The FLUKA Code: a short introduction

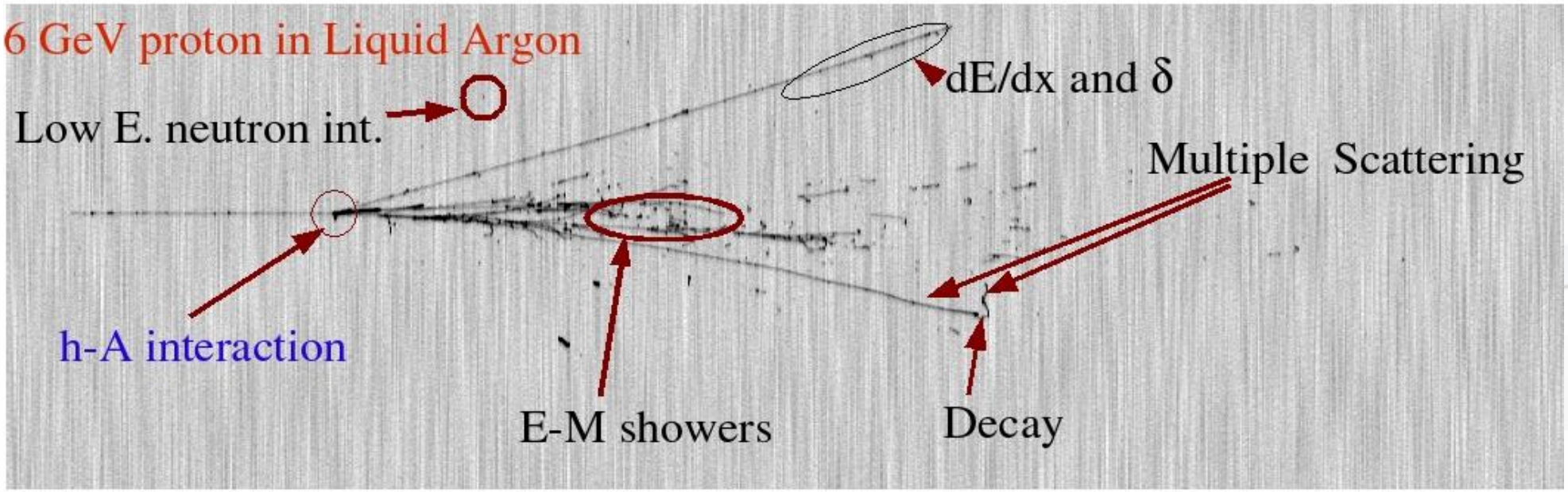
An Introduction to FLUKA:
a Multipurpose Particle Interaction and Transport MC code

OMA School on Monte Carlo simulations
Ludwig Maximilian University of Munich
6-10 November 2017

FLUKA

Main authors: A. Fassò, A. Ferrari, J. Ranft, P.R. Sala

Contributing authors: G. Battistoni, F. Cerutti, M. Chin, T. Empl, M.V. Garzelli, M. Lantz, A. Mairani, V. Patera, S. Roesler, G. Smirnov, F. Sommerer, V. Vlachoudis



Developed and maintained under an INFN-CERN agreement

>10000 registered users

http://www.fluka.org

The FLUKA International Collaboration

R. Augusto, G. Aricò, C. Bahamonde Castro, M.I. Besana, M. Brugger, F. Cerutti, A. Cimmino, L. Esposito, *Alfredo Ferrari*, R. Garcia Alia, J. Idarraga Munoz, W. Kozlowska, A. Lechner, M. Magistris, A. Mereghetti, E. Nowak, S. Roesler, F. Salvat-Pujol, P. Schoofs, E. Skordis, G. Smirnov, C. Theis, A. Tsinganis, Heinz Vincke, Helmut Vincke, V. Vlachoudis, J.Vollaire CERN

G. Battistoni, F. Broggi, M. Campanella, I. Mattei, S. Muraro, P.R. Sala, S.M. Valle INFN. Milano, Italy
N. Mazziotta INFN Bari, Italy A. Margiotta INFN & Univ. Bologna, Italy M.C. Morone Univ. Roma II, Italy
F. Ballarini, E. Bellinzona, M. Carante, A. Embriaco, A. Fontana INFN & Univ. Pavia, Italy
L. Sarchiapone INFN Legnaro, Italy V. Patera, S. Pioli INFN Frascati & Univ. Roma I , Italy
P. Colleoni, Ospedali Riuniti di Bergamo, Italy G. Magro, M. Pelliccioni CNAO Pavia, Italy
A. Mairani, CNAO Pavia, Italy & HIT, Germany

P. Degtiarenko, G. Kharashvili, JLab, USA M. Santana, SLAC, USA L. Lari ,FNAL USA
A. Empl, S. Hoang, M. Kroupa, L. Pinsky Univ. of Houston, USA
K.T. Lee, E. Semones, N. Stoffle, N. Zapp NASA, Houston, USA
A.Bahadori Kansas Univ. USA M. Trinczec, A. Trudel TRIUMF, Canada

G. Dedes, S. Mayer, K. Parodi, LMU Munich, Germany Anna Ferrari, S. Mueller HZDR Rossendorf, Germany
S. Brechet, L. Morejon, N. Shetty, S. Stransky, S. Trovati, R. Versaci, ELI-Beamlines, Prague, Czechia
T.J. Dahle, L. Fjera, A. Rorvik, K. Ytre-Hauge , Bergen Univ., Norway
F. Belloni INSTN-CEA, France

A. Fedynitch DESY Zeuthen, Germany T. T. Boehlen, Medaustron, Austria, D.Georg, MedUni, Vienna, Austria
C. Cuccagna, TERA Switzerland T.V. Miranda Lima Kantonhospital Aarau, Switzerland
M. Lantz, Uppsala Univ., Sweden F. Fiorini, Oxford Inst. Rad. Oncol., UK
P. Garcia Ortega IUFFYM, Spain I. Rinaldi, INP Lyon, France
M. Chin, Malaysia

A. Fassò, M.V. Garzelli, E. Gadioli, J. Ranft

FLUKA short description:

- FLUKA is a general purpose tool for calculations of particle transport and interactions with matter
- All Hadrons (p, n, π, K,pbar, nbar, (anti)hyperons…) [0-10000 TeV]
- Electromagnetic (γ, $e^{+/-}$) and μ and ν [1 keV - 10000 TeV]
- Nucleus-nucleus [0-10000 TeV/n]
- Low energy neutrons (0-20 MeV, multigroup, ENDF…)
- Transport in magnetic field
- Combinatorial (boolean) and Voxel geometries
- Double capability to run either fully analogue and/or biased calculations
- On-line evolution of induced radioactivity and dose
- Radiation damage predictions (NIEL, DPA)
- User-friendly GUI interface thanks to the Flair interface

http://www.fluka.org

Particles transported by FLUKA:

FLUKA name	FLUKA number	Symbol	Common name	Standard PDG number (Particle Data Group) [142]
4-HELIUM (1)	-6	α	Alpha	—
3-HELIUM (1)	-5	^{3}He	Helium 3	—
TRITON (1)	-4	^{3}H	Triton	—
DEUTERON (1)	-3	^{2}H	Deuteron	—
HEAVYION (1)	-2	—	Generic Heavy Ion with Z > 2 (see command HI–PROPE)	—
OPTIPHOT	-1	—	Optical Photon	—
RAY (2)	0	—	Pseudoparticle	—
PROTON	1	p	Proton	2212
APROTON	2	$\bar{\text{p}}$	Antiproton	-2212
ELECTRON	3	e^-	Electron	11
POSITRON	4	e^+	Positron	-11
NEUTRIE	5	ν_e	Electron Neutrino	12
ANEUTRIE	6	$\bar{\nu}_e$	Electron Antineutrino	-12
PHOTON	7	γ	Photon	22
NEUTRON	8	n	Neutron	2112
ANEUTRON	9	$\bar{\text{n}}$	Antineutron	-2112
MUON+	10	μ^+	Positive Muon	-13
MUON-	11	μ^-	Negative Muon	13
KAONLONG	12	K^0_L	Kaon-zero long	130
PION+	13	π^+	Positive Pion	211
PION-	14	π^-	Negative Pion	-211
KAON+	15	K^+	Positive Kaon	321
KAON-	16	K^-	Negative Kaon	-321
LAMBDA	17	Λ	Lambda	3122
ALAMBDA	18	$\bar{\Lambda}$	Antilambda	-3122
KAONSHRT	19	K^0_S	Kaon-zero short	310
SIGMA-	20	Σ^-	Negative Sigma	3112
SIGMA+	21	Σ^+	Positive Sigma	3222
SIGMAZER	22	Σ^0	Sigma-zero	3212
PIZERO	23	π^0	Pion-zero	111
KAONZERO	24	K^0	Kaon-zero	311
AKAONZER	25	$\bar{\text{K}}^0$	Antikaon-zero	-311
Reserved	26	—	—	—
NEUTRIM	27	ν_μ	Muon Neutrino	14
ANEUTRIM	28	$\bar{\nu}_\mu$	Muon Antineutrino	-14
Blank	29	—	—	—

table continues

FLUKA name	FLUKA number	Symbol	Common name	Standard PDG (Particle Data
Reserved	30	—	—	—
ASIGMA-	31	$\bar{\Sigma}^-$	Antisigma-minus	-3222
ASIGMAZE	32	$\bar{\Sigma}^0$	Antisigma-zero	-3212
ASIGMA+	33	$\bar{\Sigma}^+$	Antisigma-plus	-3112
XSIZERO	34	Ξ^0	Xi-zero	3322
AXSIZERO	35	$\bar{\Xi}^0$	Antixi-zero	-3322
XSI-	36	Ξ^-	Negative Xi	3312
AXSI+	37	$\bar{\Xi}^+$	Positive Xi	-3312
OMEGA-	38	Ω^-	Omega-minus	3334
AOMEGA+	39	$\bar{\Omega}^+$	Antiomega	-3334
Reserved	40	—	—	—
TAU+	41	τ^+	Positive Tau	-15
TAU-	42	τ^-	Negative Tau	15
NEUTRIT	43	ν_τ	Tau Neutrino	16
ANEUTRIT	44	$\bar{\nu}_\tau$	Tau Antineutrino	-16
D+	45	D^+	D-plus	411
D-	46	D^-	D-minus	-411
D0	47	D^0	D-zero	421
DOBAR	48	$\bar{\text{D}}^0$	AntiD-zero	-421
DS+	49	D_s^+	D_s-plus	431
DS-	50	D_s^-	D_s-minus	-431
LAMBDAC+	51	Λ_c^+	Lambda$_c$-plus	4122
XSIC+	52	Ξ_c^+	Xi$_c$-plus	4232
XSIC0	53	Ξ_c^0	Xi$_c$-zero	4132
XSIPC+	54	$\Xi_c'^+$	Xi$'_c$-plus	4322
XSIPC0	55	$\Xi_c'^0$	Xi$'_c$-zero	4312
OMEGAC0	56	Ω_c^0	Omega$_c$-zero	4332
ALAMBDC-	57	$\bar{\Lambda}_c^-$	Antilambda$_c$-minus	-4122
AXSIC-	58	$\bar{\Xi}_c^-$	AntiXi$_c$-minus	-4232
AXSIC0	59	$\bar{\Xi}_c^0$	AntiXi$_c$-zero	-4132
AXSIPC-	60	$\bar{\Xi}_c'^-$	AntiXi$'_c$-minus	-4322
AXSIPC0	61	$\bar{\Xi}_c'^0$	AntiXi$'_c$-zero	-4312
AOMEGAC0	62	$\bar{\Omega}_c^0$	AntiOmega$_c$-zero	-4332
Reserved	63	—	—	—
Reserved	64	—	—	—

The FLUKA Code design

- Based, as far as possible, on original and well-tested microscopic models
- Full cross-talk between all components: hadronic, electromagnetic, neutrons, muons, heavy ions
- It is a "condensed history" MC code, however with the possibility to use single instead of multiple scattering

FLUKA is NOT a toolkit! Its physical models are fully integrated

- **The user does not need to choose a "physics list"**
- **The user has, however, the possibility to optimize CPU vs accuracy when needed**

- Fluka provides powerful built-in scoring, well tested and suited for most applications
  - **The user does not need to write external code to get results and statistics**

Online evolution and buildup of induced activity

In all accelerator-related applications... and in many more, the evaluations of induced activation and prompt/residual dose rates are essential. In **Fluka**:

- Reliable interaction models, with particular care of the latest stages (evaporation, fragmentation, break-up)
- Decay β+/-'s, γ's, EC electrons, α's, produced according to a database (based on ENSDF, www.nndc.bnl.gov)
  - Screening and Coulomb corrections for $\beta^{+/-}$ spectra
- Analytical calculation of activity build-up and decay

In the same run, prompt and delayed results according to user-defined irradiation and cooling profiles

Activities: 2D and 3D distributions, and full inventories at each buildup/cooling time

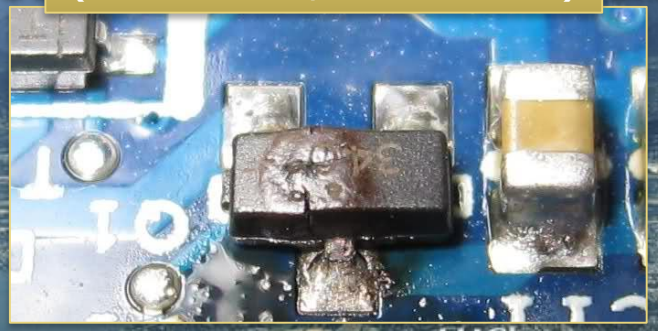
Energy deposition (dose, decay heat), particle fluences (including dose equivalent with folding online) at each buildup/cooling time

What can be done with FLUKA?

Some examples

Cern & Fluka (~90 orldwide, http://www.fluka.org)

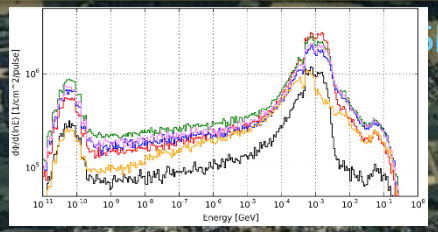
Radiation damage (electronics, insulations)



Shielding, residual dose rates

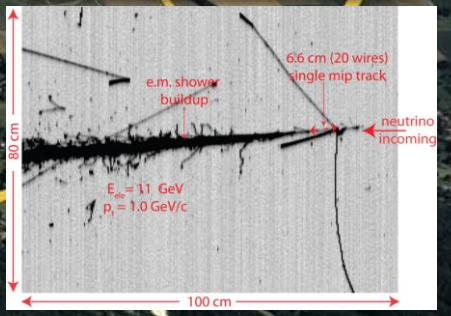
Activation, Waste disposal

Spallation sources (n_ToF), secondary beams (ν!!)



