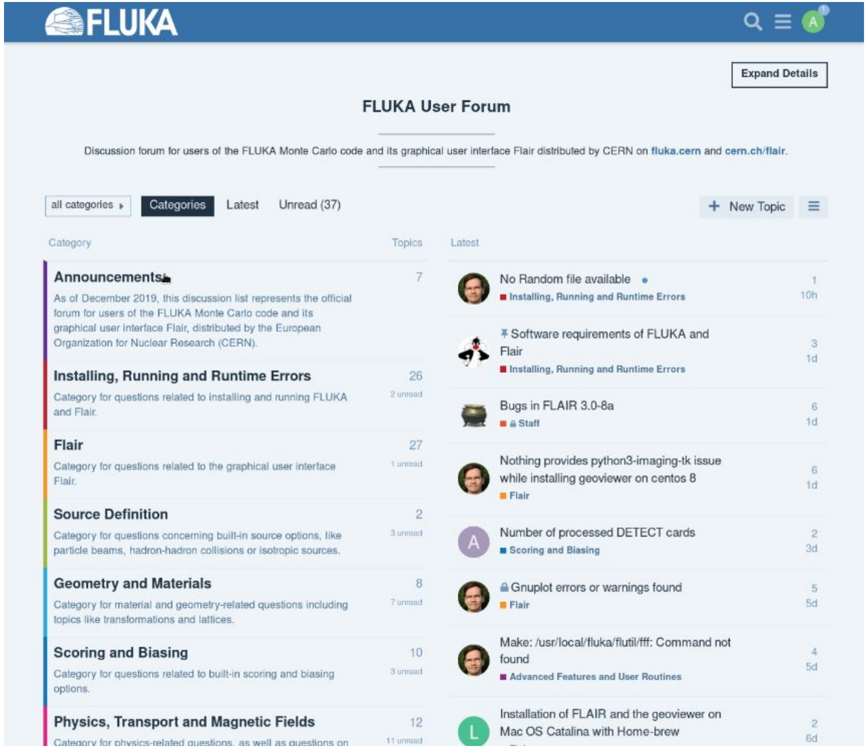
FLUKA Training

Three Beginner Online Training courses held in 2020 and one in 2021.

Two Beginner Training courses:
- University of Brussels in 2022
- NEA, Paris in 2023

and one Advance course held:
- Argonne NL, Chicago in 2023

The next beginner's course is planned for 15-19 April 2024 in Madrid



FLUKA capabilities

- hadron-hadron and hadron-nucleus interactions
- nucleus-nucleus interactions (including deuterons!)
- photon interactions (>100 eV)
- electron interactions (> 1 keV; including electronuclear)
- muon interactions (including photonuclear)
- neutrino interactions
- low energy (<20 MeV) neutron interactions and transport
- particle decay
- ionization and multiple (single) scattering (including all ions down to 250 eV/u)
- coherent effects in crystals (channelling)
- magnetic field, and electric field in vacuum
- combinatorial geometry and lattice capabilities
- voxel geometry and DICOM importing
- analogue or biased treatment
- on-line buildup and evolution of induced radioactivity and dose
- built-in scoring of several quantities (including DPA and dose equivalent)

In support of a
wide range of applications

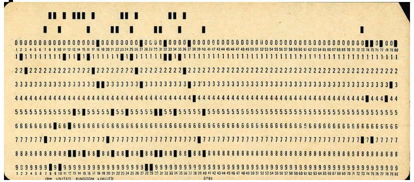
✔ Accelerator design
✔ Particle physics
✔ Cosmic ray physics
✔ Neutrino physics
✔ Medical applications
✔ Radiation protection (shielding design, activation)
✔ Dosimetry
✔ Radiation damage
✔ Radiation to electronics effects
✔ ADS systems, waste transmutation
✔ Neutronics

Basic Input

Some history

- FLUKA's story begun a long time ago (1960s)…

  …no graphical interfaces, input and output via text file

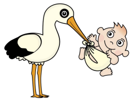


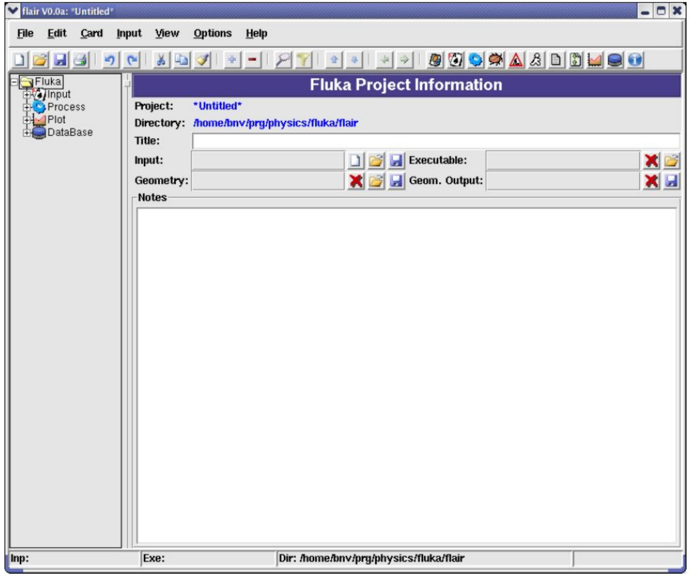
- Inputfile can be very long > 50k lines
- Inputfile based on "*cards*": **`.inp`** file
- Each card has 1 name, 6 values (called WHATs), 1 string (called SDUM)
- Two examples of cards (the actual meaning is not relevant here):

```
BEAMPOS        4750.5     130.0    4866.5                                NEGATIVE
BEAM             -0.4       0.2       5.0      1.E-4     1.E-4           ELECTRON
```

Card name | WHAT(1) | WHAT(2) | WHAT(3) | WHAT(4) | WHAT(5) | WHAT(6) | SDUM

FLAIR

- In 2006, Flair was born! 
  - FLUKA advanced graphical user interface
  - Input file creation
  - Geometry visualization and construction
  - Simulation execution
  - Results visualization

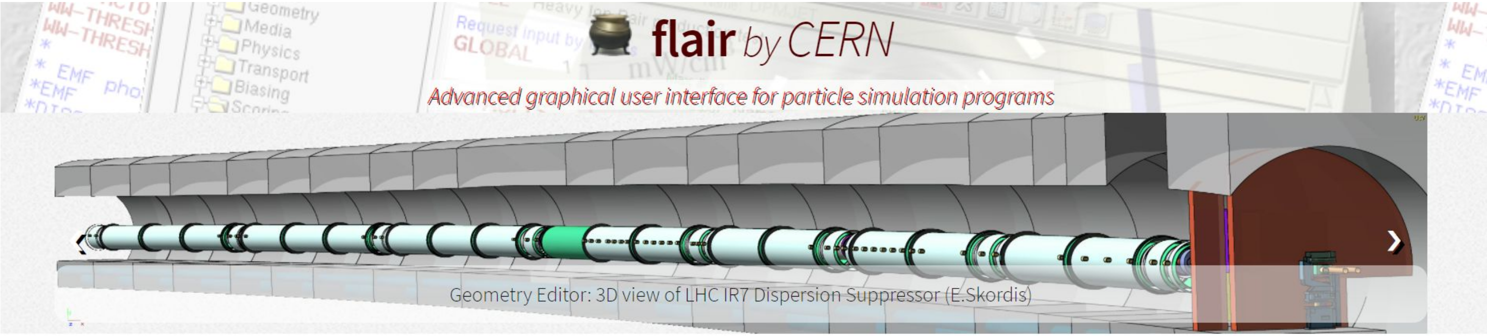


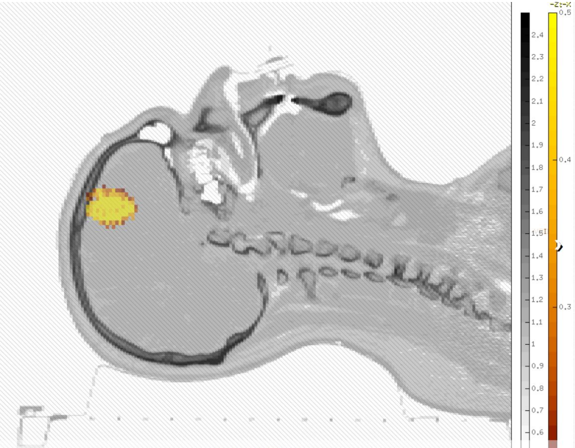
- Flair acts as an intermediate layer between the user and the input file
- It allows a user friendly editing of the FLUKA input
- Based on a `.flair` file and generates the `.inp` file that is run by FLUKA

Flair ≠ FLUKA

FLAIR

https://flair.cern





Head CT with PET CT dose distribution

Current Version

- Latest version: **3.3-0.2**
- Released on: ***Fri 24-Nov-2023***
- Powered by python**3**, tkinter, gnuplot, pydicom

Authors

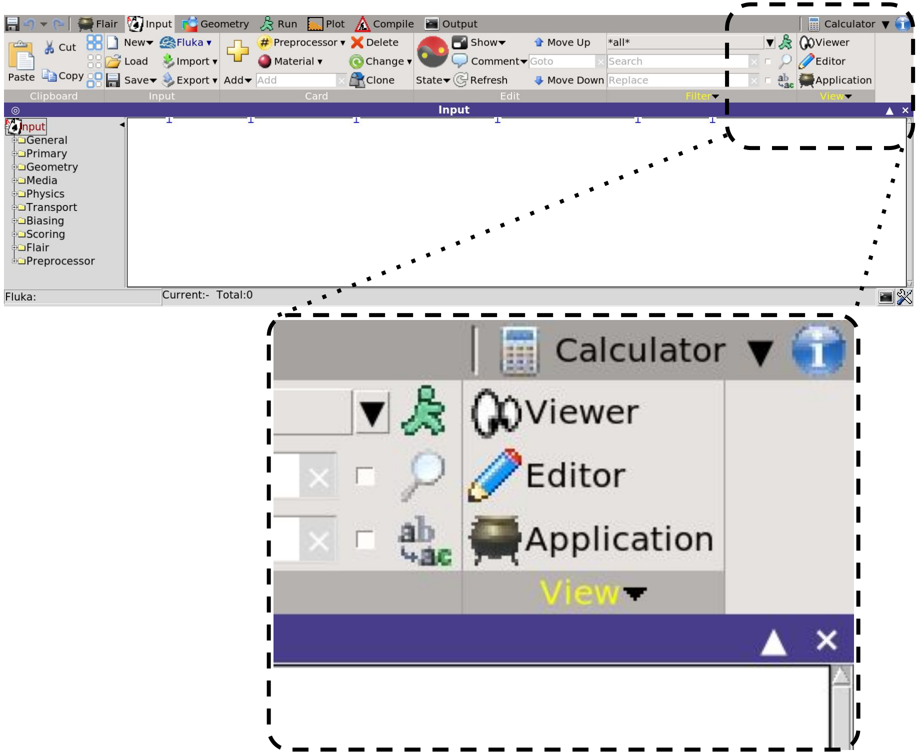
authors: Vasilis Vlachoudis *(lead author)*
Christian Theis
Wioletta Kozlowska

Features

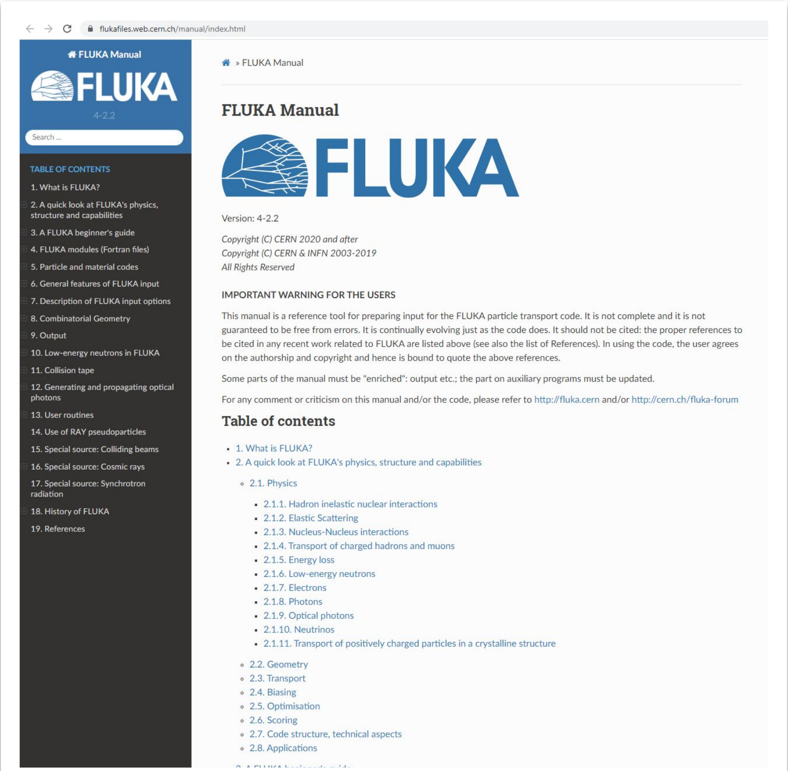
- modern and intuitive design
- Input editor for error free inputs
- Interactive geometry editor, photorealistic ray tracer and debugger
- run and monitor the simulation
- back-end for post-processing of results
- I/O of other simulation formats (MCNPX,GDML,...)
- Medical file importing, DICOM, RT-PLAN,DOSE,...
- extended material library

