

Flair Advanced Features

Advanced FLUKA Course



Ceometry

*DIS

n [U,C] natural or instinctive ability (to do something well, /fleə(r)/ to select or recognize what is best, more useful, etc. [Oxford Advanced Dictionary of Current English]

What is flair [1/2]

FLUKA Advanced Interface [http://www.fluka.org/flair]

- All-in-one User friendly graphical Interface;
- Minimum requirements on additional software;
- Working in an intermediate level
 Not hiding the inner functionality of FLUKA

Front-End interface:

- Fully featured Input file Editor
 - Mini-dialogs for each card, allows easy and almost error free editing
 - Uniform treatment of all FLUKA cards
 - Card grouping in categories and card filtering
 - Error checking and validation of the input file during editing
- **Geometry:** interactive visualization editing, transformation, optimizations and debugging (tomorrows talk);
- **Compilation** of the FLUKA Executable;
- Running and monitoring of the status of a/many run(s)

What is flair ^[2/2]

Back-End interface:

- Inspection of the output files (core dumps and directories)
- Output file(s) viewer dividing into sections
- Post processing (merging) the output data files
- Plot generation through an interface with gnuplot;

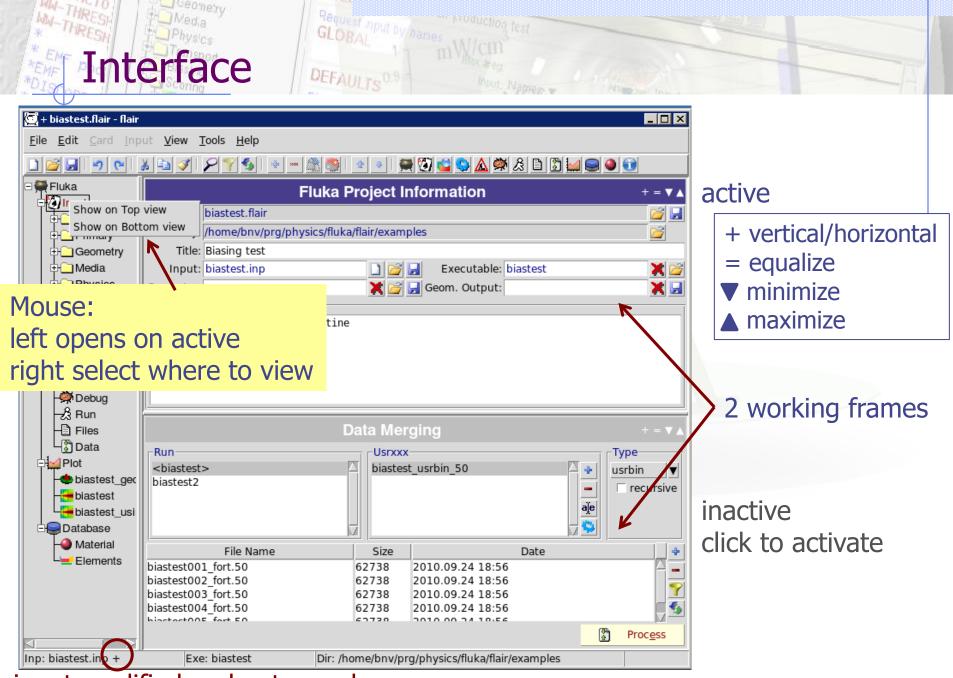
Other Goodies:

- Access to FLUKA manual as hyper text
- Checking for release updates of FLUKA and flair
- Nuclear wallet cards
- Library of materials
- Database of geometrical objects (Not yet completed)
- Programming python API
- Everything is accessible with keyboard shortcuts

Concepts: Flair Project

- Store in a single file all relevant information:
 - Project notes
 - Links to needed files: input file, source routines, output files ...
 - Multiple runs from the same input file, as well running status
 - Procedures on how to run the code
 - Rules on how to perform data merging
 - Information on how to post process and create plots of the results
- You can consider Flair as an editor for the project files.
- Can handle any FLUKA input format (reading & writing), but internally it works using the names format for the input, free with names for the geometry (Recommended way of working)
- The format is plain ASCII file with extension: .flair

Note: If you want to copy a project you need to copy also all linked files especially the input and source routines!



