



Highlights on Advanced Topics

FLUKA Beginner's Course

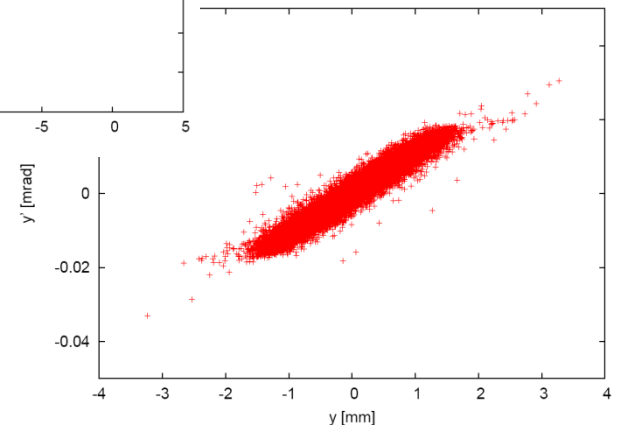
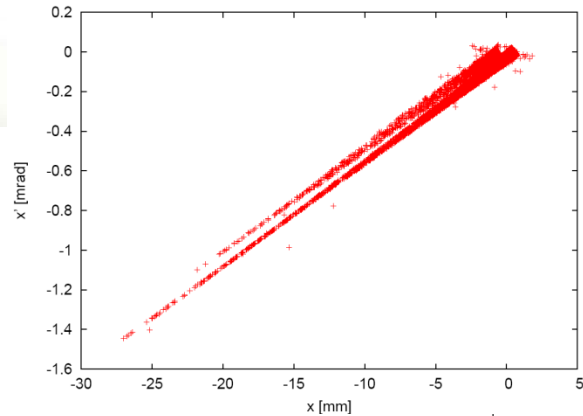