The FLUKA manual

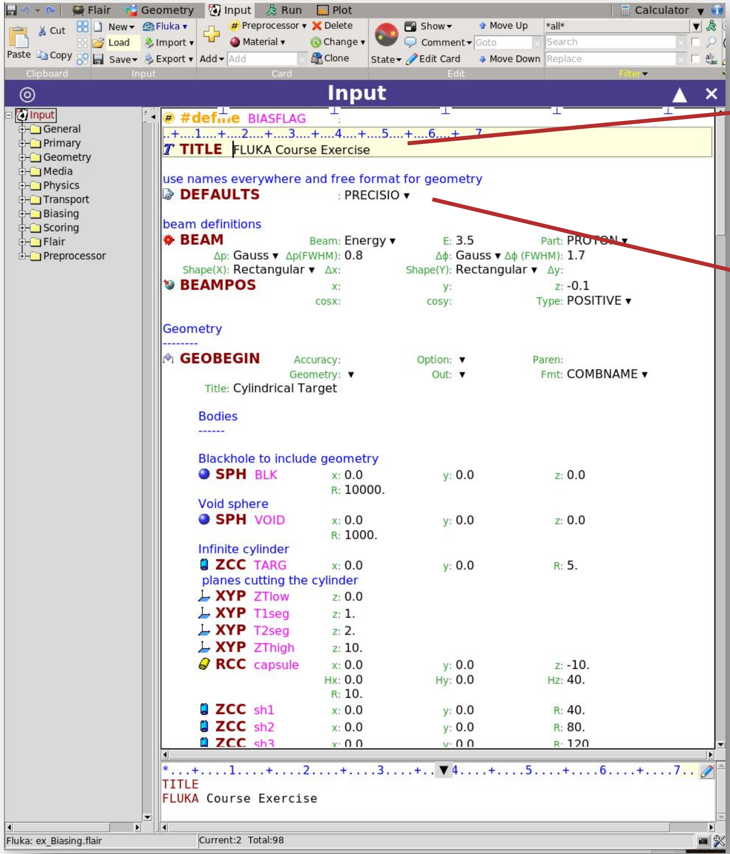
Available in Flair clicking on the “info” button



…or via the FLUKA Web-site



The input at a glance



TITLE

Assign a title to the simulations

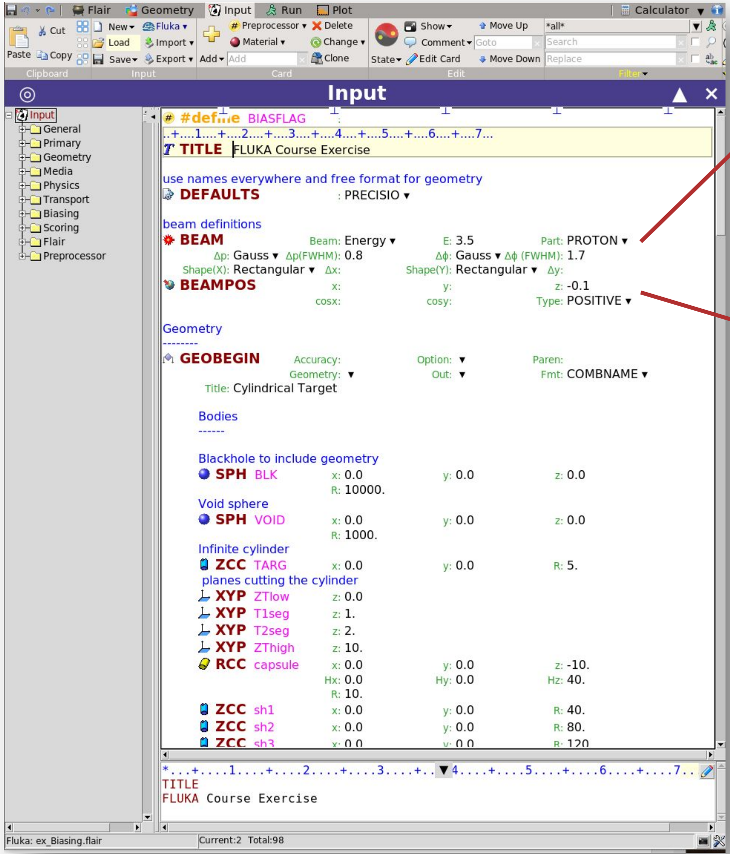
- The title is printed in the output files
- Not a mandatory card

DEFAULTS

Select one of the physics defaults settings

- To be defined at the very beginning of input, only preceded by the **TITLE** and **GLOBAL** cards
- Any of the physics defaults can be overridden later in the input with specific cards
- Given the progress over time in computer power, it is a reasonable approach to always select the most detailed physics defaults (**PRECISIO**) and override specific settings later depending on the needs of the problem

The input at a glance



BEAM

Specify beam particle properties

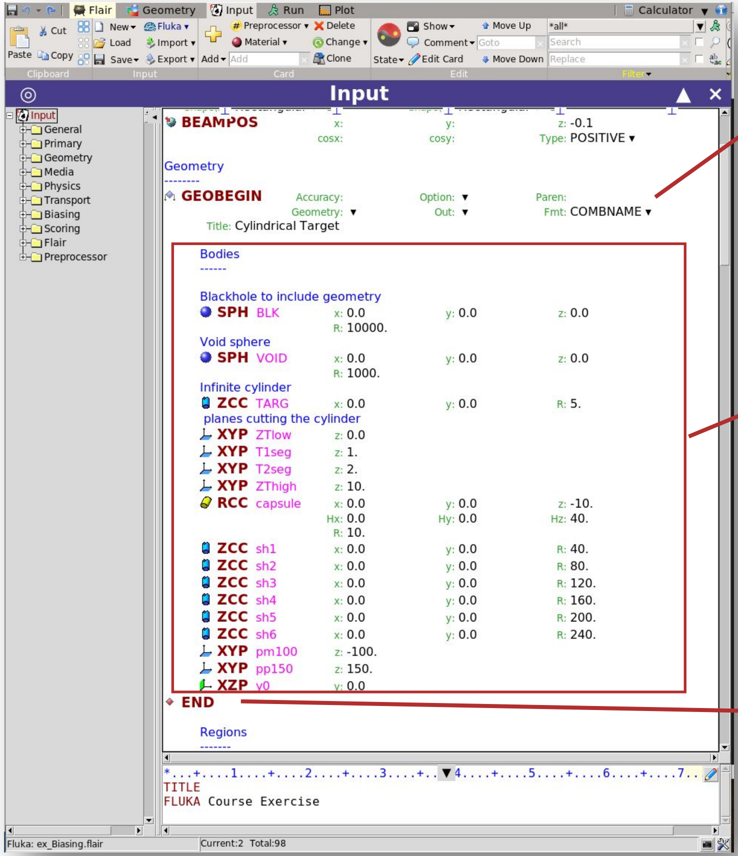
- Particle type
- Momentum or kinetic energy
- Momentum distribution
- Angular distribution
- Shape in the X-Y plane

BEAMPOS

Define beam spot and direction

- Beam spot is defined with its x, y and z coordinates [cm] Default: Origin of the coordinate system
- Beam axis is defined via direction cosines with respect to the x and y axes.
- The third direction cosine (cosz) is automatically calculated by FLUKA, its sign to be provided via Type=POSITIVE/NEGATIVE

The input at a glance



GEOBEGIN

Start of input section that defines geometry

Body definitions

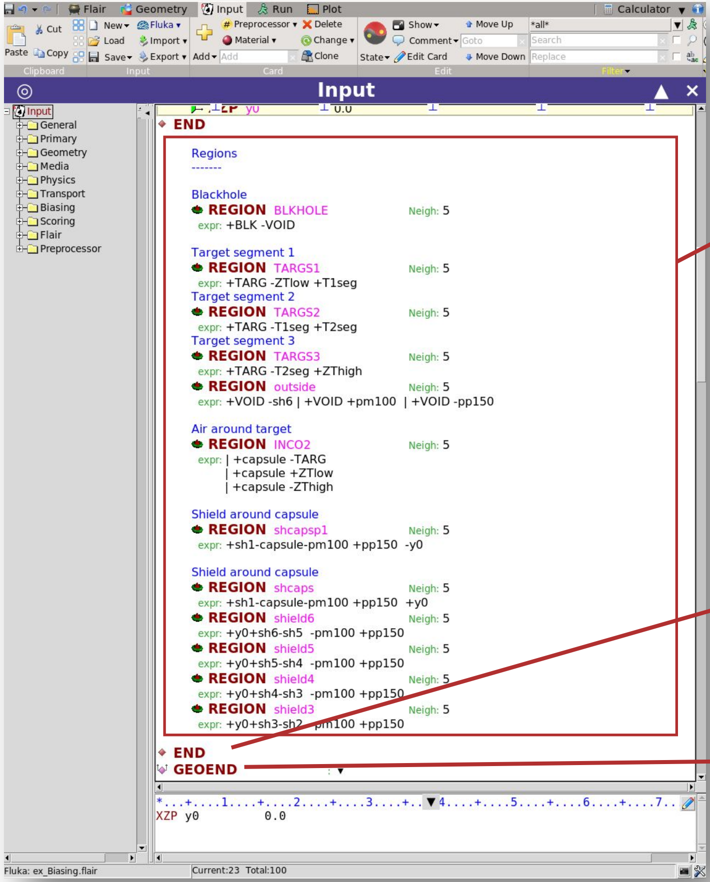
A body definition consists of:

- **3-letter code** indicating the **body type**
- **unique body name** (alphanumeric identifier, 8 character maximum, case sensitive)
- **set of geometrical quantities** defining the body, e.g. the body dimensions and the position in the coordinate system (all values in cm!)

END

Ends the body definition

The input at a glance



Region definitions

A region definition consists of:

- **Unique region name** (alphanumeric identifier, 8 character maximum, case sensitive, must start with an alphabetical character)
- Estimate of the **number of neighboring zones**
- A single **Boolean zone expression** or a **series of Boolean zone expressions** combined via the **union operator**

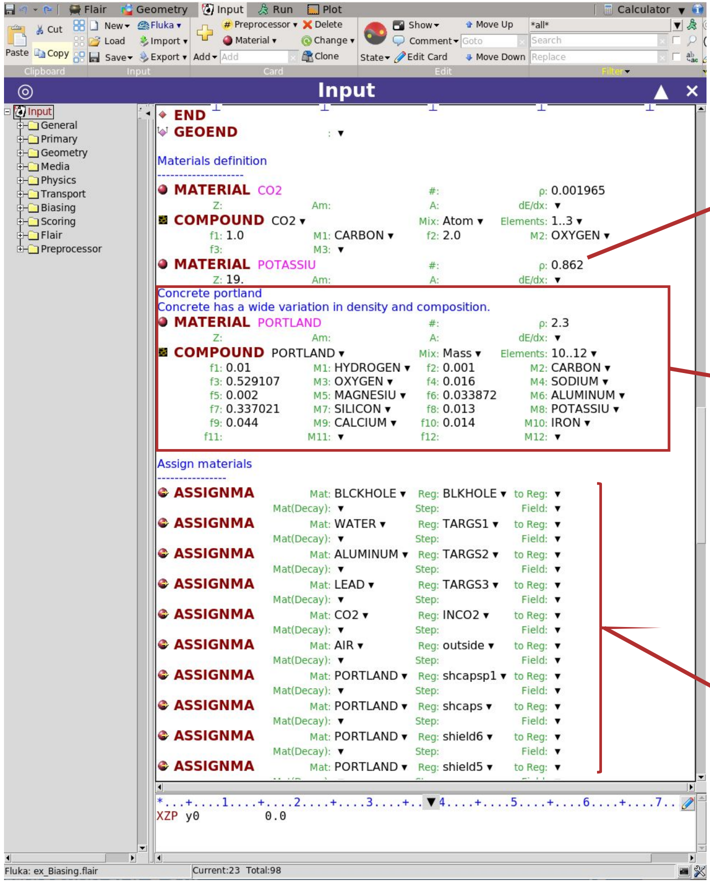
END

Ends the region definition

GEOEND

End of input section that defines geometry

The input at a glance



MATERIAL

Definition of a non-predefined single element

- Each material must have a unique name
- Definition of charge, mass number and density in g/cm^3