Energy deposition (quenching, damage)

(ν) Experiments



LHCb

ATLAS

ALICE

SUISSE

FRANCE

SPS 7 km

SPS: 450 GeV p

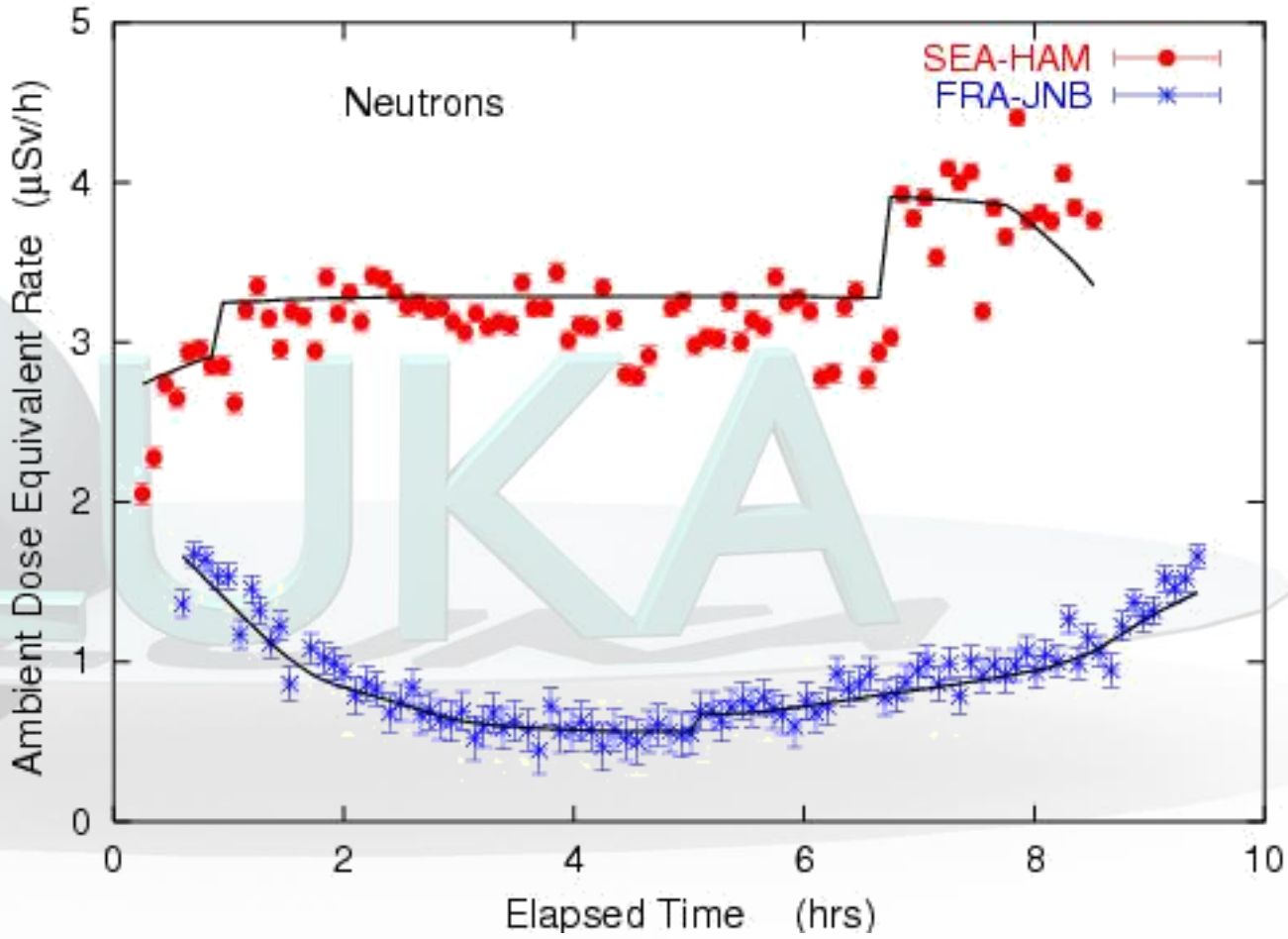
LHC 27 km

Dosimetry + cosmic rays

Complete simulation of cosmic rays interactions in the atmosphere
- Dedicated "cosmic" package available to users

Model of airplane geometry
Response of dosimeters

Dose to aircrew on commercial flights , depending on route



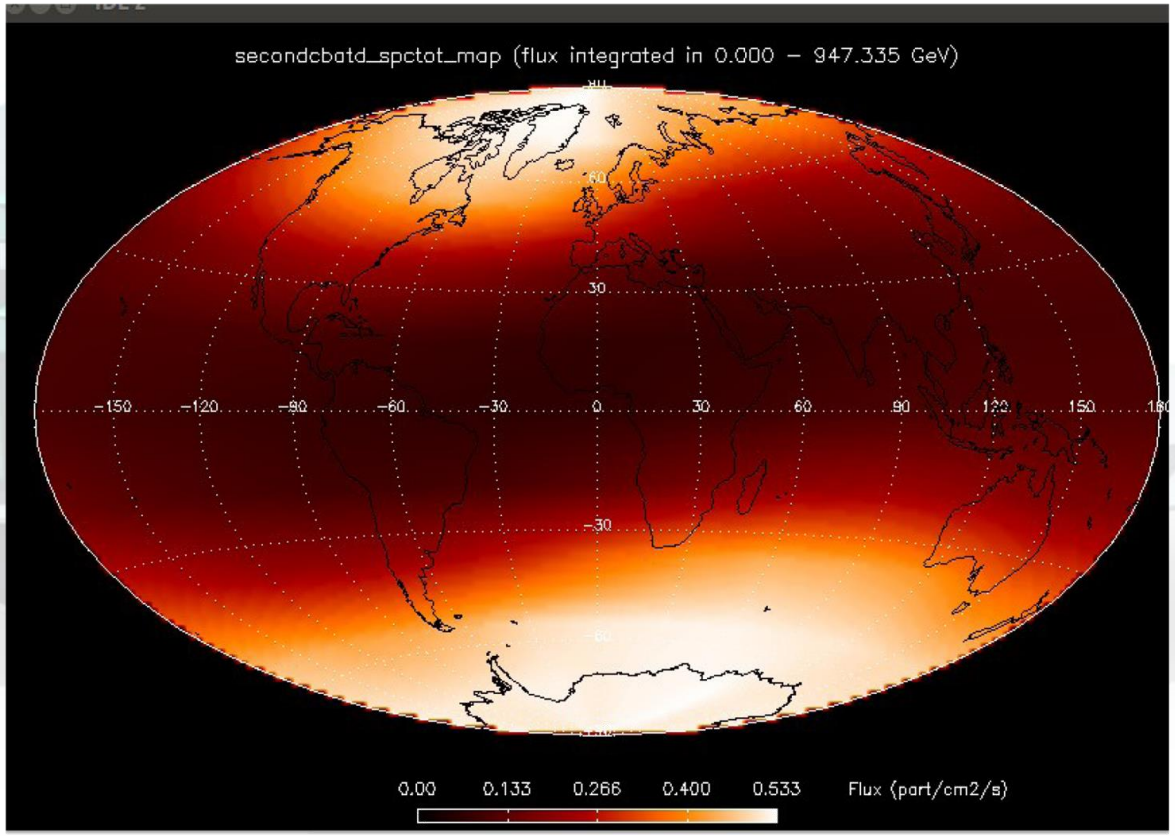
Ambient dose equivalent from neutrons at solar maximum on commercial flights from Seattle to Hamburg and from Frankfurt to Johannesburg.