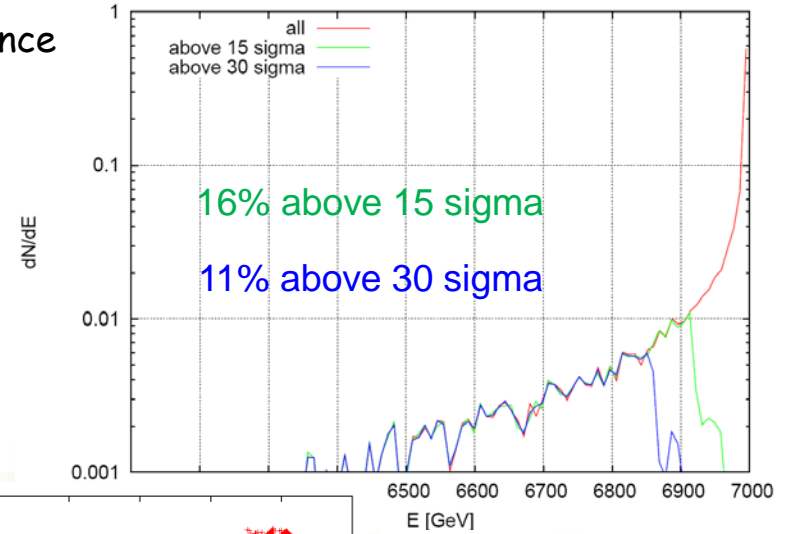
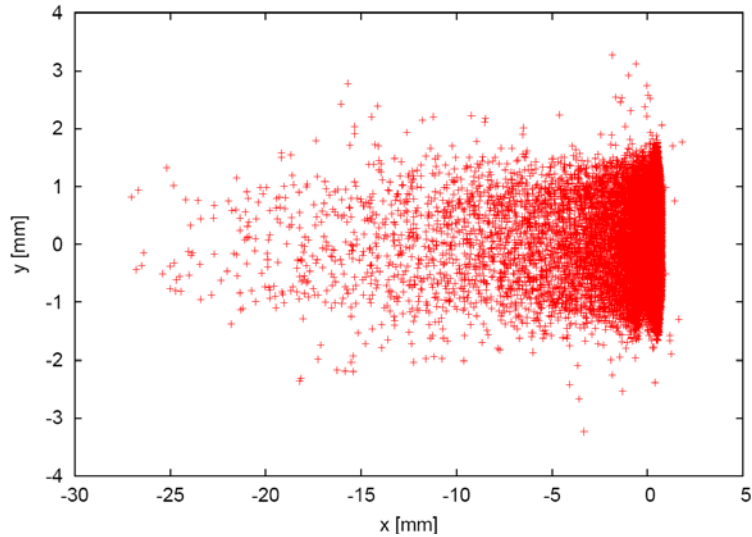
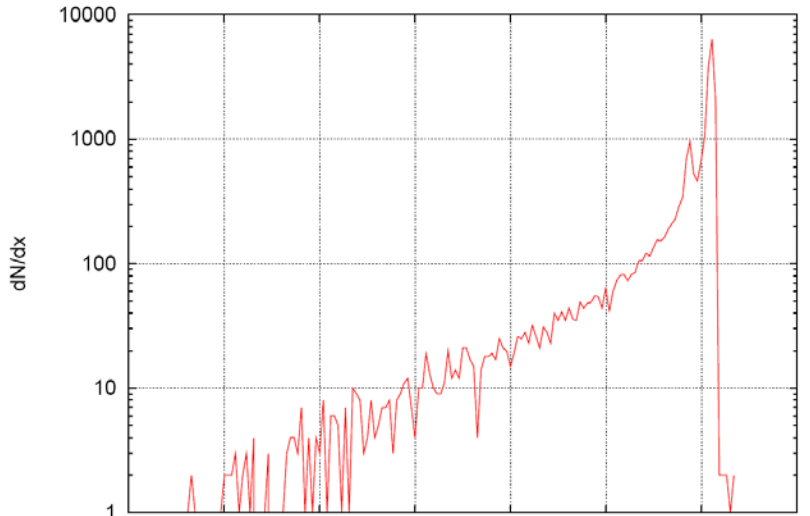
This is not the end ...