MATERIAL/COMPOUND

Definition of composite materials

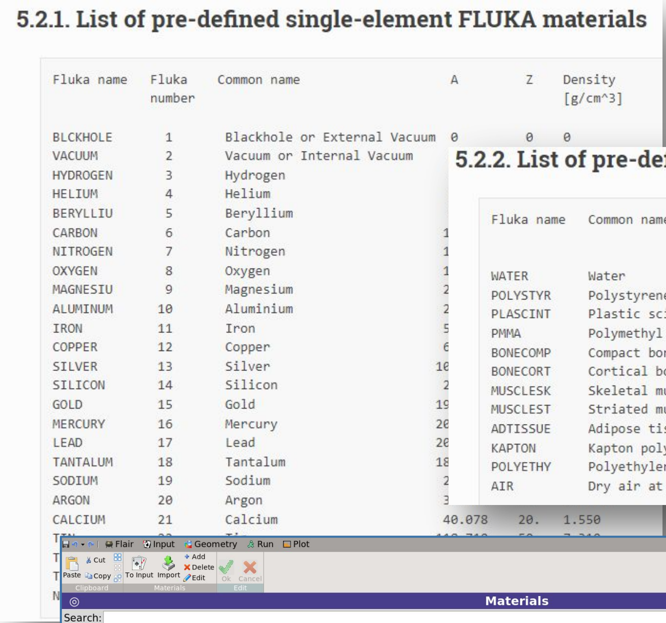
- Each composite material must have a unique name
- Definition of components and their abundances in terms of either atom content, mass fraction or volume fraction
- Definition of density in g/cm^3

ASSIGNMA

Assignment of material to a region

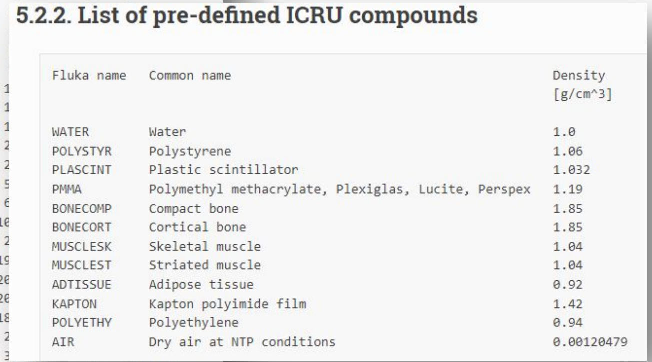
Pre-defined materials

- A number of common materials (23 elements and 12 compounds) are pre-defined in FLUKA and can be assigned to a region without the corresponding material declaration.
- 2 special materials are also included:
  - **VACUUM**: obvious definition. Static electrical fields can be defined only in vacuum.
  - **BLCKHOLE**: Ideal absorber, must be assigned to the "black body" region surrounding your geometry but can also be used elsewhere in the geometry, e.g. for perfect shielding/collimation, to reduce CPU-time by killing tracking in certain regions etc.
- In addition, Flair comes with an extensive library of materials (elemental and compounds) that can be imported into the input



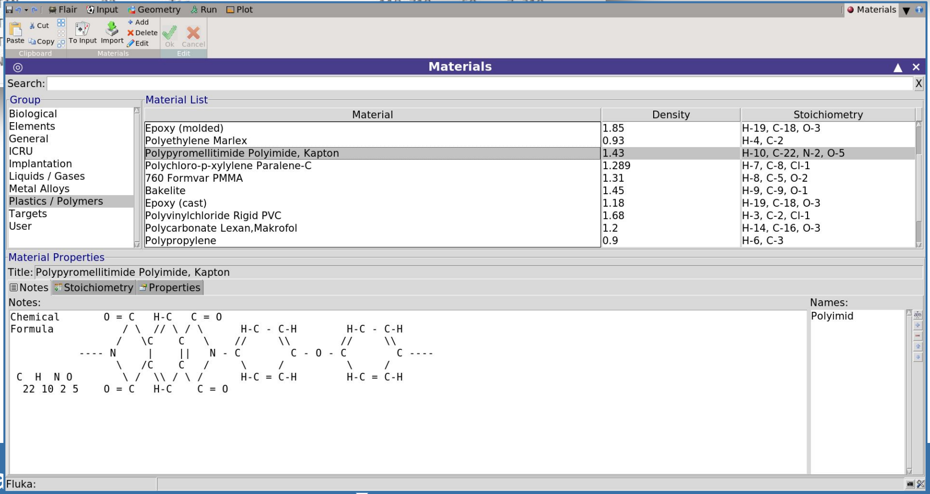
5.2.1. List of pre-defined single-element FLUKA materials

Fluka name	Fluka number	Common name	A	Z	Density [g/cm^3]
BLCKHOLE	1	Blackhole or External Vacuum	0	0	0
VACUUM	2	Vacuum or Internal Vacuum			
HYDROGEN	3	Hydrogen			
HELIUM	4	Helium			
BERYLLIU	5	Beryllium			
CARBON	6	Carbon			
NITROGEN	7	Nitrogen			
OXYGEN	8	Oxygen			
MAGNESIU	9	Magnesium			
ALUMINUM	10	Aluminium			
IRON	11	Iron			
COPPER	12	Copper			
SILVER	13	Silver			
SILICON	14	Silicon			
GOLD	15	Gold			
MERCURY	16	Mercury			
LEAD	17	Lead			
TANTALUM	18	Tantalum			
SODIUM	19	Sodium			
ARGON	20	Argon			
CALCIUM	21	Calcium	40.078	20.	1.550

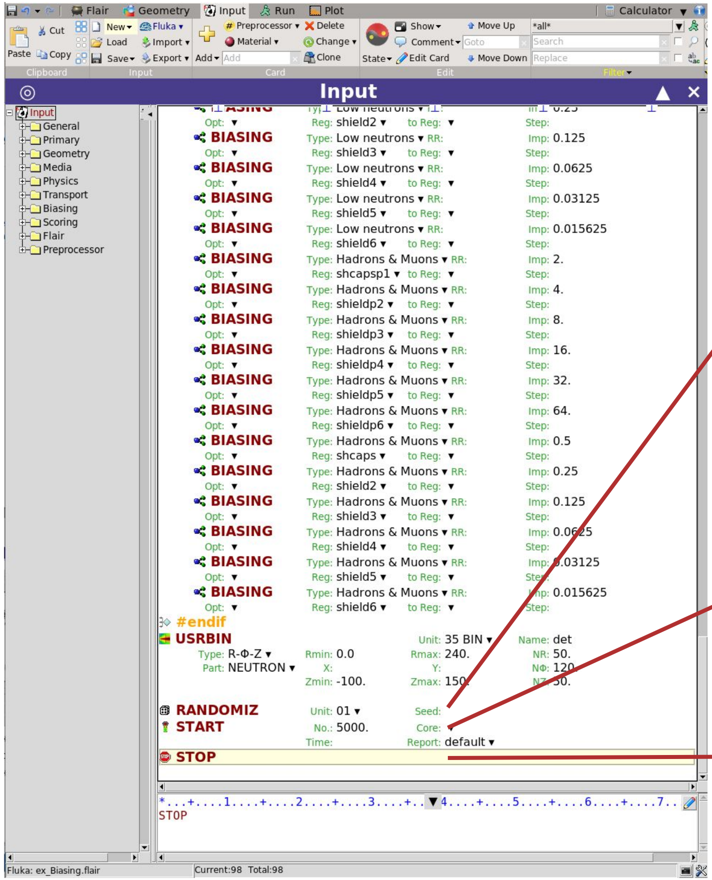


5.2.2. List of pre-defined ICRU compounds

Fluka name	Common name	Density [g/cm^3]
WATER	Water	1.0
POLYSTYR	Polystyrene	1.06
PLASCINT	Plastic scintillator	1.032
PMMA	Polymethyl methacrylate, Plexiglas, Lucite, Perspex	1.19
BONECOMP	Compact bone	1.85
BONECORT	Cortical bone	1.85
MUSCLESK	Skeletal muscle	1.04
MUSCLEST	Striated muscle	1.04
ADTISSUE	Adipose tissue	0.92
KAPTON	Kapton polyimide film	1.42
POLYETHY	Polyethylene	0.94
AIR	Dry air at NTP conditions	0.00120479



The input at a glance



RANDOMIZ

Initialization of random number sequence ("seed")

- Allows using different random sequences as needed when several simulations are run on several CPU in parallel
- Flair takes care of the using different "random seeds" when spawning runs

START

Definition of number of primary particles

- Starts the simulation
- Results returned as average over the cascades induced by the given number of primary particles

STOP

Stop the execution of the program

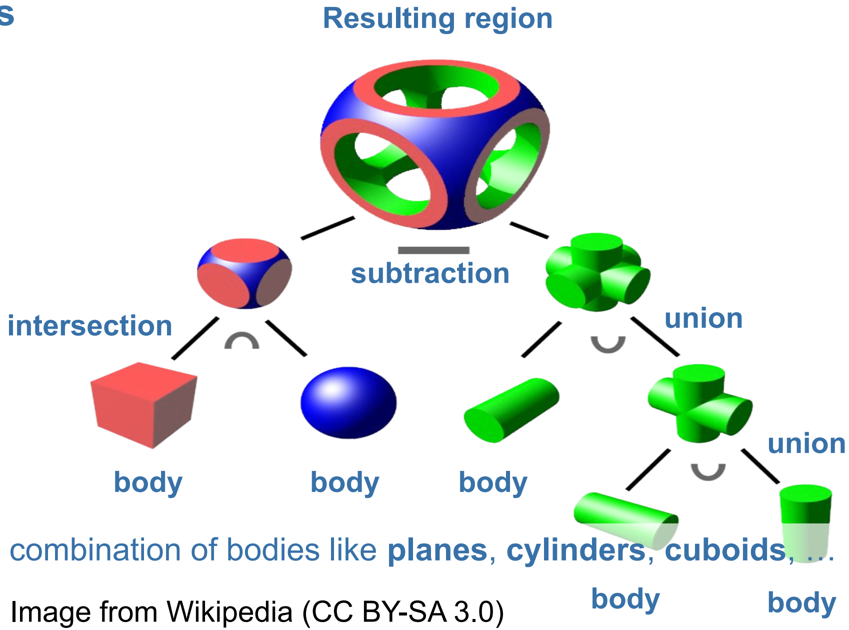
- Not really mandatory (program stops at the end of the input)
- Can become handy for debugging purposes

Combinatorial Geometry

Principle of combinatorial geometry

- Basic objects called **bodies** (such as cylinders, spheres, parallelepipeds, etc.) are combined to form more complex objects called **regions**
- This combination is done using **Boolean operations**

Operation	Math	FLUKA
Union	U	|
Intersection	∩	+
Subtraction	∩ ^	-



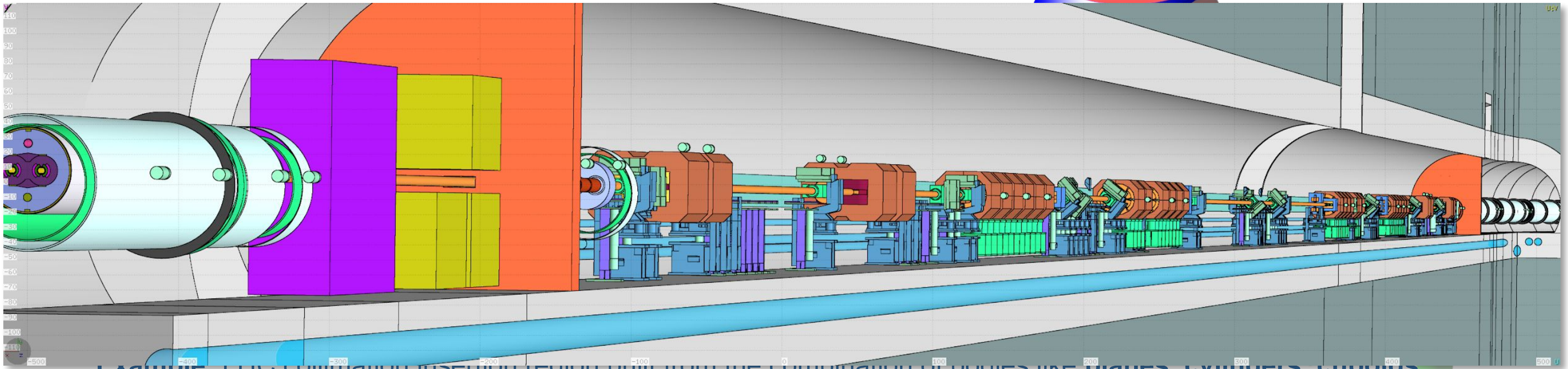
Example: LHC collimation insertion region built from the combination of bodies like **planes**, **cylinders**, **cuboids**, …

Image from Wikipedia (CC BY-SA 3.0)

Principle of combinatorial geometry

- Basic objects called **bodies** (such as cylinders, spheres, parallelepipeds, etc.) are combined to form more complex objects called **regions**
- This combination is done using **Boolean operations**

Resulting region



Example: LHC collimation insertion region built from the combination of bodies like **planes**, **cylinders**, **cuboids**, ...