Solid lines: FLUKA simulation

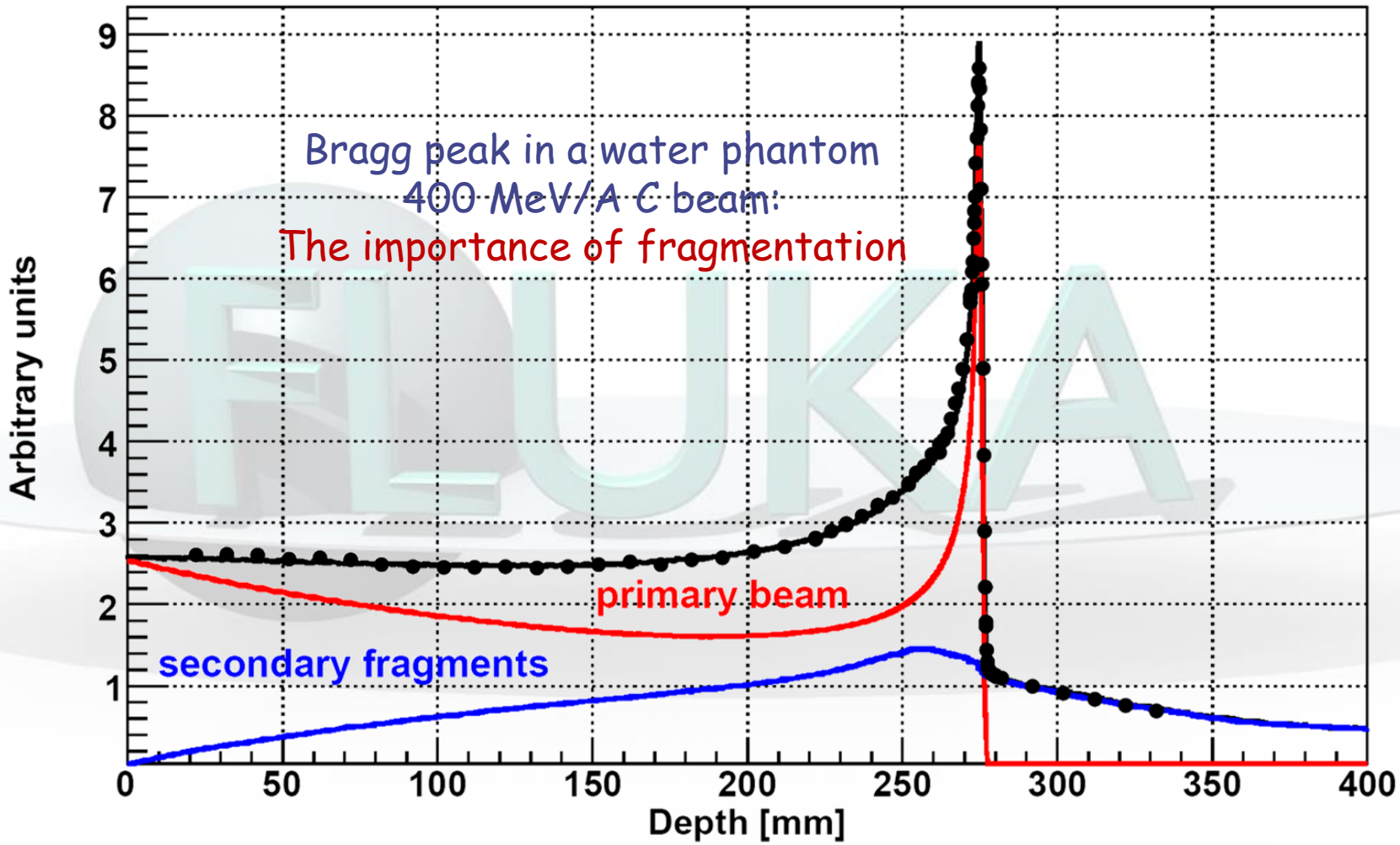
Roesler et al., Rad. Prot. Dosim. 98, 367 (2002)

The neutron albedo from GCR's at 400 km altitude*

*In collaboration with CEA-Saclay

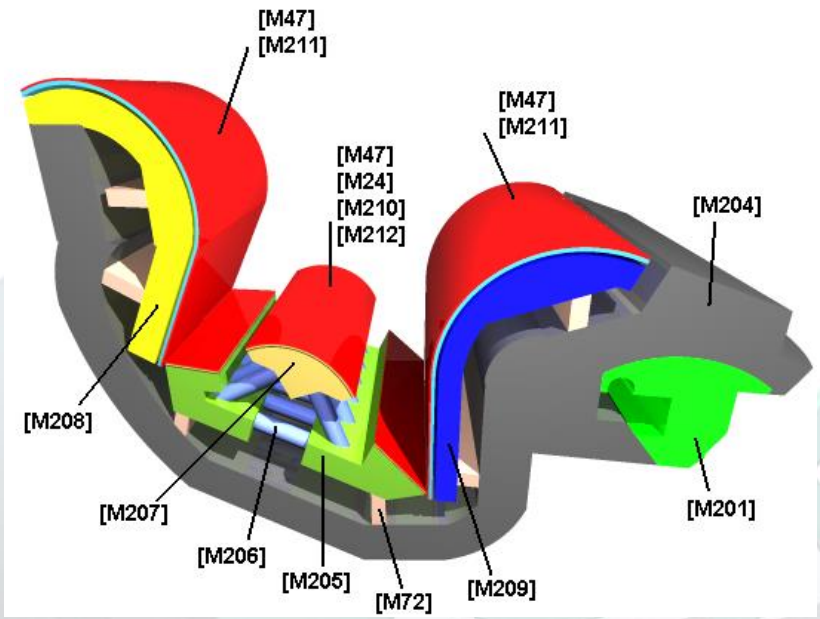


Medical physics : Radiotherapy

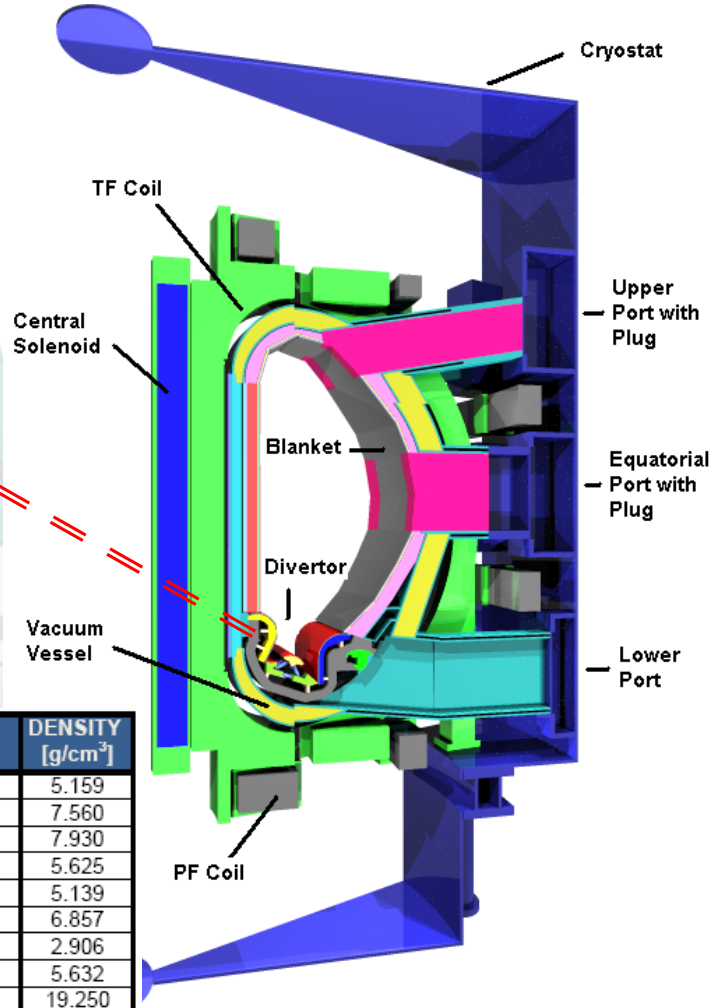


Exp. Data (points) from Haettner et al, Rad. Prot. Dos. 2006
Simulation: A. Mairani PhD Thesis, 2007, Nuovo Cimento C, 31, 2008

ITER geometry: divertor cassette



[FLUKA ray tracer by D. Pastor]