- As said in the first day, most applications can be run through data-cards only, exploiting the FLUKA built-in capacities
- Sometimes, more is needed...
- A set of template User Routines is provided in the `$FLUPRO/usermvax` directory
- These can be modified by the user to fit his input/output needs
- All shared parameters and variables (in COMMON blocks) are located in `$FLUPRO/flukapro`
- We'll give here some hints of what can be done, more can be found in the manual, in the fluka-discuss archive, or

at the fourth FLUKA advanced course and workshop
dates and venue not yet defined

Implementing customized beam distributions - 1

LHC proton beam at the Dispersion Suppressor entrance



Implementing customized beam distributions - 2

Input card: **SOURCE**

User routine: [\\$FLUPRO/usermvax/source.f](#)
to be linked in one's own executable !

In source.f

you can sample beam particle position, direction, and energy
from an external file or any (analytic or numerical) distribution

you can assign different weights to primary particles

you can load reaction products in the same primary history

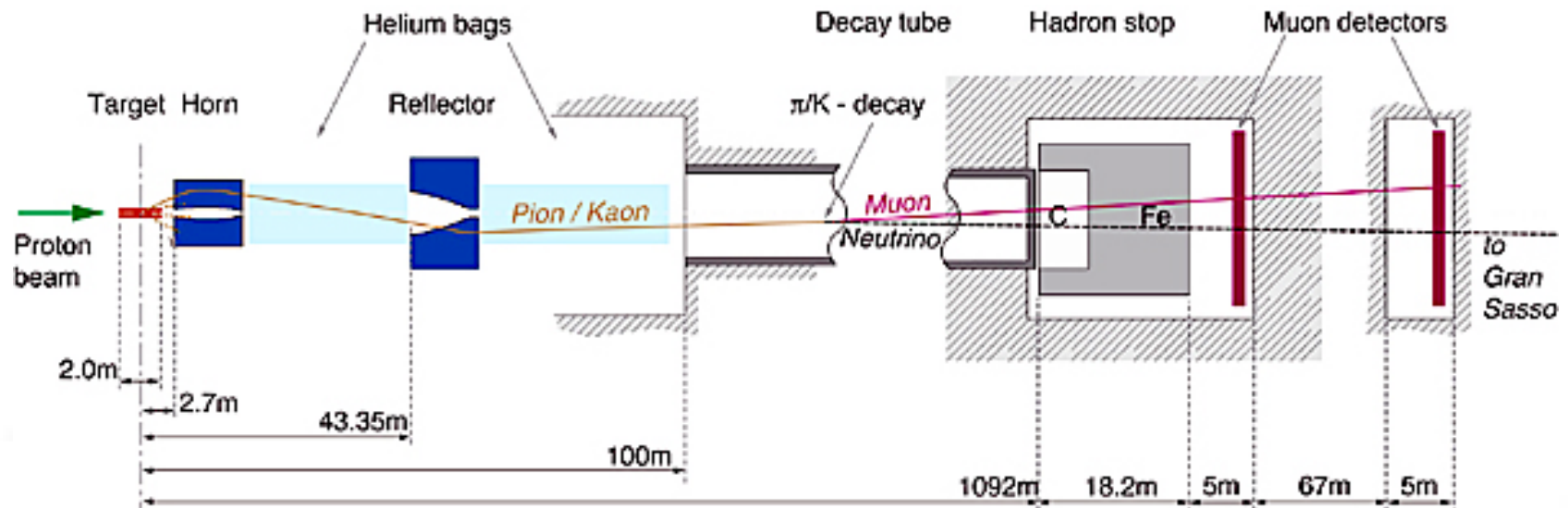
several sampling routines already exist in the FLUKA code

parameters input in the **SOURCE** card are available

a **BEAM** card with a momentum/energy higher than the maximum one is still needed
for initialization purposes (to define the tabulation limit)

Implementing magnetic field - 1

CERN Neutrino to Gran Sasso



The two magnetic lenses (blue in the sketch) align positive mesons towards the Decay tunnel, so that neutrinos from the decay are directed to GranSasso, 730~km away
Negative mesons are deflected away
The lenses have a finite energy/angle acceptance