Image from Wikipedia (CC BY-SA 3.0)

body **body**

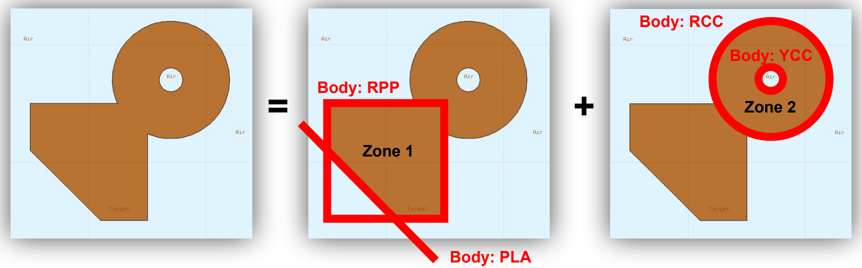
Bodies and regions

Following ***bodies*** are available in FLUKA:

- **Planes**
  - **XYP**, **XZP**, **YZP**: Infinite half space delimited by a coordinate plane
  - **PLA**: Generic infinite half-space, delimited by a **PLA**ne
- **Boxes**
  - **RPP**: **R**ectangular **P**aralleleP**iped
- **Sphere and spheroid**
  - **SPH**: **SPH**ere
  - **ELL**: **ELL**ipsoid of revolution
- **Cylinders and cones**
  - **XCC**, **YCC**, **ZCC**: Infinite **C**ircular **C**ylinder, parallel to coordinate axis
  - **RCC**: **R**ight **C**ircular **C**ylinder
  - **XEC**, **YEC**, **ZEC**: Infinite **E**lliptical **C**ylinder, parallel to coordinate axis
  - **REC**: **R**ight **E**lliptical **C**ylinder
  - **TRC**: **T**runcated **R**ight angle **C**one
- **Other**
  - **QUA**: **QUA**dric

Regions are defined by combining FLUKA bodies using Boolean operations:

- Regions are obtained by the **union of sub-regions** (called zones); in the simplest case a region consists of a single zone
- Zones are defined by **intersections** and/or **subtractions** of **bodies** (Boolean zone expressions)

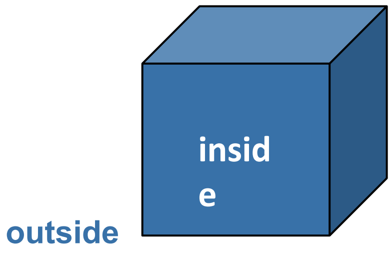


- Zones / regions must be finite
- Each point in space must belong to one (and only one) region
- Regions are of **homogeneous material composition** (i.e. only one material can be assigned to a region)

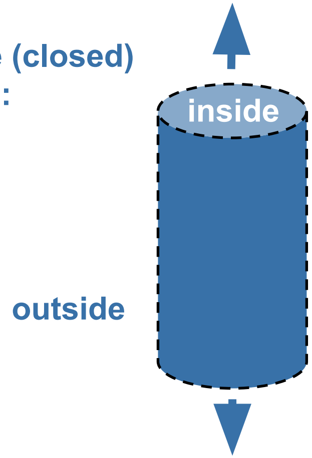
Inside and outside a body

- Each body splits the space into two domains: **inside** and **outside**
  - This concept will be later used when defining zones and regions
    - **+body** refers to the volume **inside** of the body
    - **-body** refers to the volume **outside** of the body
  - The concept of inside and outside is applied to all bodies including infinite planes

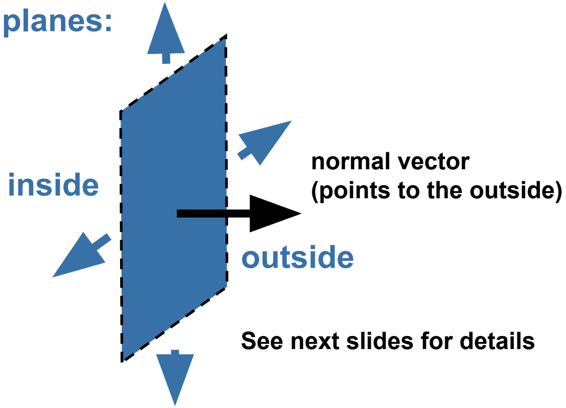
Finite bodies:



Infinite (closed) bodies:



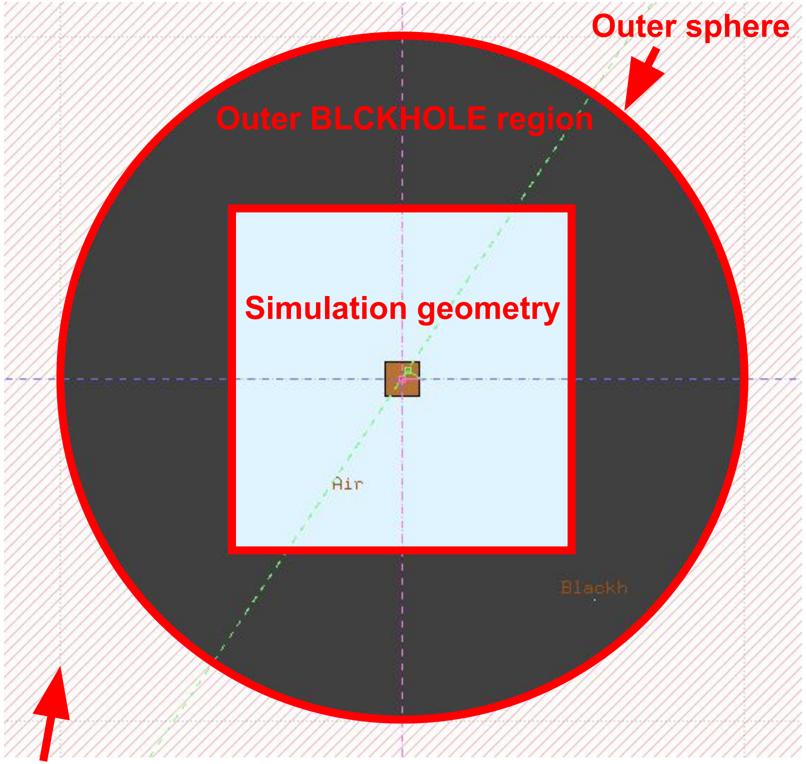
Infinite planes:



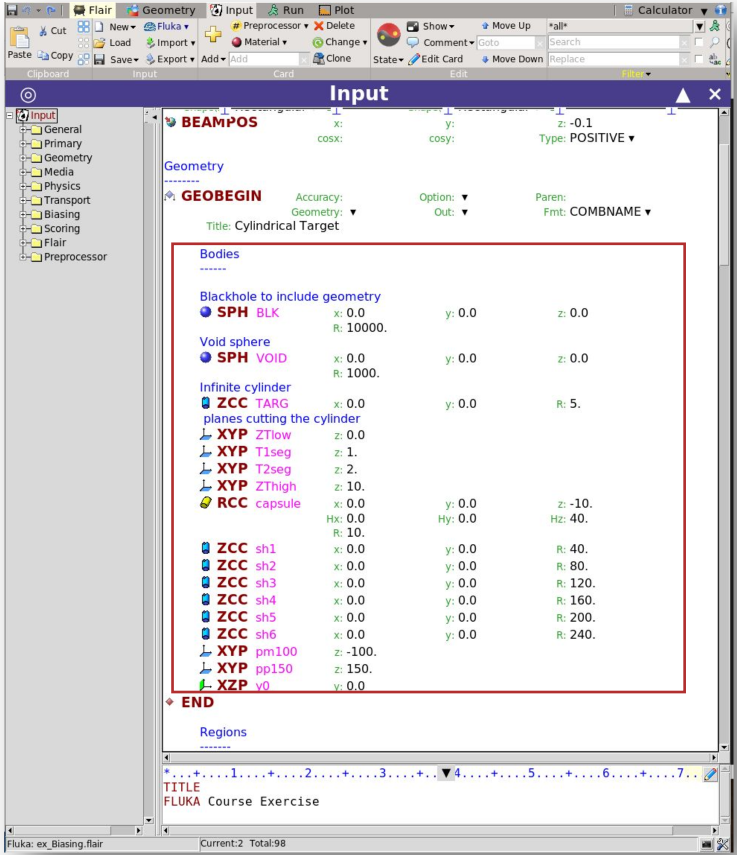
See next slides for details

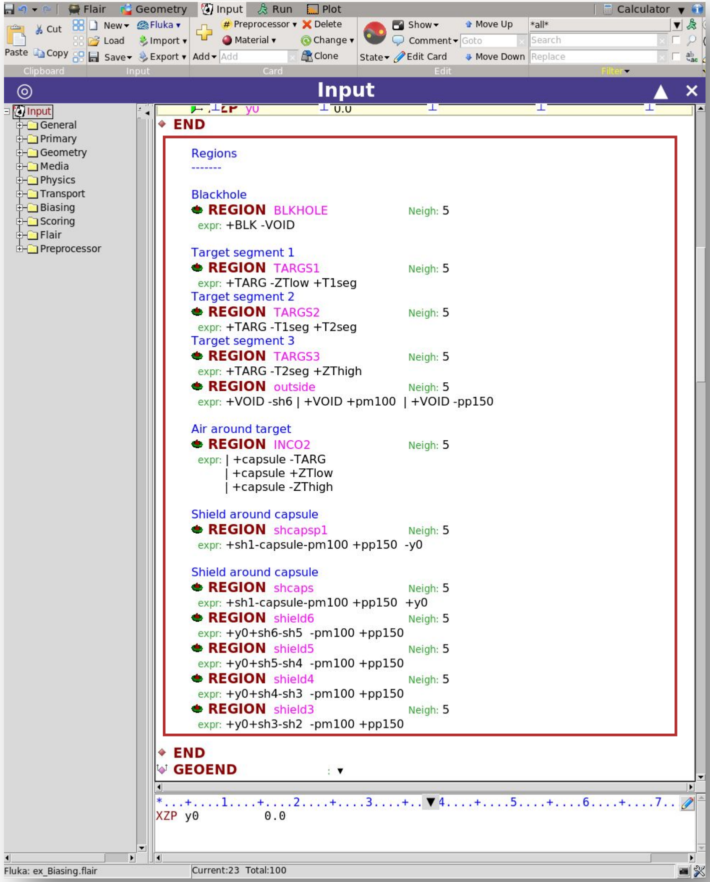
The outer “black hole” confinement

- FLUKA defines a special material called BLCKHOLE:
  - BLCKHOLE is an **all-absorbing material**
  - Particles vanish when entering a region filled with BLCKHOLE
- The entire geometry must be embedded in a region filled with BLCKHOLE
  - This avoids tracking particles to infinity
  - The outer surface of this BLCKHOLE region must be a single closed body (e.g. a sphere)