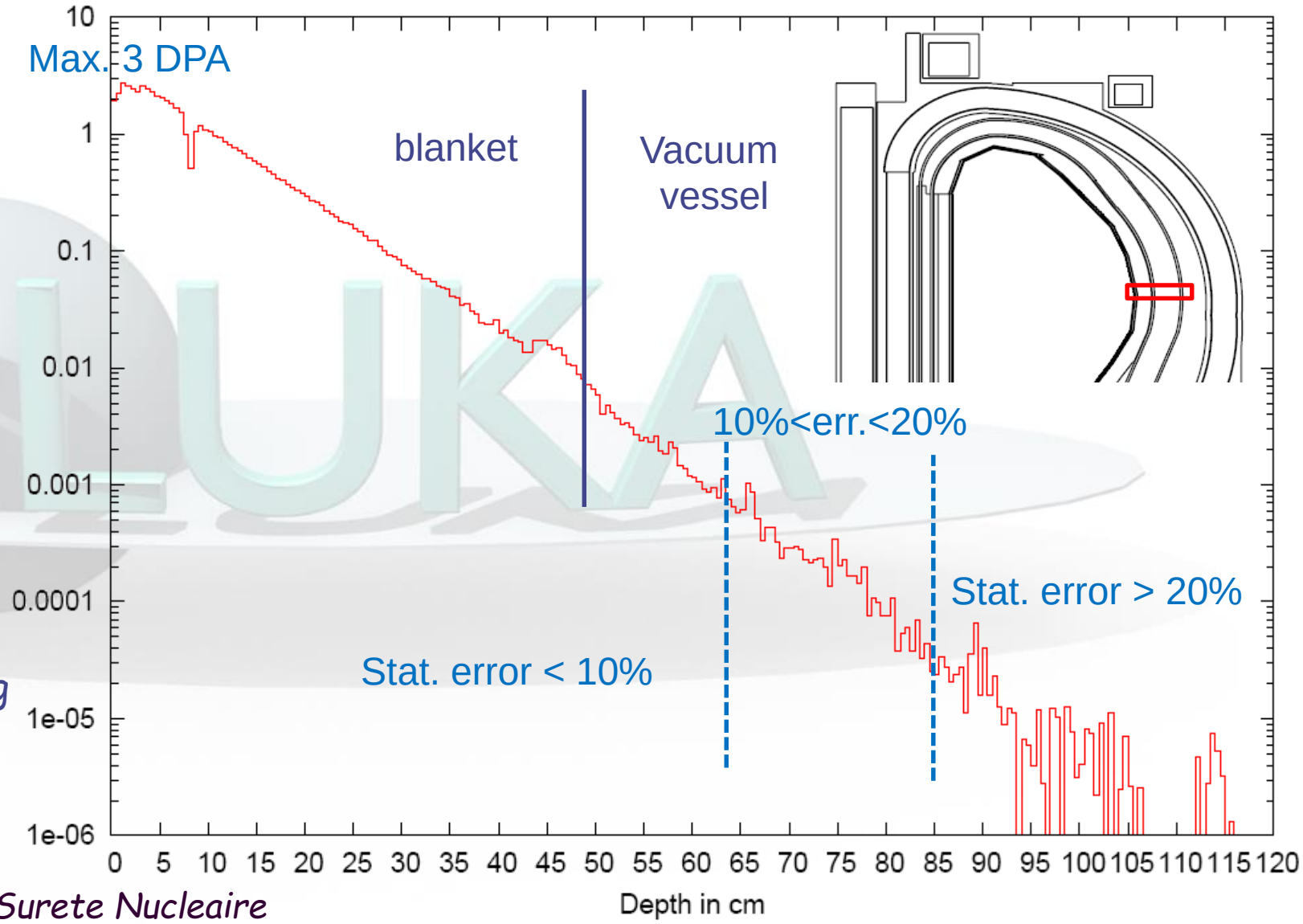


COMPONENT & LOCATION		MATERIAL NAME	DESCRIPTION	DENSITY [g/cm^3]
DIVERTOR CASSETTE	Support Structure 1	M204	H2O (39%), SS316L(N)-IG (52%), M72 (9%)	5.159
	Support Structure 2	M201	NiAl (100%)	7.560
	Inner & Outer Support Structure	M12	SS316L(N)-IG (100%)	7.930
	Target Structure Inboard	M208	H2O (32%), SS316L(N)-IG (50%), M72 (18%)	5.625
	Target Structure Outboard	M209	H2O (39%), SS316L(N)-IG (44%), M72 (17%)	5.139
	Target Structure Dome	M207	H2O (15%), SS316L(N)-IG (85%)	6.857
	Pipes for the Dome	M206	H2O (72%), SS316L (28%)	2.906
	Target Strucure Dome Right & Left	M205	H2O (30%), M72 (70%)	5.632
	Plasma Facing W	M47	TUNGSTEN (100%)	19.250
	Plasma Facing Cu	M24	COPPER (100%)	8.960
	Plasma Facing	M210	Cu-Cr-Zr (100%)	8.814
	Plasma Facing	M211	W (57.65%), H2O (21.1%), Cu-Cr-Zr (11.87%), Cu (9.38%)	13.179
	Plasma Facing	M212	H2O (76.4%), SS316L(N)-IG (23.6%)	2.570
	Connection	M72	Fe (56.62%), Cr (22%), Ni (12.5%), Mo (2.25%), N (0.3%), Mn (5%), C (0.06%), Nb (0.2%), V (0.2%), Si (0.75%), [illegible] (0.04%), S (0.03%), Co (0.05%)	7.650

DPA: ITER, blanket module

For a safety evaluation of the ITER machine, IRSN* asked CERN to perform neutronic and activation calculations with the FLUKA Monte Carlo code.

The report presents FLUKA results on radionuclide inventory, ambient residual dose rate, dose evaluation, residual power for the cooling circuit, and displacement per atom (DPA)



*Institute de Radioprotection et Surete Nucleaire

The MC course: an Introduction

How:

This course is ***not a full FLUKA course***. It is intended to provide a basic and general overview on:

a) The most relevant FLUKA instructions and options

b) The basic principle of combinatorial geometry in FLUKA

c) The physics models adopted in FLUKA

d) Running, scoring and plotting in FLUKA

For further information, refer to the existing documentation.
In this course the basic FLUKA input and the graphical interface Flair will be presented.

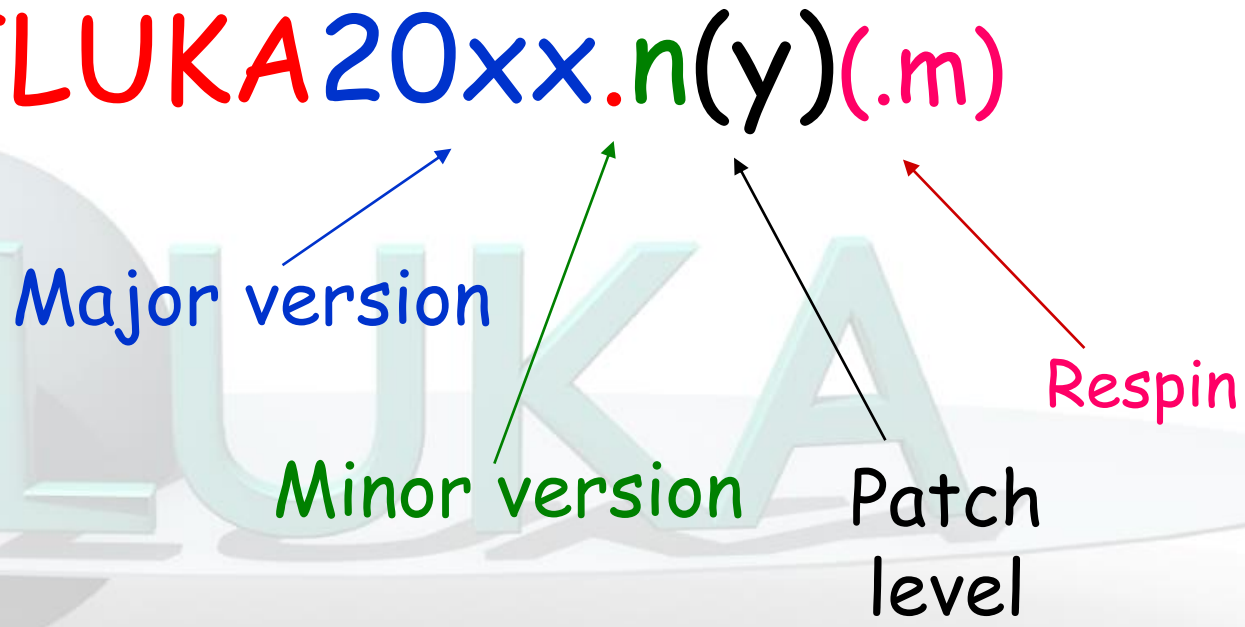
Practical exercises:
particle accelerators, dose scoring (depending on the time)