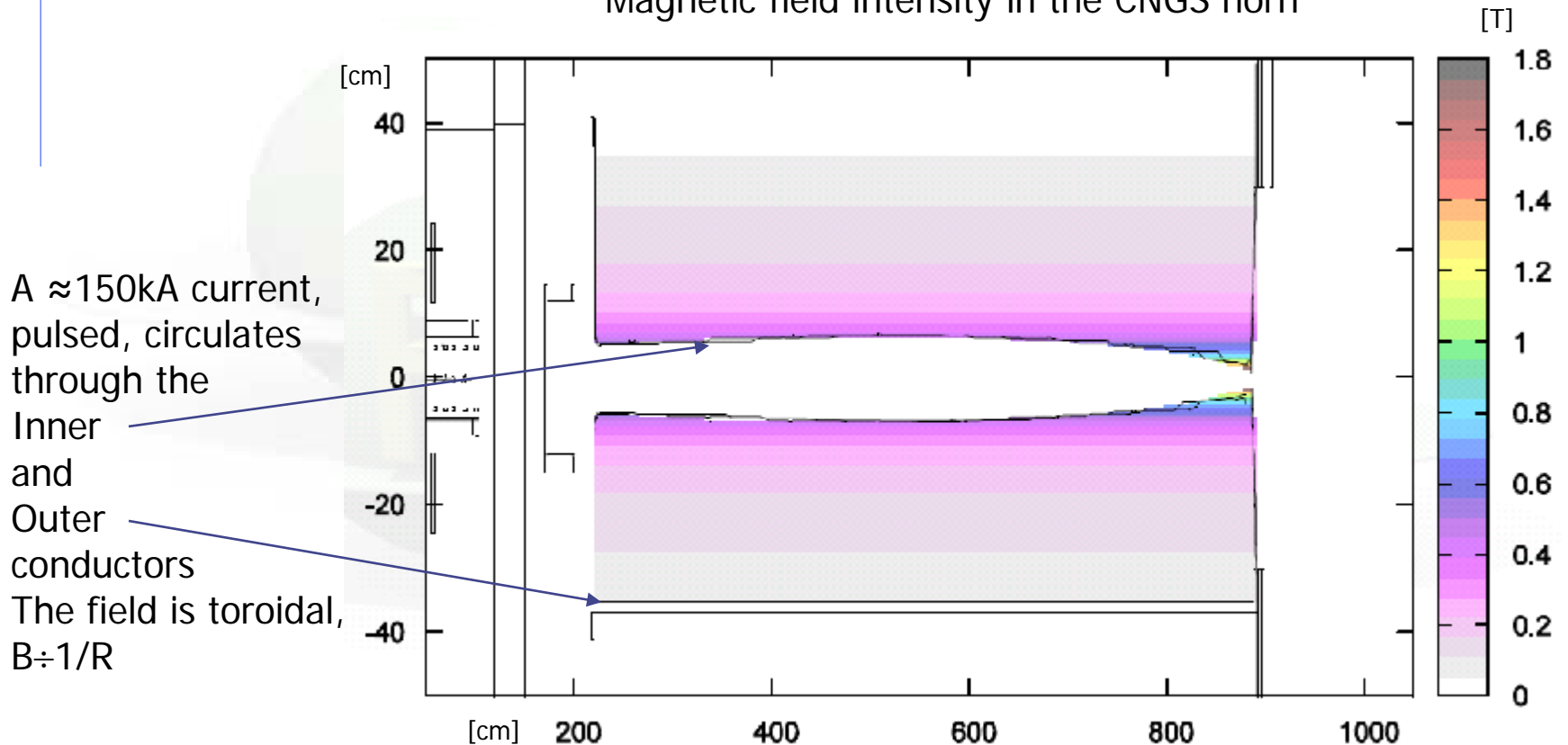
Implementing magnetic field - 2

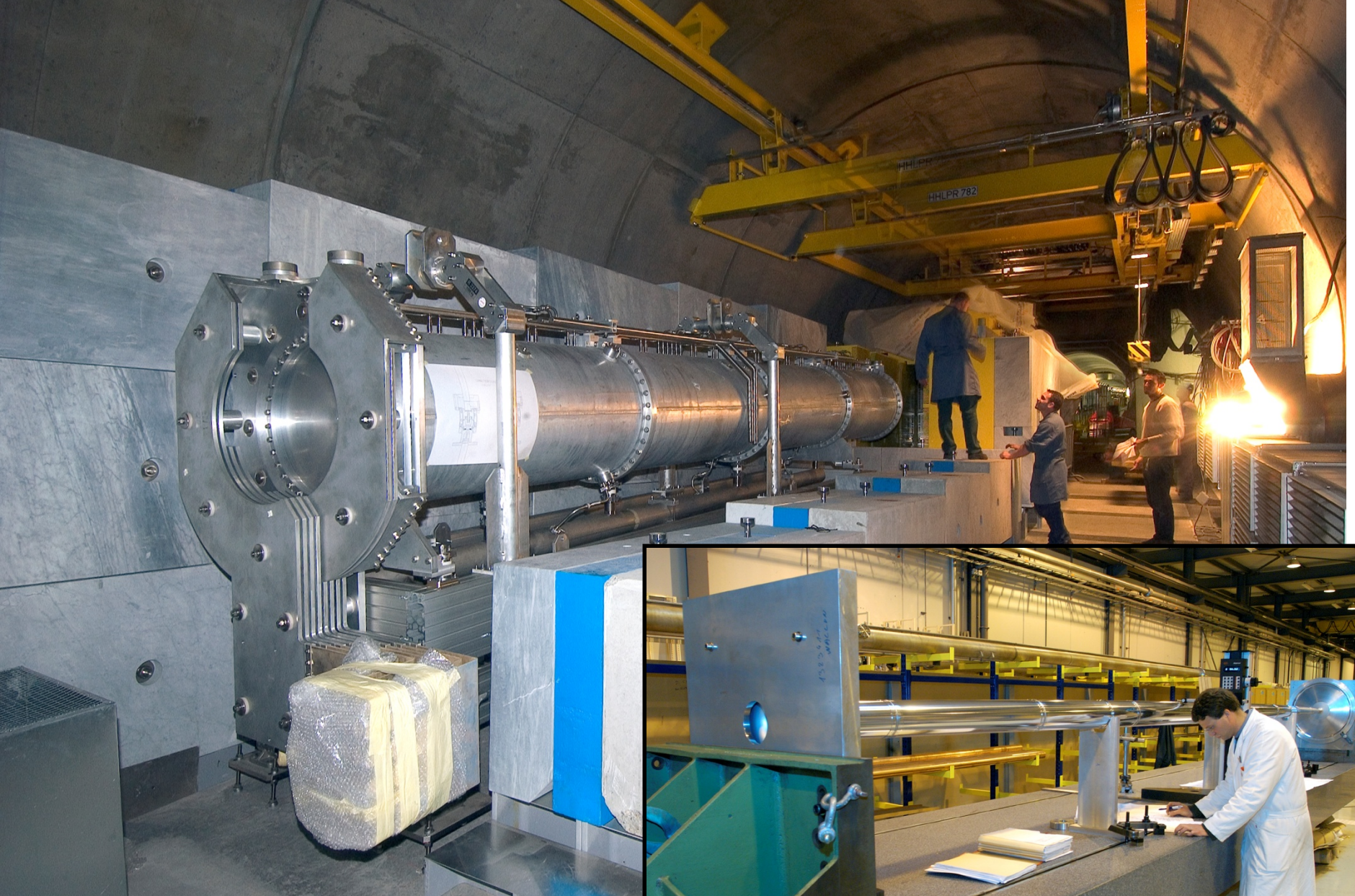
Input card: **ASSIGNMA**

User routine: **`$FLUPRO/usermvax/magfld.f`**

to be linked in one's own executable !

Magnetic field intensity in the CNGS horn





Implementing magnetic field - 4

```
SUBROUTINE MAGFLD ( X, Y, Z, BTX, BTY, BTZ, B, NREG, IDISC )
```

```
IF ( NREG .EQ. NRHORN ) THEN
```

```
  RRR = SQRT ( X**2 + Y**2 )
```

```
  BTX = -Y / RRR
```

```
  BTY = X / RRR
```

```
  BTZ = ZERZER
```

```
  B = 2.D-07 * CURHOR / 1.D-02 / RRR
```