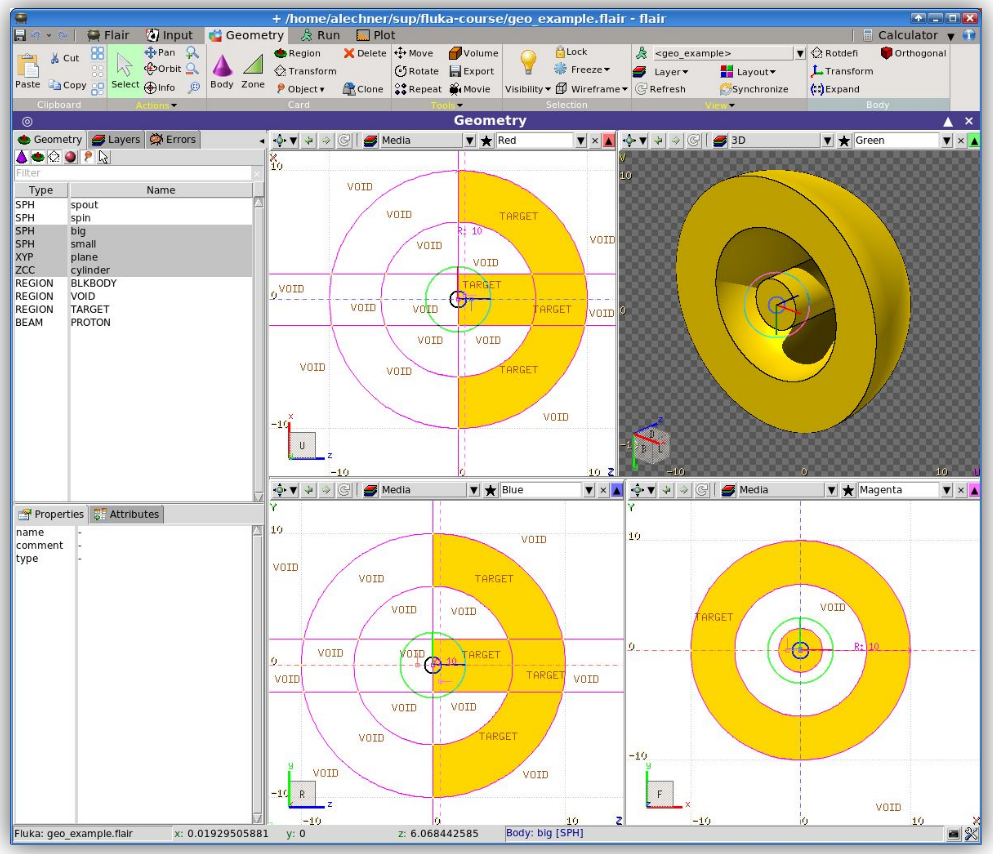
Geometry input in Flair





Flair geometry editor

Allows creating geometries at a (few) mouse click(s)…



Scoring of physical quantities

FLUKA scoring

- It is said that Monte Carlo (MC) is a **"mathematical experiment"**; the MC equivalent of the result of a real experiment (*i.e.*, of a **measurement**) is called an **estimator**

- Just as a real measurement, an estimator is obtained by sampling from a statistical distribution and has a **statistical error** (and in general also a **systematic** one)

- There are often several different techniques to measure the same physical quantity: in the same way, **the same quantity can also be calculated using different kinds of estimators**

- FLUKA offers **numerous different estimators**, *i.e.* **scoring** for various quantities of interest can be requested directly from the input file

Built-in and user scoring

- Several **pre-defined estimators** can be activated in FLUKA; one usually refers to these estimators as **built-in scoring** capabilities
- Users may build their own custom scoring through **user routines**
- However, **built-in scoring**:
  - covers most **common needs**
  - has been **extensively tested**
  - takes **biasing automatically into account**
  - has **refined algorithms** for track subdivision (apportioning)
  - comes with **utility programs** that allow to evaluate statistical errors
- Therefore users are strongly encouraged to **prefer built-in scorings** wherever possible
- Standard scoring can be adapted by means of simple user routines (`fluscw.f`, `comscw.f`), activated via `USERWEIG` card

FLUKA scoring

What?

Energy deposition and derivatives (dose), fluence or current versus energy, angle or other kinematic variables, time, DPA, residual activity…

Where?

In regions, across boundaries, on region-independent grids

When?

At the end of each cycle or at each event

Output?

Saved in `[inputname]nnn_fort.##` files, where `nnn` is the cycle number & `##` is the logical unit number chosen by the user

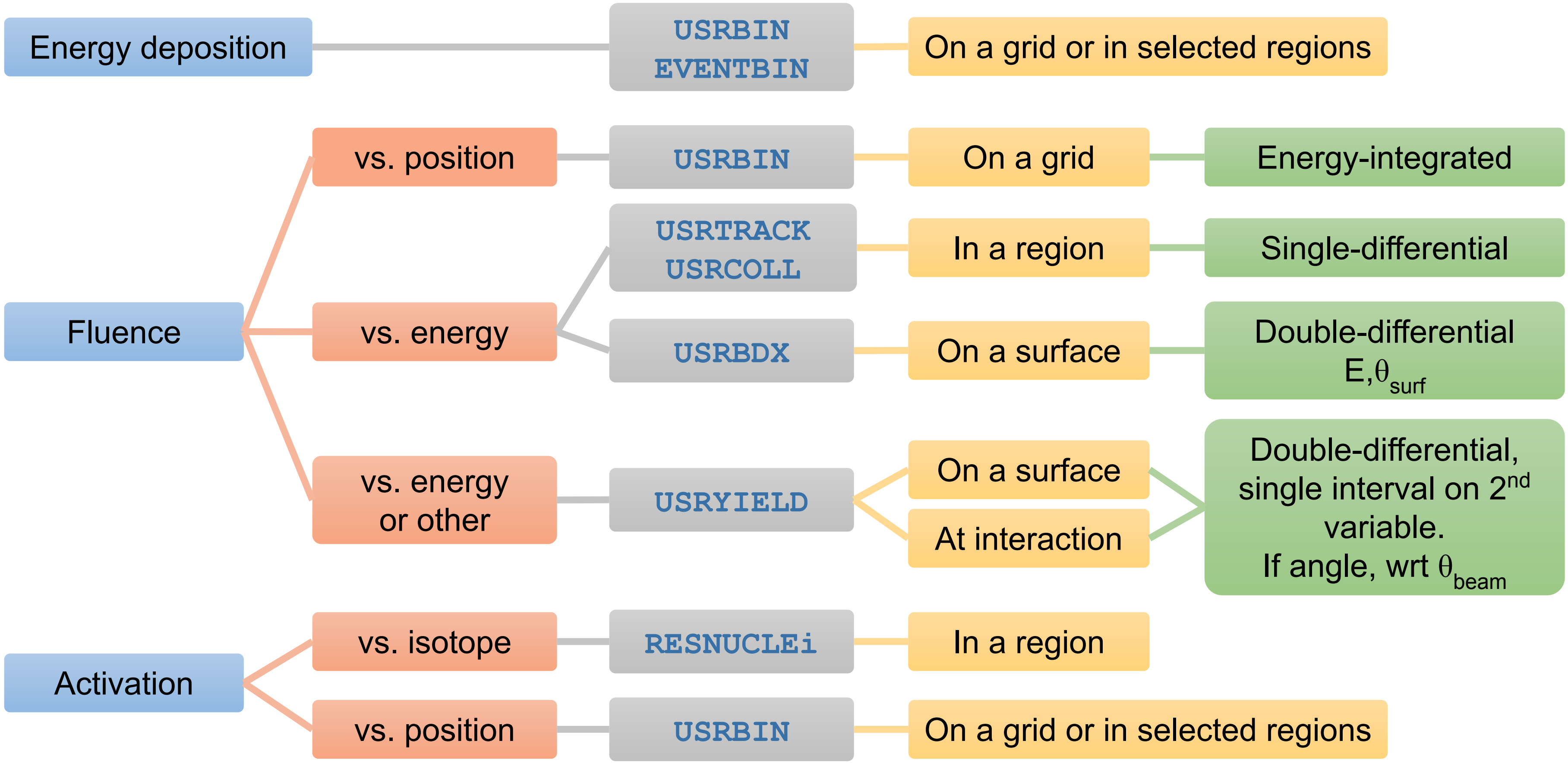
Results?

Post-processing utilities merge cycles, calculate average and rms, provide data files for plotting. Available via **Flair**

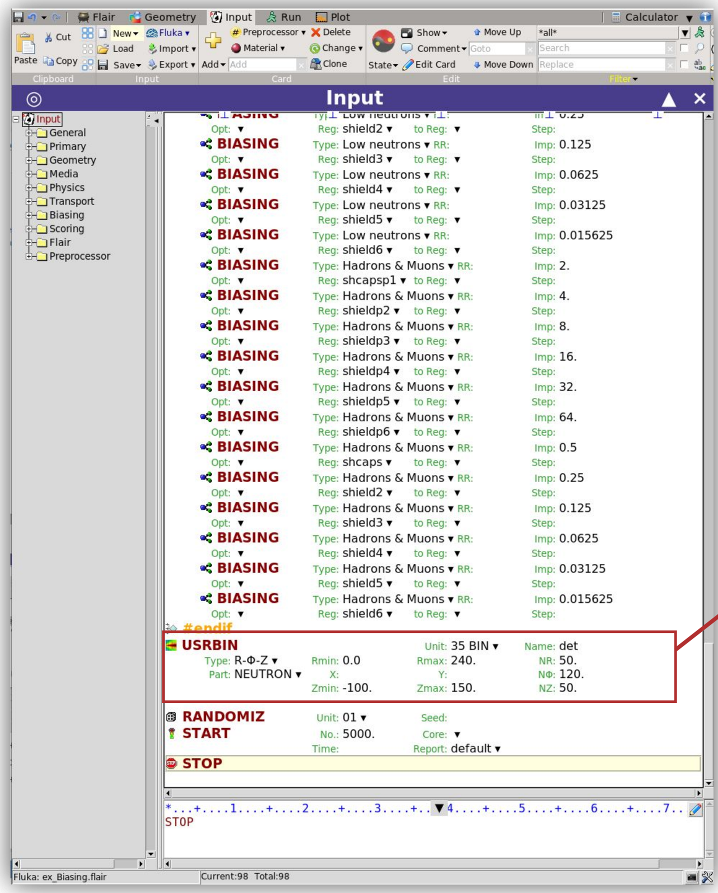
Results normalised **per primary**

User code needed for processing of custom scoring!

The FLUKA estimator zoo



A scoring example



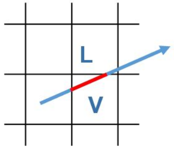
USRBIN

Scores distributions of one of several quantities in a regular spatial structure (mesh) independent from the geometry or on a region basis.

Here: neutron fluence in a cylindrical mesh around beam axis

- R: 0 - 240cm in 50 bins
- z: -100cm to 150cm in 50 bins
- Phi: 120 bins

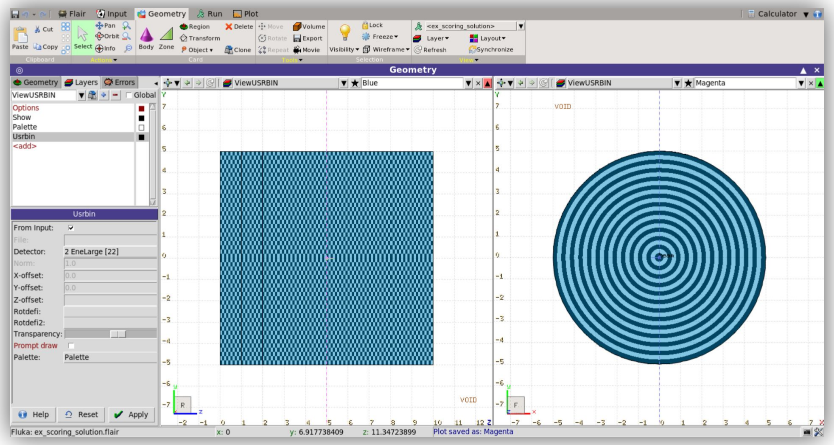
Results in units of 1/cm^2 per primary particle



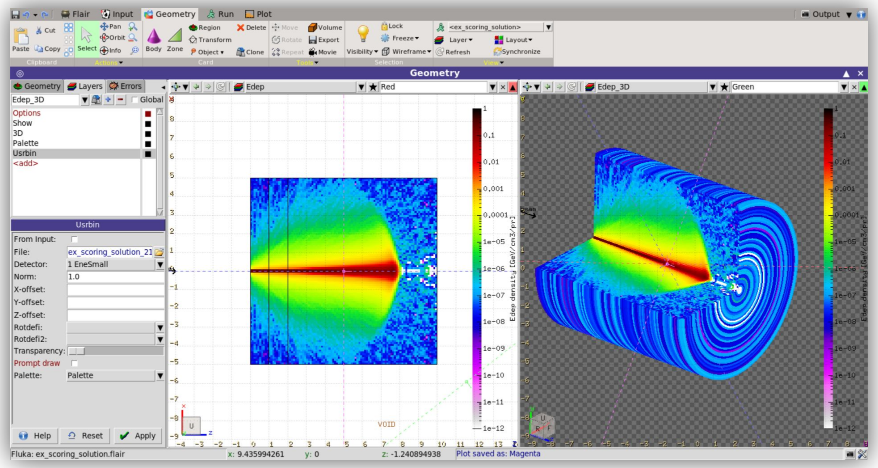
Path length **L** [cm] inside the bin divided by the bin volume **V** [cm^3] → cm / cm^3 = 1/cm^2