Agenda

	6 Nov	7 Nov	8 Nov	9 Nov	10 Nov
8:30 - 9:00	General introduction (A. Ferrari)	Penelope models (low-energy elctromagnetic) (J.M. Fernández-Varea)	Geometry in FLUKA (E. Skordis)	Heavy ion beams and radioactivity in FLUKA (A. Mairani)	Geant4 for space and medicine (G. Dedes)
9:00 - 9:30	Principles of Monte Carlo calculations and codes (F.Salvat)				
9:30 - 10:00		Electron accelerator modeling with Penelope + Tutor (L. Brualla)	FLUKA models (A. Ferrari)	Monte Carlo treatment planning in FLUKA (A. Mairani)	
10:00 - 10:30					
10:30 - 11:00	coffee break		coffee break	coffee break	coffee break
11:00 - 11:30	Introduction to Geant4 (G. Dedes)	Transfer to Munich + lunch	Ionization and transport in FLUKA (F. Salvat-Pujol)	Scoring in FLUKA (A. Mairani)	Outlook in Penelope (F. Salvat-Pujol)
11:30 - 12:00					
12:00 - 12:30				Transfer to Munich + lunch	
12:30 - 13:00	lunch		lunch		lunch
13:00 - 13:30					
13:30 - 14:00		Exercises with Penelope			
14:00 - 14:30	Structure and operation of PENELOPE (F. Salvat)		Accelerators in FLUKA (E. Skordis)	Exercises with FLUKA	Outlook in GeantV (A. Gheata)
14:30 - 15:00					
15:00 - 15:30	Introduction to FLUKA (A. Ferrari)	coffee break	Social activity		GeantV: machine learning techniques (S. Vallecorsa)
15:30 - 16:00		Exercises with Geant4			
16:00 - 16:30	coffee break			coffee break	coffee break
16:30 - 17:00	Introduction to Flair (E. Skordis)	Poster session		Exercises: medical applications with FLUKA	Outlook in FLUKA (A. Ferrari)
17:00 - 17:30					
17:30 - 18:00	Geometry in Penelope (F. Salvat)				Closing remarks
18:00 - 18:30					

A glimpse of FLUKA

The FLUKA version



In this course we are using FLUKA2011.2c.6

The FLUKA license (it is not GPL):

- Standard download: **binary library + user routines**.
  - FLUKA can be used freely for scientific and academic purposes, ad-hoc agreement for commercial purposes
  - It cannot be used for weapon related applications
- It is possible, by explicit signing of license, to download the **source** for researchers of scientific/academic Institutions.
  - FLUKA can neither be copied into other codes (not even partially), nor translated into another language without permission
- For **commercial use**, trial version (limited in time and random seeds) available. Commercial license to be negotiated with CERN & INFN.

Please register on www.fluka.org .....and read the license!

The FLUKA mailing lists

- fluka-users@fluka.org

  Users are automatically subscribed here when registering on the web site.
  It is used to communicate the availability of new versions, patches, etc.

- fluka-discuss@fluka.org

  Users are encouraged to subscribe at registration time, but can uncheck the relevant box. It is used to have user-user and user-expert communication about problems, bugs, general inquiries about the code and its physics content

  Users are strongly encouraged to keep this subscription

Using FLUKA

Platform: Linux with g77 (on 32 and 64 bit machines)
and gfortran (on 64 bit machines)

Work in progress: Mac OSX with gfortran

The code may be compiled/run only using operating systems, compilers (and associated) options tested and approved by the development team

Standard Input:

- Command/options driven by "data cards" (ascii file)
  Graphical interface is available
- Standard Geometry ("Combinatorial geometry"): input by "data cards"

Standard Output and Scoring:

- Apparently limited but highly flexible and powerful
- Output processing and plotting interface available

A brief overview:

your first input

FLUKA Manual

in *continuous* development (as the program)!

More a User Guide than a Reference Manual (only a short summary about physics)

Many formats: pdf: FM.pdf, ASCII : fluka2011.manual, Tk interface accessible through FLAIR, html on FLUKA website

REMEMBER!

The first place to look at when puzzled!

...and the very best friend of a beginner user!

FLUKA and FLAIR

FLUKA users can….

- prepare their own input with a text editor (remember: the input is a simple text file)
- use their own tools for plotting results
- submit jobs by command line

FLAIR (Fluka Advanced InteRface) can be used as well
It helps the users with the aforementioned tasks
(see FLAIR lecture)

You can choose your favorite way…

…but a good user should be able to go both ways!

Structure of the input file

General definitions

Beam definition
Materials: definition and assignment
Random number initialization
Start/Stop of simulation

Physics settings

Defaults
Physical processes
Transport thresholds
Low energy neutrons
Induced radioactivity

Biasing

Geometry related biasing
Interaction/decay biasing

Geometry

Setup description
Voxel phantoms

Output settings

Estimators / scoring cards

FLUKA input file commands

Commands aka cards, aka options, aka directives, aka definitions

One keyword (command), 6 floating point numbers (WHATs), one string (SDUM)

Example of a FLUKA command

```
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
BEAM          1.E+04        0.0        0.0        0.0        0.0        0.0PROTON
*keyword     momentum mom.spread  diverg.    X-width    Y-width  ignored particle
*             WHAT(1)    WHAT(2)    WHAT(3)    WHAT(4)    WHAT(5)    WHAT(6)   SDUM
```

- Command keywords MUST be uppercase, numbers MUST have the decimal point
- Some commands require more than one "card"
- With few exceptions, the order of commands is irrelevant
- Repeated commands can add themselves or override previous commands
- A line with a * character in column 1 is a comment
- Text after an exclamation mark (!) is ignored (does not work within the geometry)
- Almost all the WHAT() have a default value
- Commands can be issued in fixed or free format
- Special commands, called #directives, allow for input parametrization

Fixed format

Fixed format:

```
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
BEAM            1.E+04    0.0D+00         0.0         0.0         0.0         0.0PROTON
*keyword     momentum mom.spread  diverg.    X-width     Y-width  ignored particle
*              WHAT(1)    WHAT(2)    WHAT(3)    WHAT(4)    WHAT(5)    WHAT(6)   SDUM
```

- The "traditional" FLUKA format is (A8, 2X, 6E10.0, A8)

  Numbers: 9 digits at most can be used!
- All WHAT fields are in floating point format, *even integers*

  They must always be written with the decimal point
- Exponential notation numbers (e.g. 1.234E+5), must be right aligned
- Double precision format (e.g. 1.234D+5) is allowed
- Blank numerical fields are read as 0.0

  In most cases (*not all!*) such values are ignored and

  the corresponding default values are used
- Blank lines NOT ALLOWED in geometry declaration (tolerated elsewhere)
- FLAIR takes care of all the alignment problems for you

Names instead of numbers

- FLUKA also allows to use keywords (names) - 8 characters maximum length - instead of numbers inside FLUKA commands
- Examples later (for instance materials, or geometrical region, can be inserted using their names instead of numbers)
- This *helps* the user, improving the *readability* of the input FLUKA file
- Internally, FLUKA works always by NUMBERS, and keeps name-to-number bidirectional tables / functions