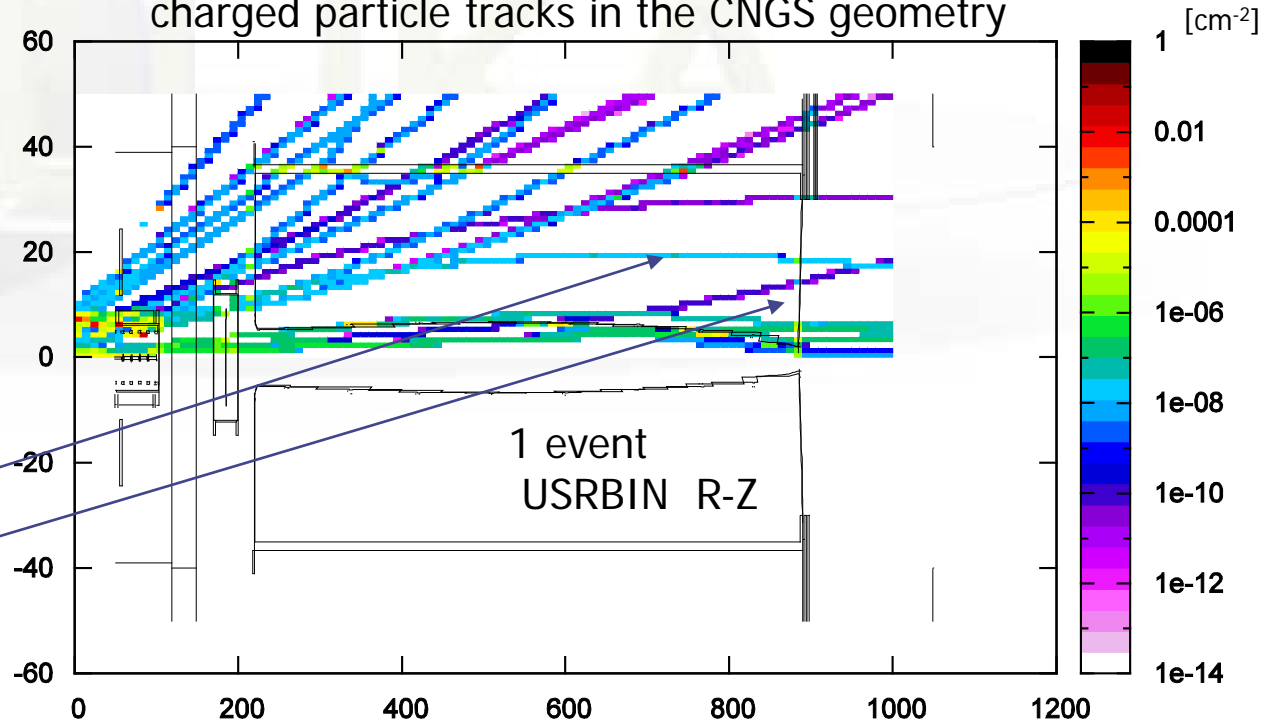
```
END IF
```

In magfld.f

you can define the magnetic field
analytically or
by interpolating an external map