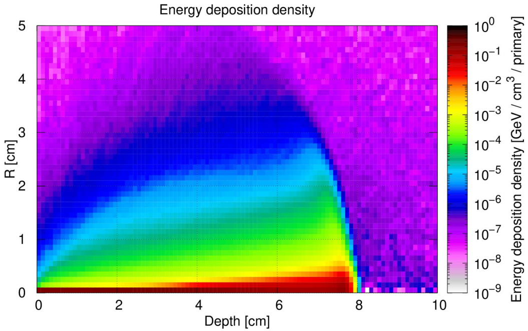
Visualization with Flair

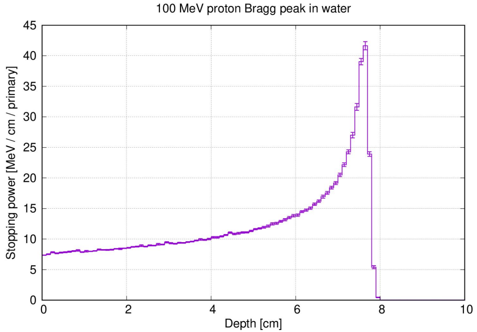
…of the scoring mesh



and the final results (examples)



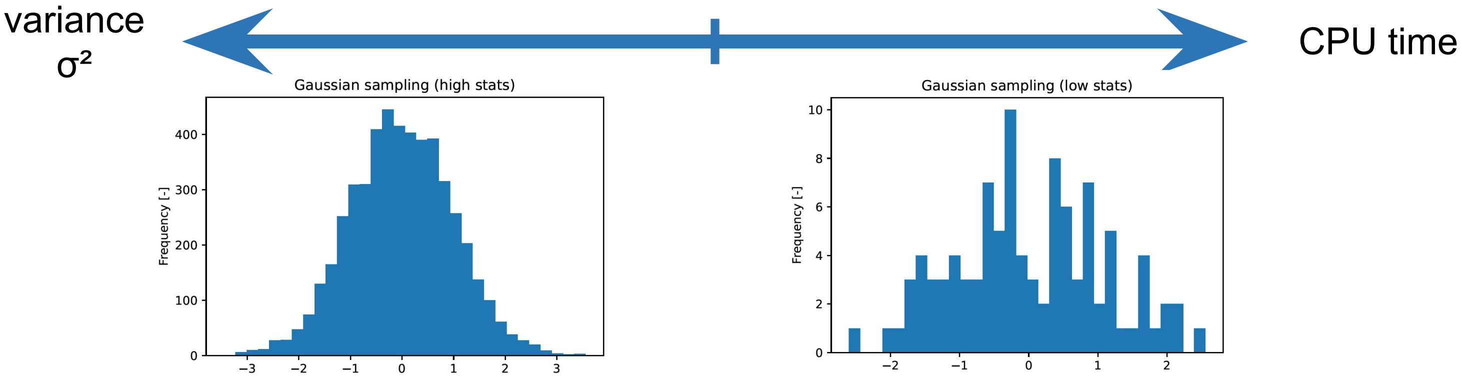




Biasing techniques

Introduction to biasing

- Statistical bias: tendency causing a result to differ from the underlying fact
- In the context of FLUKA
  - Deliberately altering simulation parameters to improve variance or CPU time
  - This bias is countered by changing weights of particles



- Goodness of simulations : Figure of Merit = $\frac{1}{\sigma^2 t}$
  - The larger the better

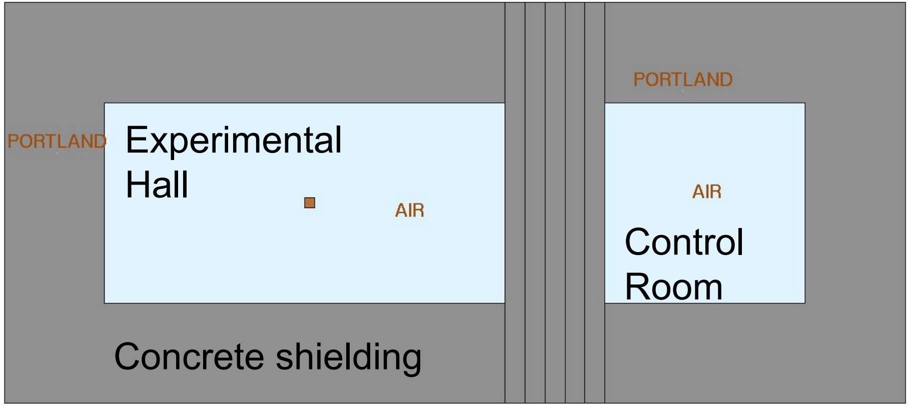
Non-biased Monte Carlo simulations

Characteristics

- Samples from
  actual phase-space distributions
- Preserves correlations
- Reproduces fluctuations

Drawbacks

- Converges slowly
- Rare events are… “rare”



Non-biased Monte Carlo simulations

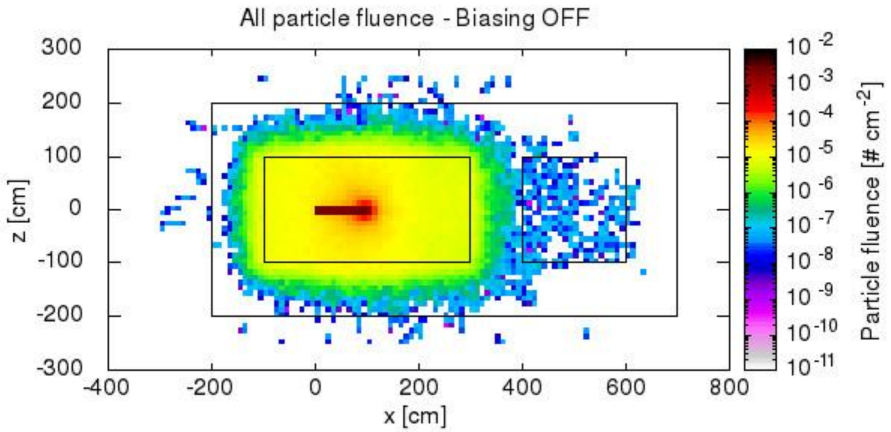
200000

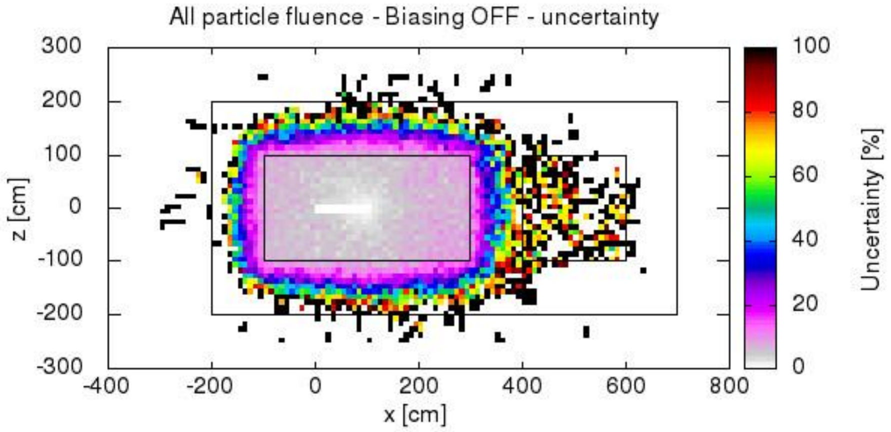
Characteristics

- Samples uniformly from
  the phase-space distribution
- Preserves correlations
- Reproduces fluctuations

Drawbacks

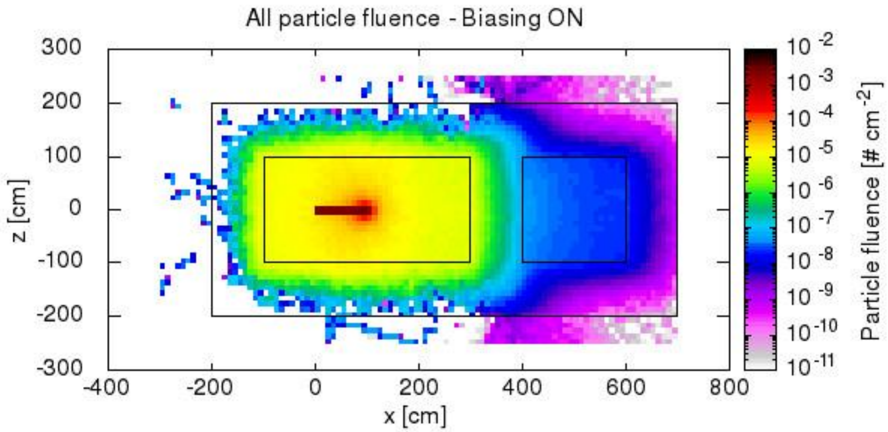
- Converges slowly
- Rare events are… "rare"

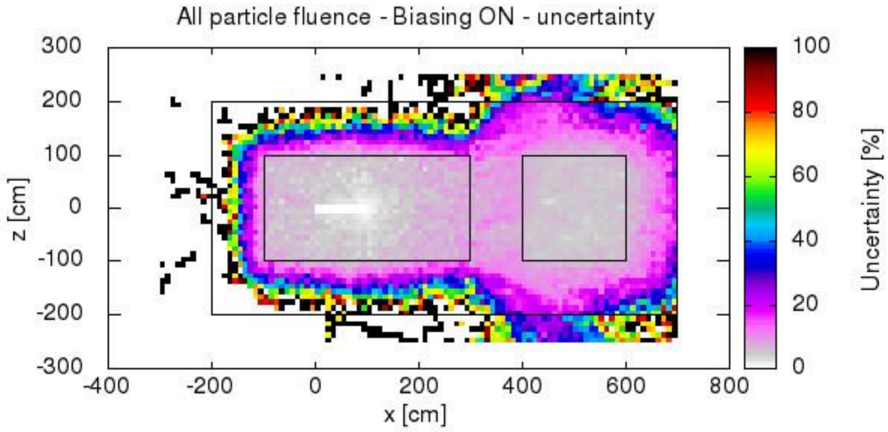




Biased Monte Carlo simulations

200000





Characteristics

- Samples from distorted distributions
- Converges "quickly"

Drawbacks

- Cannot reproduce fluctuations and correlations
- Requires active reasoning and experience
- Requires user's time to be implemented

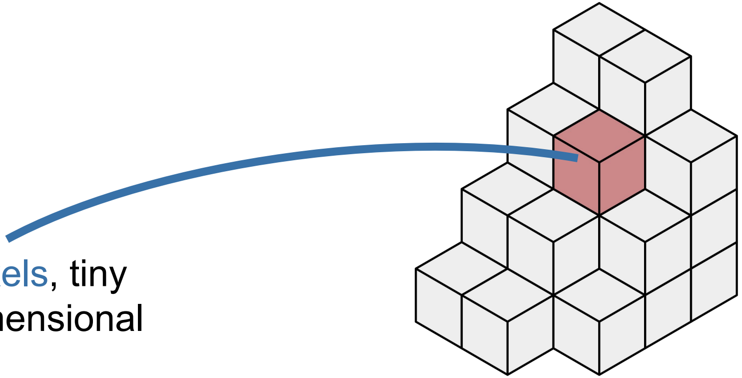
Biasing techniques in FLUKA

- *Region Importance Biasing* (**`BIASING`**)
- *Mean Free Path Biasing* (**`LAM-BIAS`**)
- Leading Particle Biasing (**`EMF-BIAS`**)
- Multiplicity Tuning (**`BIASING`**)
- Lifetime / Decay-length Biasing (**`LAM-BIAS`**)
- Weight Windows (**`WW-FACTO`**, **`WW-THRES`**, **`WW-PROFI`**)
- Low-energy neutrons non-analogue absorption (**`LOW-BIAS`**)
- Low-energy neutrons downscattering (**`LOW-DOWN`**)
- User defined biasing (`usbset.f` , `usimbs.f`)