A basic input: step 1: Physics settings

```
TITLE
My Basic Input example
* Set the defaults for precision simulations
DEFAULTS                                                              PRECISIO
* Define the beam characteristics
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAM             3.5 -0.082425       -1.7        0.0        0.0          PROTON
* Define the beam position
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAMPOS          0.0         0.0       -0.1        0.0        0.0
*
GEOBEGIN                                                              COMBNAME
    0     0
* Black body
SPH blkbody     0.0 0.0 0.0 100000.0
* Void sphere
SPH void        0.0 0.0 0.0 10000.0
* Cylindrical target
RCC target1     0.0 0.0  0.0 0.0 0.0 10.0 5.0
RCC target2     0.0 0.0 20.0 0.0 0.0 10.0 5.0
RCC target3     0.0 0.0 40.0 0.0 0.0 10.0 5.0
END
* Black hole
BLKBODY       5 +blkbody -void
* Void around
VOID          5 +void -target1 -target2 -target3
* Target
TARGET1       5 +target1
TARGET2       5 +target2
TARGET3       5 +target3
END
GEOEND
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
MATERIAL         24.0                7.18                             CHROMIUM
MATERIAL          0.0             0.73E-3                             AMMONIA
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
COMPOUND          1.0  NITROGEN       3.0  HYDROGEN                   AMMONIA
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASSIGNMA    BLCKHOLE    BLKBODY
ASSIGNMA      VACUUM       VOID
ASSIGNMA     AMMONIA    TARGET3
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..*
ASSIGNMA    CHROMIUM    TARGET1    TARGET2        1.0

* Set the random number seed
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RANDOMIZ          1.0 54217137.
*
* Set the number of primary histories to be simulated in the run
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
START           1000.
STOP
```

A basic input: step 1: Physics settings

Defaults set reasonable FLUKA defaults for a specified kind of problem:

SDUM = CALORIMEtry : calorimetry simulations

EET/TRANsmut : energy transformer or transmutation calculations

EM-CASCAde : pure EM cascades

ICARUS : studies related to the ICARUS experiment

HADROTHErapy : hadrotherapy calculations

NEUTRONS : pure low-energy neutron runs

NEW-DEFAults : reasonable minimal set of new defaults (very coarse!) – set by default

PRECISIOn : precision simulations

SHIELDINg : hadron shielding calculations without gammas

A basic input: step 2: Beam

```
TITLE
My Basic Input example
* Set the defaults for precision simulations
DEFAULTS                                                              PRECISIO
* Define the beam characteristics
BEAM              3.5 -0.082425       -1.7        0.0        0.0           PROTON
* Define the beam position
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAMPOS           0.0         0.0       -0.1        0.0        0.0
*
GEOBEGIN                                                              COMBNAME
    0    0
* Black body
SPH blkbody    0.0 0.0 0.0 100000.0
* Void sphere
SPH void       0.0 0.0 0.0 10000.0
* Cylindrical target
RCC target1    0.0 0.0  0.0 0.0 0.0 10.0 5.0
RCC target2    0.0 0.0 20.0 0.0 0.0 10.0 5.0
RCC target3    0.0 0.0 40.0 0.0 0.0 10.0 5.0
END
* Black hole
BLKBODY       5 +blkbody -void
* Void around
VOID          5 +void -target1 -target2 -target3
* Target
TARGET1       5 +target1
TARGET2       5 +target2
TARGET3       5 +target3
END
GEOEND
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
MATERIAL         24.0                7.18                               CHROMIUM
MATERIAL          0.0             0.73E-3                               AMMONIA
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
COMPOUND          1.0  NITROGEN       3.0  HYDROGEN                     AMMONIA
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASSIGNMA     BLCKHOLE    BLKBODY
ASSIGNMA       VACUUM       VOID
ASSIGNMA      AMMONIA    TARGET3
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..*
ASSIGNMA     CHROMIUM    TARGET1    TARGET2       1.0

* Set the random number seed
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RANDOMIZ          1.0 54217137.
*
* Set the number of primary histories to be simulated in the run
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
START          1000.
STOP
```

A basic input: step 2: Beam

The card BEAM defines the particle type and energy (or momentum).

The card BEAMPOS controls particle starting position and direction.

For complex particle sources (complex distributions in energy, space and direction) a special user routine, SOURCE, can be used*. The card SOURCE has to be added in the input file.

** Some pre-defined cases (volume sources, cosmic ray source, uniform isotropic source etc are already built-in and available via data cards, see the manual*

A basic input: step 3: Geometry

```
TITLE
My Basic Input example
* Set the defaults for precision simulations
DEFAULTS                                                              PRECISIO
* Define the beam characteristics
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAM              3.5 -0.082425      -1.7       0.0       0.0           PROTON
* Define the beam position
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAMPOS           0.0       0.0      -0.1       0.0       0.0
*
GEOBEGIN                                                              COMBNAME
    0    0
* Black body
SPH blkbody    0.0 0.0 0.0 100000.0
* Void sphere
SPH void       0.0 0.0 0.0 10000.0
* Cylindrical target
RCC target1    0.0 0.0  0.0 0.0 0.0 10.0 5.0
RCC target2    0.0 0.0 20.0 0.0 0.0 10.0 5.0
RCC target3    0.0 0.0 40.0 0.0 0.0 10.0 5.0
END
* Black hole
BLKBODY       5 +blkbody -void
* Void around
VOID          5 +void -target1 -target2 -target3
* Target
TARGET1       5 +target1
TARGET2       5 +target2
TARGET3       5 +target3
END
GEOEND
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
MATERIAL         24.0                 7.18                            CHROMIUM
MATERIAL          0.0              0.73E-3                             AMMONIA
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
COMPOUND          1.0  NITROGEN       3.0  HYDROGEN                    AMMONIA
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASSIGNMA    BLCKHOLE    BLKBODY
ASSIGNMA      VACUUM       VOID
ASSIGNMA     AMMONIA    TARGET3
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..*
ASSIGNMA    CHROMIUM    TARGET1    TARGET2       1.0

* Set the random number seed
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RANDOMIZ          1.0 54217137.
*
* Set the number of primary histories to be simulated in the run
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
START          1000.
STOP
```

A basic input: step 3: Geometry

The Combinatorial Geometry in FLUKA must be preceded by a GEOBEGIN card and followed by a GEOEND card.

The whole geometry must be surrounded by a region of "blackhole" limited by a closed body.

For details on the Combinatorial Geometry (bodies, regions and optional region volumes) please wait for the Geometry lecture on Wednesday.

A basic input: step 4: Materials

```
TITLE
My Basic Input example
* Set the defaults for precision simulations
DEFAULTS                                                              PRECISIO
* Define the beam characteristics
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAM              3.5 -0.082425      -1.7       0.0       0.0           PROTON
* Define the beam position
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAMPOS           0.0       0.0      -0.1       0.0       0.0
*
GEOBEGIN                                                              COMBNAME
    0    0
* Black body
SPH blkbody    0.0 0.0 0.0 100000.0
* Void sphere
SPH void       0.0 0.0 0.0 10000.0
* Cylindrical target
RCC target1    0.0 0.0  0.0 0.0 0.0 10.0 5.0
RCC target2    0.0 0.0 20.0 0.0 0.0 10.0 5.0
RCC target3    0.0 0.0 40.0 0.0 0.0 10.0 5.0
END
* Black hole
BLKBODY       5 +blkbody -void
* Void around
VOID          5 +void -target1 -target2 -target3
* Target
TARGET1       5 +target1
TARGET2       5 +target2
TARGET3       5 +target3
END
GEOEND
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
MATERIAL         24.0                7.18                               CHROMIUM
MATERIAL          0.0             0.73E-3                               AMMONIA
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
COMPOUND          1.0  NITROGEN       3.0  HYDROGEN                     AMMONIA
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASSIGNMA     BLCKHOLE    BLKBODY
ASSIGNMA       VACUUM       VOID
ASSIGNMA      AMMONIA    TARGET3
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..*
ASSIGNMA     CHROMIUM    TARGET1    TARGET2        1.0
* Set the random number seed
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RANDOMIZ          1.0 54217137.
*
* Set the number of primary histories to be simulated in the run
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
START          1000.
STOP
```

Materials

FLUKA handles:

- elemental materials (by default natural composition, the user can set a specific isotope, being aware of low energy neutron cross sections availability)
- compounds (chemical molecules, alloys, mixtures…)

Each material is uniquely identified by an index/name

FLUKA has a set of predefined materials.