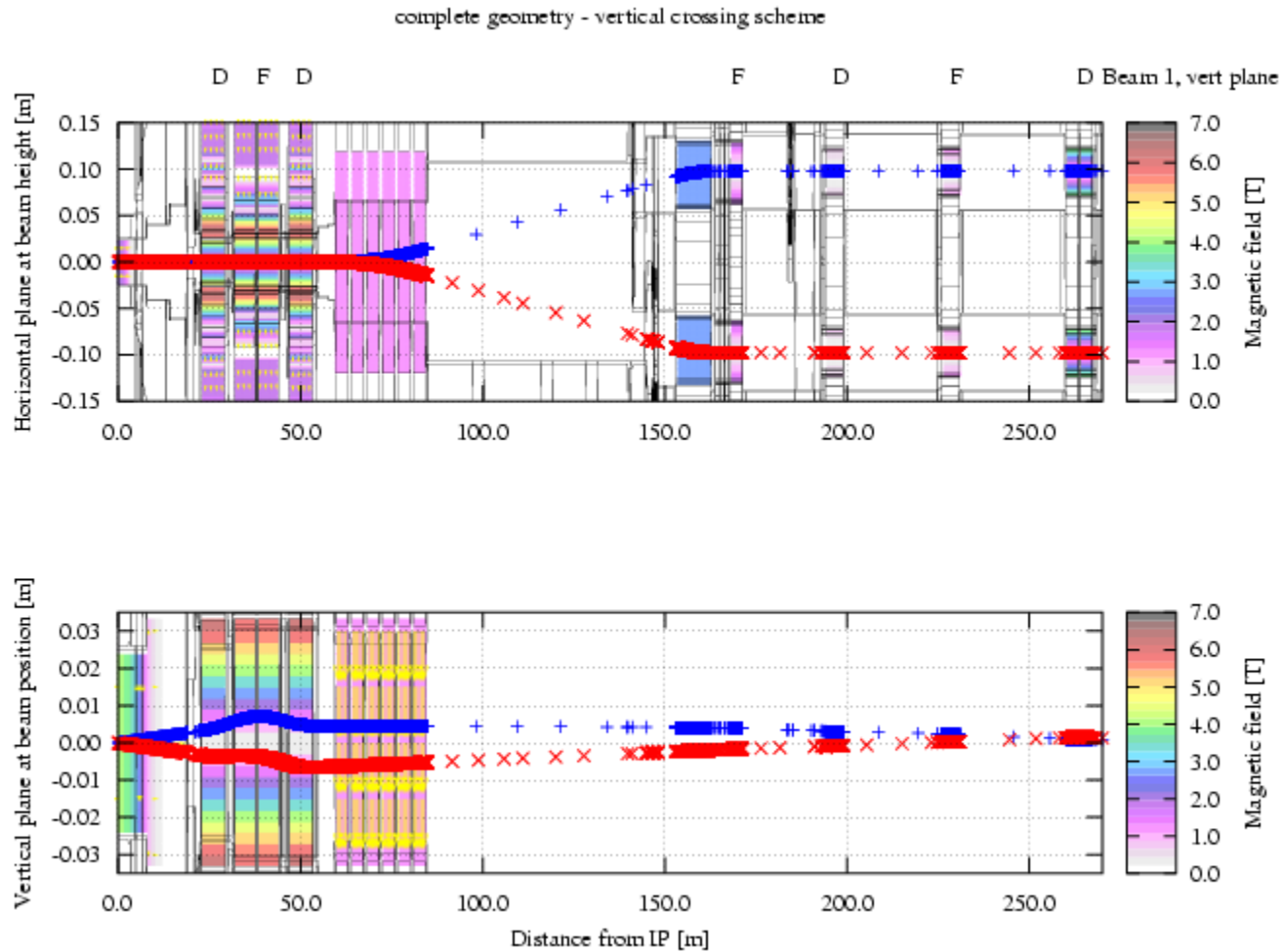
charged particle tracks in the CNGS geometry



Focused
De-focused
Many Escaping

Implementing magnetic field - 5

beam crossing in ATLAS



Implementing customized scoring - 1

Input card: **USERDUMP**

User routine: **\$FLUPRO/usermvax/mgdraw.f**
to be linked in one's own executable !

In mgdraw.f

you can get particle trajectories and (continuous and local) energy losses

you can go over reaction products

you can access information at each boundary crossing, particle step,
energy deposition event, *interaction*

In mdstck.f (for which no activating card is needed) as well,
with the additional – dangerous – possibility of influencing
the subsequent transport of secondaries

Implementing customized scoring-2

Input card: **USERWEIG**

User routines: [\\$FLUPRO/usermvax/comscw.f](#)
[\\$FLUPRO/usermvax/fluscw.f](#)

to be linked in one's own executable !

In comscw.f

you can apply a user defined weight on (even discard) deposited energy, stars or residual nuclei

you can extract information (and dump it on a file) about the involved particles

In fluscw.f

same as comscw.f for fluence scoring

Implementing customized flags

identification of ancestors of muon-neutrinos reaching Gran Sasso

out of target



ancestor as exiting the target

decay parent



parent decaying in ν_μ

No input card

User routines: [\\$FLUPRO/usermvax/stuprf.f](#)
[\\$FLUPRO/usermvax/stupre.f](#)

to be linked in one's own executable !

In `stuprf.f`

you can keep track of the particle origin

`stupre.f` applies in case of electrons, photons and positrons

Implementing region independent importance biasing

Input card: **BIASING**

User routines: **\$FLUPRO/usermvax/usimbs.f**

to be linked in one's own executable !

usimbs.f

is called for every particle step in user selected regions in order to return the change in the importance from the beginning to the end of the step, in a region-independent way!

It can be a great time saver as segmenting a complex geometry for region importance biasing turns out to be a cumbersome task