Medical applications

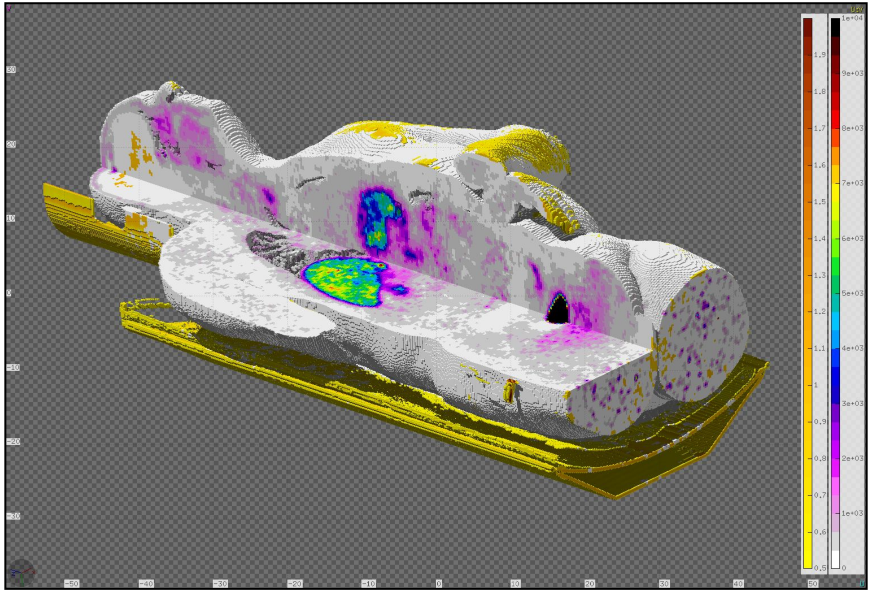
(…a short
“teaser”)

Voxel geometries



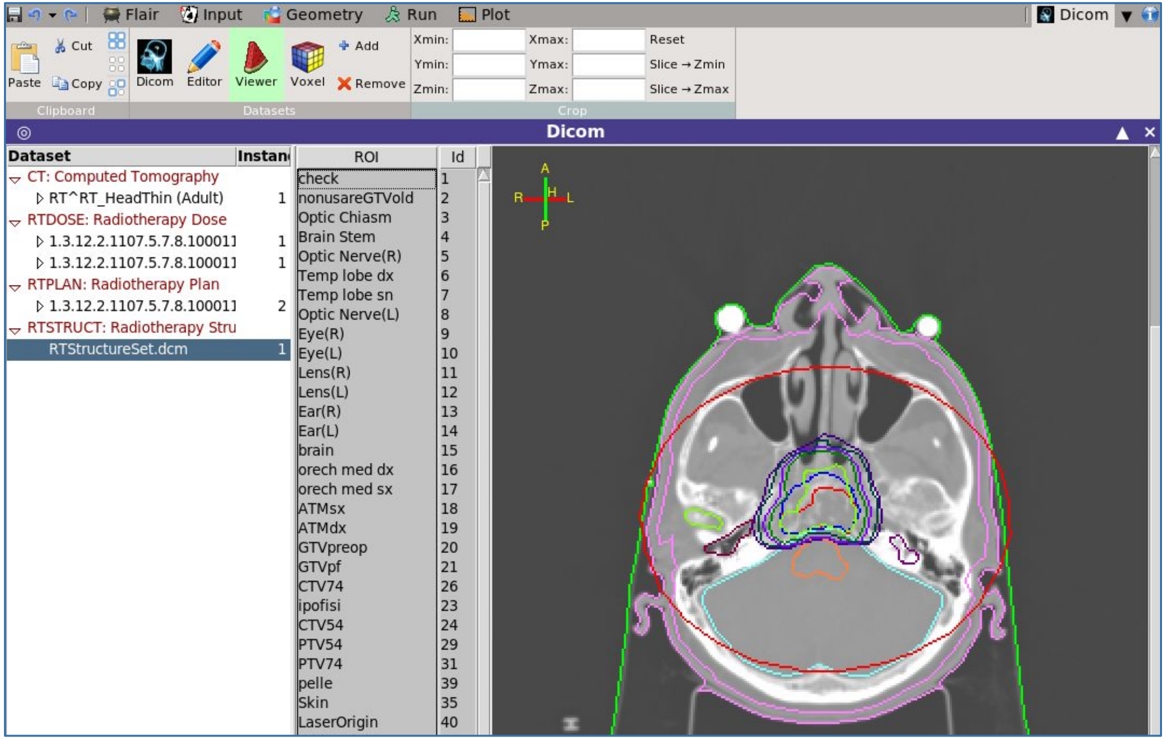
- A geometry can be described in terms of voxels, tiny parallelepipeds of equal size forming a 3-dimensional grid
- Voxel geometries are especially useful for importing CT scans, e.g. for dosimetric calculations of radiotherapy treatments
- Flair can process CT scans in the DICOM(*) format using the pydicom module and convert them to FLUKA voxel geometries or USRBIN-compatible files

(*) DICOM (Digital Imaging and Communications in Medicine) is a medical standard for distributing any kind of medical image.



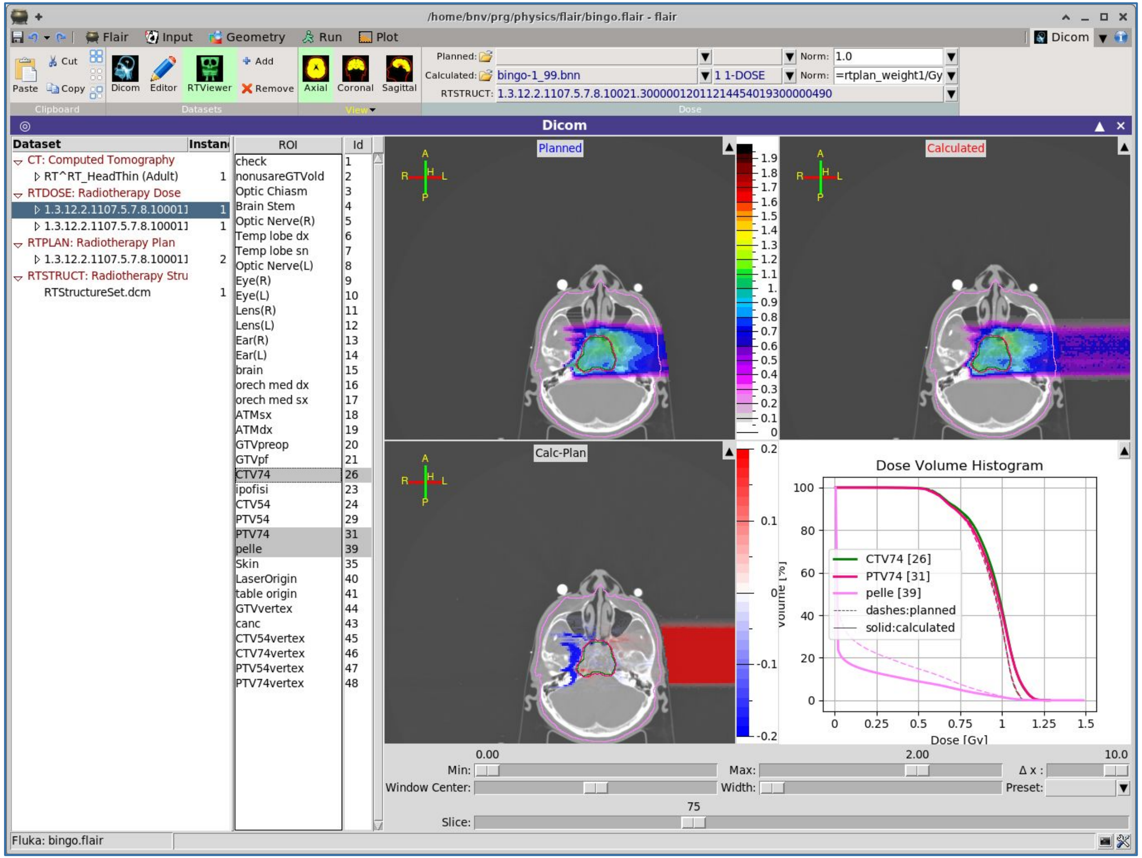
Defining organs

- DICOM files can be browsed, visualised and edited (e.g. anonymised)
- Voxels can be grouped into “organs”
- ROIs (Regions Of Interest) can be defined
- The voxel geometry is contained in an RPP and can be placed within a larger combinatorial FLUKA geometry



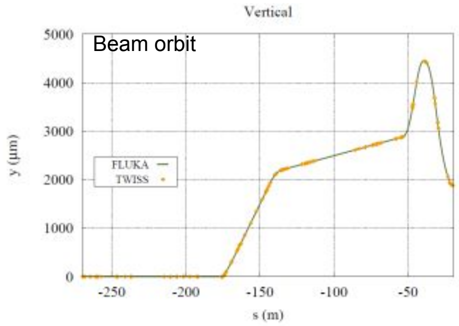
Calculating dose to organs

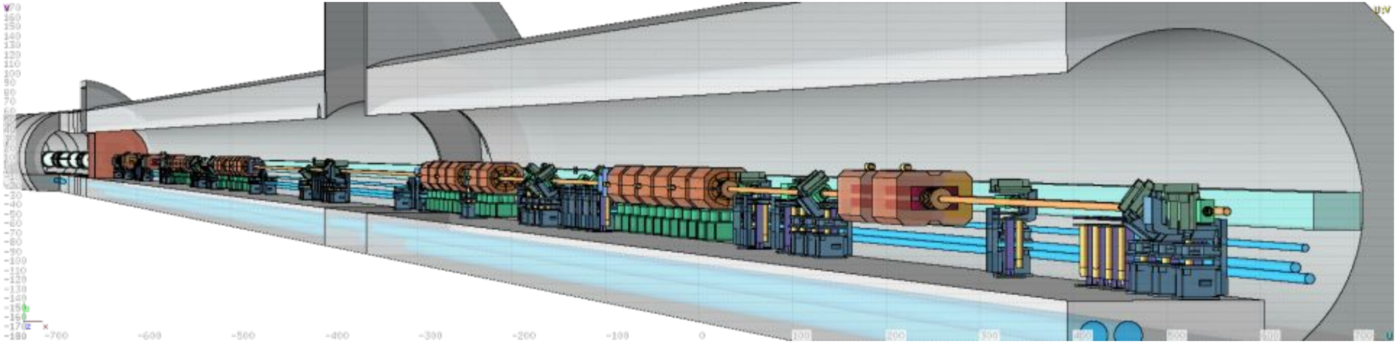
- Correction factors for the density and dE/dx can be specified
- The RTPLAN can be converted to a FLUKA input
- RTDOSE: the calculated data can be compared to the planned dose
- Automatic generation of DVH (Dose Volume Histogram)
- Relevant cards: **`VOXELS`**, **`CORRFACT`**, **`RAD-BIOL`**, **`TPSSCORE`**

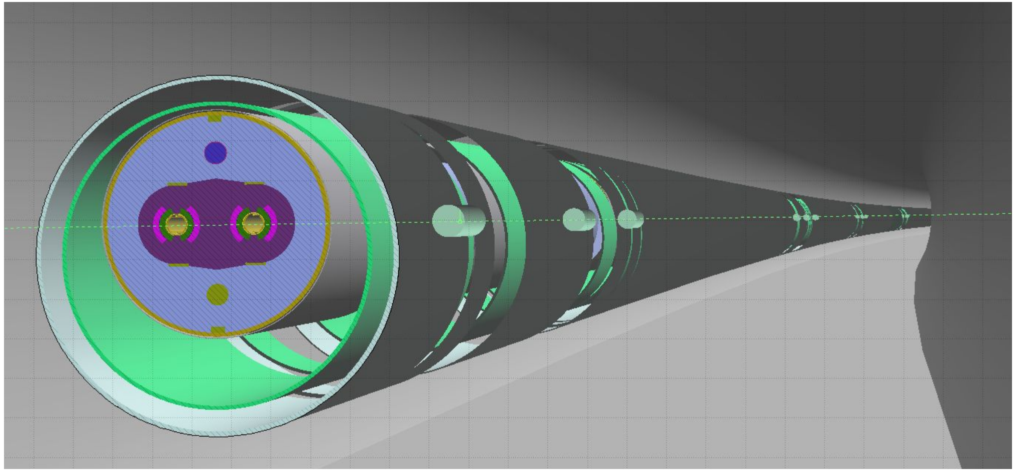


Some examples

Accelerator geometries







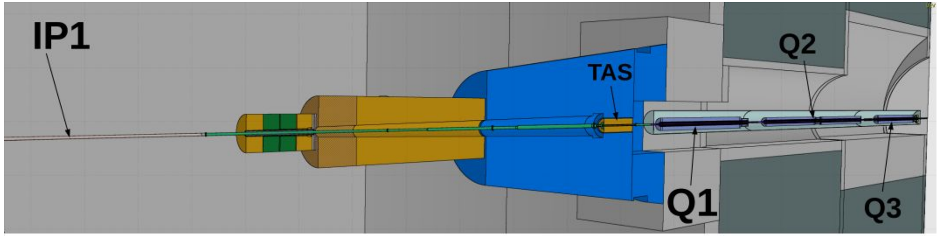
From DETAILED MODELS OF ACCELERATOR COMPONENTS WITH ASSOCIATED SCORING and the ELEMENT SEQUENCE AND RESPECTIVE MAGNETIC STRENGTHS, as given IN THE MACHINE OPTICS (TWISS) FILES,
the **AUTOMATIC CONSTRUCTION OF COMPLEX BEAM LINES**, including collimator settings and element displacement (BLMs), is achievable, profiting from roto-translation directives and replication (lattice) capabilities.