Users can both use/modify these and define their own ones.

Basic cards:

MATERIAL	material declaration
COMPOUND	compound definition (a MATERIAL card is mandatory for a compound declaration)
ASSIGNMA	material assignment to regions of geometry

Predefined materials

In FLUKA 2 special materials +23 natural elements of most common use, e.g. Oxygen, Carbon, Iron… (check them out in the manual, Chap. 5), are predefined.

The first two are very important:

- BLCKHOLE (mat #1): material with infinite absorbance;
- VACUUM (mat #2)

12 compound materials with the composition suggested by ICRU are predefined as well (again, check the manual!).

All predefined materials can be used WITHOUT explicit MATERIAL / COMPOUND cards.

WARNING: user defined MATERIAL cards OVERRIDE PREDEFINED materials having the same name.

Material definition: MATERIAL

Defines a new material or override a previous one.

- [SDUM] material name
- [WHAT(1)] atomic number Z
- [WHAT(2)] atomic weight (leave it empty)
- [WHAT(3)] density [g/cm^3]
- [WHAT(4)] material number (leave it empty)
- [WHAT(5)] alternate material to be used for dE/dx (normally empty)
- [WHAT(6)] mass number A (leave it empty unless you want a specific isotope)

Material definition: COMPOUND

Defines a new compound.

Each COMPOUND card must be associated to a MATERIAL card.

More COMPOUND card can be used to define a compound.

- [SDUM] compound name
- [WHAT(1)] amount of the first component
- [WHAT(2)] first component material
- [WHAT(3)] amount of the second component
- [WHAT(4)] second component material
- [WHAT(5)] amount of the third component
- [WHAT(6)] third component material

How to define the "amount"?

By atom relative content, mass fraction or volume fraction.

Material definition: ASSIGNMA

A material must be associated to each of the geometry regions, except to those defined as blackhole.

The command ASSIGNMAt assigns a material to one (or more) region in the geometry

(for the region definition see the geometry lecture or the manual).

The assigned material could be either a single element material or a compound.

A basic input: step 5:Random Seed, START, and STOP

```
TITLE
My Basic Input example
* Set the defaults for precision simulations
DEFAULTS                                                              PRECISIO
* Define the beam characteristics
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAM              3.5 -0.082425      -1.7       0.0       0.0          PROTON
* Define the beam position
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
BEAMPOS           0.0        0.0       -0.1       0.0       0.0
*
GEOBEGIN                                                              COMBNAME
    0    0
* Black body
SPH blkbody     0.0 0.0 0.0 100000.0
* Void sphere
SPH void        0.0 0.0 0.0 10000.0
* Cylindrical target
RCC target1     0.0 0.0  0.0 0.0 0.0 10.0 5.0
RCC target2     0.0 0.0 20.0 0.0 0.0 10.0 5.0
RCC target3     0.0 0.0 40.0 0.0 0.0 10.0 5.0
END
* Black hole
BLKBODY       5 +blkbody -void
* Void around
VOID          5 +void -target1 -target2 -target3
* Target
TARGET1       5 +target1
TARGET2       5 +target2
TARGET3       5 +target3
END
GEOEND
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
MATERIAL         24.0                7.18                               CHROMIUM
MATERIAL          0.0              0.73E-3                              AMMONIA
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
COMPOUND          1.0  NITROGEN       3.0  HYDROGEN                     AMMONIA
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
ASSIGNMA     BLCKHOLE    BLKBODY
ASSIGNMA       VACUUM       VOID
ASSIGNMA      AMMONIA    TARGET3
*
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..*
ASSIGNMA     CHROMIUM    TARGET1    TARGET2        1.0

* Set the random number seed
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
RANDOMIZ          1.0 54217137.
*
* Set the number of primary histories to be simulated in the run
* ..+....1....+....2....+....3....+....4....+....5....+....6....+....7..
START          1000.
STOP
```

A basic input: step 5: Random Seed, START, and STOP

The random number generator is initialized to read a vector of 97 seeds from an external file. Different numbers input will initialize different and independent random number sequences.

A START card at the end of the input file is mandatory.

It defines the number of particle histories required.

The START card is optionally followed by a STOP card, which stops the execution of the program.

Running the code

- Download and Install according to instruction in the manual or in the web page
- ALWAYS define an environmental variable FLUPRO, that points to the installation directory (export FLUPRO=my_path_to_fluka)
- Create your input and run your jobs in an area different from FLUPRO
- RUN:

A script is provided to run fluka:

No. of previous cycle (default is 0)

No. of Last cycle (default is 5)

$FLUPRO/flutil/rfluka -e $FLUPRO/flukahp -N0 -M1 ex1

Specifies the executable name: if it is flukahp in $FLUPRO (default) then it can be omitted

Name of the input file. It must be a file named ****.inp

What rfluka does, and what a cycle is:

Remember:

1. Many primaries are needed to obtain convergence of the results towards the "true"values
2. Statistical errors need to be evaluated
3. Running for more than a few hours is "dangerous" (ops… it crashed after one week…)
4. Monte Carlo uses pseudo-random number sequences

Standard way for FLUKA:

1. Divide a "run" in "batches" or "cycles"
2. Run many of them, keeping track of the random number sequence
3. At the end of the run, sum the results from the cycles and calculate statistical errors
4. Cycles can be added later, continuing in the random sequences
5. Cycles can be run in parallel (multi-core), starting from different initial random seeds

What rfluka does, and what a cycle is:

- rfluka creates a temporary subdirectory: $PWD/fluka_nnnn ($PWD means the current directory) where nnnn is the system process-id assigned to FLUKA. There all necessary assignments are defined and output files are written.
- At the end of the cycle, it copies the output/log/results to the working directory, ADDING A NUMBER to the input file name
- It also copies the random number seed for the next run

Running ex1.inp , after the n-th cycle one gets
ex100n.log, ex100n.out, ex100n.err (n=cycle)
and ranex100m (seed for cycle m = n+1)

One also gets results from scoring, if defined (see lecture). Utilities to sum results are provided.

A brief overview:

Standard Output and Plotting

The FLUKA Standard Output

- FLUKA provides a standard output file that contains plenty of useful information:

  (fortran unit 11, *inp###.out* from rfluka)

- The output file contains the echo and interpretation of your input cards → helps in understanding if what you wrote is what you wanted
- The output file contains the physics settings and threshold settings
- The output file contains basic summary quantities → helps to check that the input is meaningful (..ops, all my particles escaped from the geometry..why?)

- **It must be checked at least once when setting up a simulation and always in case of doubts/crashes**

  (together with *inp###.err* and *inp###.log* files)

The FLUKA Standard Output

Fatal error messages are also printed in the output file.

For example:

```
**** Region n.         4 (TARGS3  ) has no assigned material, run stopped ****
Abort called from PRCHCK reason NO MATERIAL ASSIGNED TO A REGION Run stopped!
STOP NO MATERIAL ASSIGNED TO A REGION
```

Error messages should be of concern to the user.

For more details on possible error messages, see the FLUKA manual.

FLUKA plot

Simulation results can be plotted externally, or via Flair.

Flair makes use of Gnuplot. See Gnuplot manual for all available commands.

Flair also allows Geometry plotting and debugging.

See Flair lecture for more details.

Thanks for your attention!