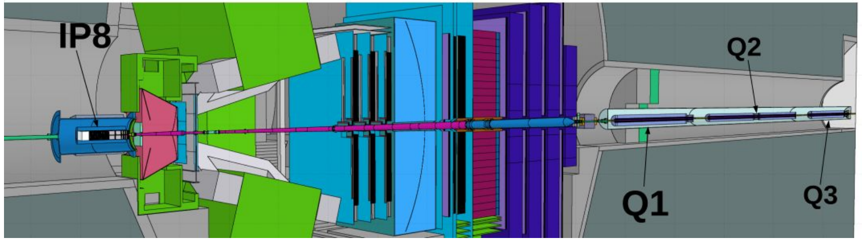
LINE BUILDER

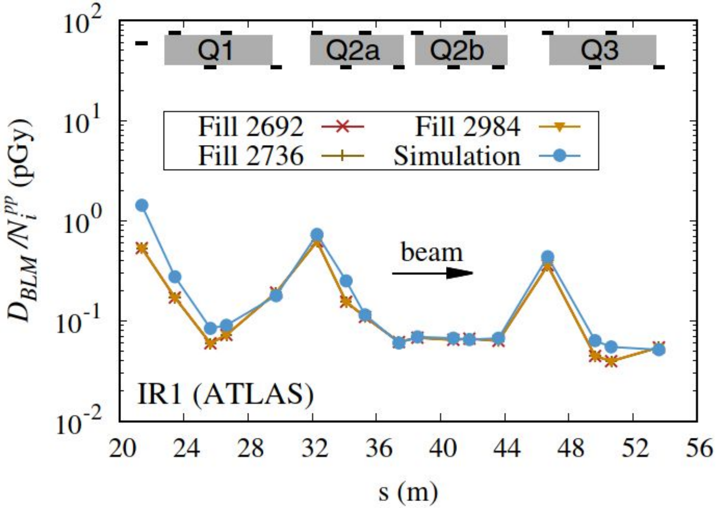
[A. Mereghetti et al., IPAC2012, WEPPD071, 2687]

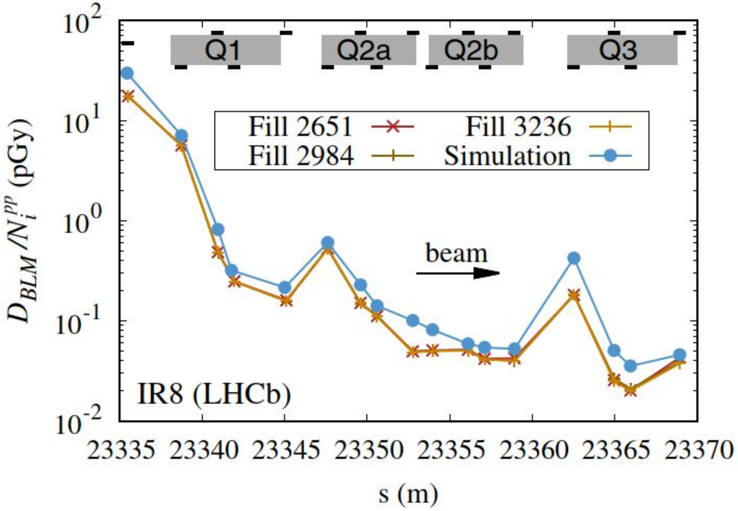
Beam loss description at the LHC

[A. Lechner et al., Phys. Rev. AB 22 (2019) 071003]









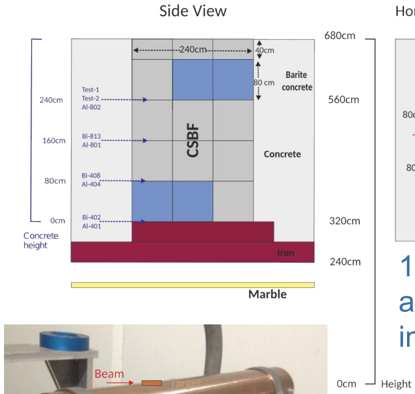
Activation benchmarking

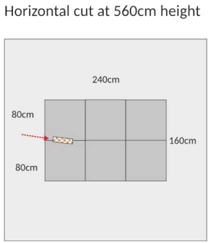
@ CERN SHIELDING BENCHMARK FACILITY (24 GeV/c p)

[E. Iliopoulou and R. Froeschl]

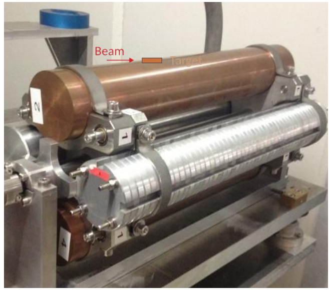
Situated laterally above the CHARM target
for deep shielding penetration studies (Detector calibration, Detector inter-comparison, Activation)

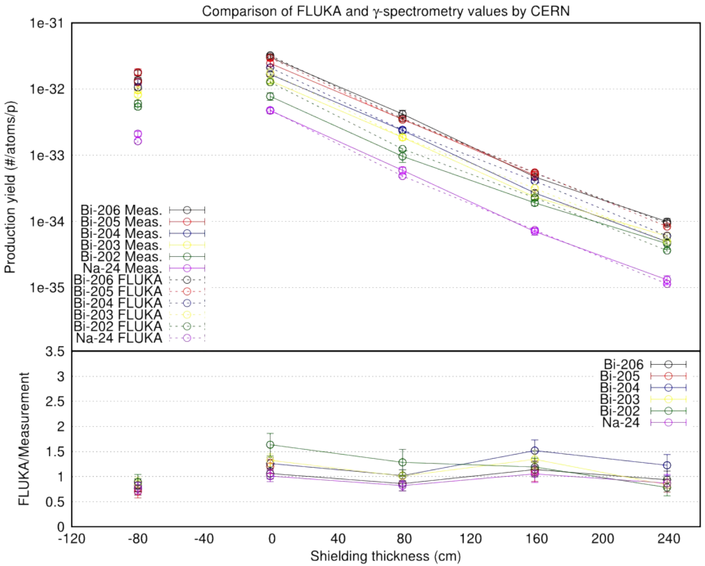
360cm of concrete and barite concrete
plus 80cm of cast iron





11 Bismuth and Aluminum samples at different heights in CSBF and also inside CHARM (@ -80cm)





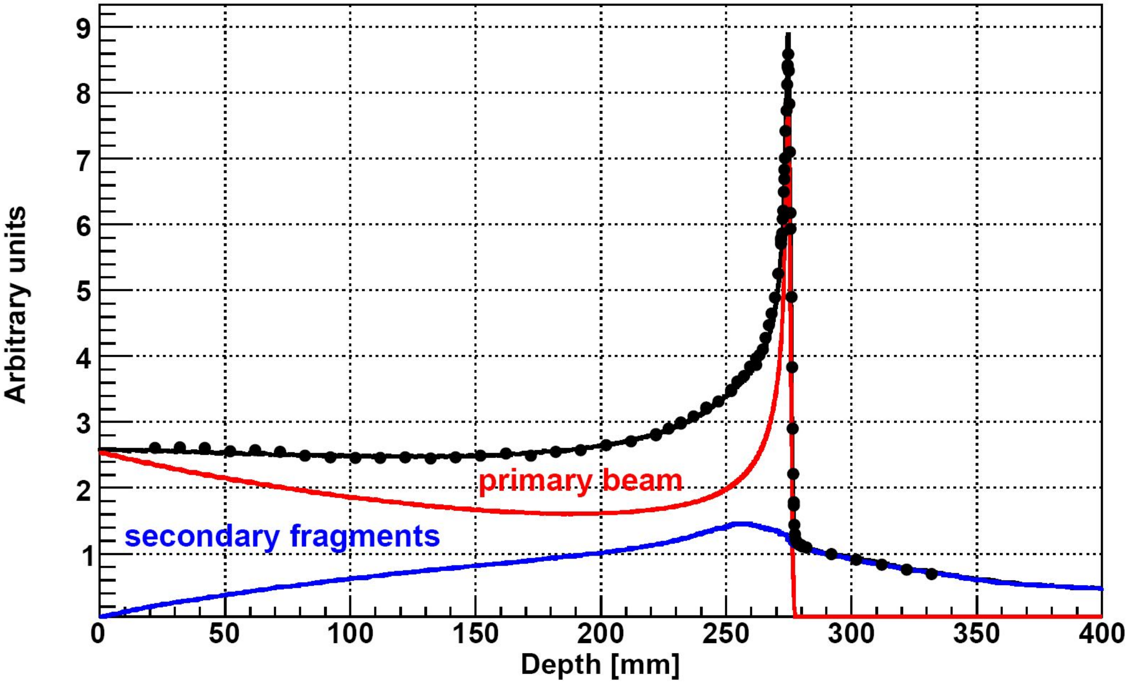
@ CHARM (CERN High energy AcceleRator Mixed field facility,

to study radiation effects on electronic components)

5×10^{11} protons/pulse, 350ms pulse length, max. average beam intensity 6.6×10^{10} p/s
three 50cm long 8cm diameter targets: Copper, Aluminum, Aluminum with holes

R2E

Medical physics: radiotherapy



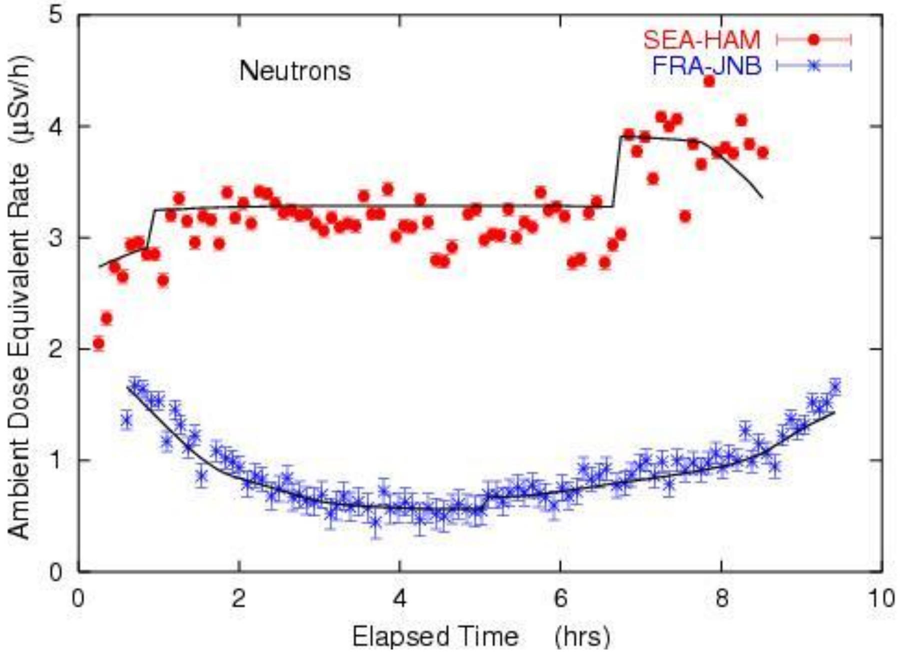
Bragg peak in a water phantom
400 MeV/A C beam:
The importance of fragmentation

[Exp. Data (points) from Haettner et al, Rad. Prot. Dos. 2006
Simulation: A. Mairani PhD Thesis, 2007, Nuovo Cimento C, 31, 2008]

Dosimetry and cosmic rays

- Complete simulation of cosmic rays interactions in the atmosphere, by means of a dedicated CR package available to users
- Model of airplane geometry
- Response of dosimeters

Ambient dose equivalent from neutrons
at solar maximum on commercial flights
from Seattle to Hamburg and from Frankfurt to Johannesburg



[Solid lines: FLUKA simulation
S. Roesler et al.,
Rad. Prot. Dosim. 98 (2002) 367]

FLUKA