input modified and not saved

Command line options

Usage: flair [options] <filename | filename.flair | filename.inp> Options:

- -d/D Activate/Deactivate the beta-development features
- -e exe
 Use exe as fluka executable
- -g Open geometry editor window
- -i inputfile Fluka input file (w/o the .inp extension)
- -r
 Load most recent project
- -R # Load recent project (number 1..10 or filename)
- List recent projects
- -x Run through an xterm (default)
- -X Run without an xterm (useful in case of start failure)
- -1 Load the first flair file in the folder



Request aput by names GLOBAL 1 mW/cm DEFAULTS 0.9 hour Names v Anaput Inc.

Keyboard:

Almost everything is possible with the keyboard see manual for shortcuts Ctrl-Enter: Execute most important action Ins/Del: Add or Delete

Mouse:

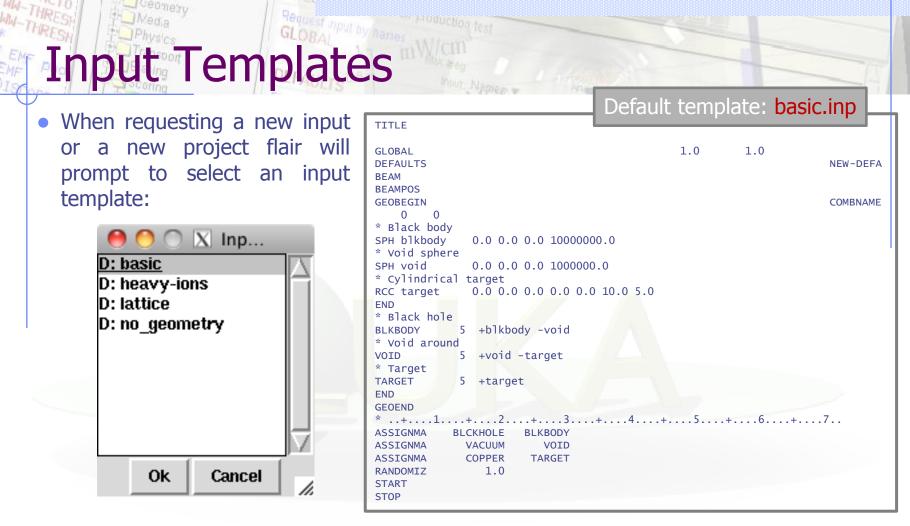
right-click anywhere to get a popup menu

Listboxes:

all listboxes are searchable. Typing only the characters (A-Z) and numbers (0-9) all other are ignored

- LabelFrames:

can be collapsed/expanded by clicking on the label



- Flair default templates are prefixed with "D:"
- User templates will prefixed with "U:"

The user can create his own set of input templates. They are normal FLUKA input be files and they have to be placed in the directory ~/.flair/templates (create the directory if not existing)

Card Categories

For easier access, cards are groups in the following categories:

- **General** General purpose (TITLE, DEFAULTS, GLOBAL...);
- **Primary** Definition of the primary starting particles;
- **Geometry** Cards related to the definition of the geometry bodies/regions/lattices plotting and rotations/translations;
 - Bodies
 Subcategory containing only the bodies definition;
 - Transformations Subcategory containing only the geometrical directives;
- Media Definition and assignment of materials;
- **Physics** Setting physics properties of the simulation;
- **Transport** Modify the way particles are transported in FLUKA;
- **Biasing** Cards for importance biasing definition;
- **Scoring** Cards related to scoring;
- Flair flair special cards;
- **Preprocessor** Definitions for creating conditional input files.

Concepts: Extended Cards [1/2]

• Flair is treating the input file as a list of extended cards;

• Each extended card contains:

- Comment: All commented lines preceding the card(s) as well the inline comments;
- Tag: The 8 character word identifying the card. All tags not recognized by flair will be converted to #error;
- WHATs: Multiple number of WHATs (0=sdum, 1-6 first line, 7-12 continuation line...)
- Extra: multi line string of extra information for special cards like REGION, TITLE, PLOTGEOM etc.
- State (Enable/Disable);
- Flair recognize automatically (and separates them from the comments) all the disabled valid FLUKA cards;

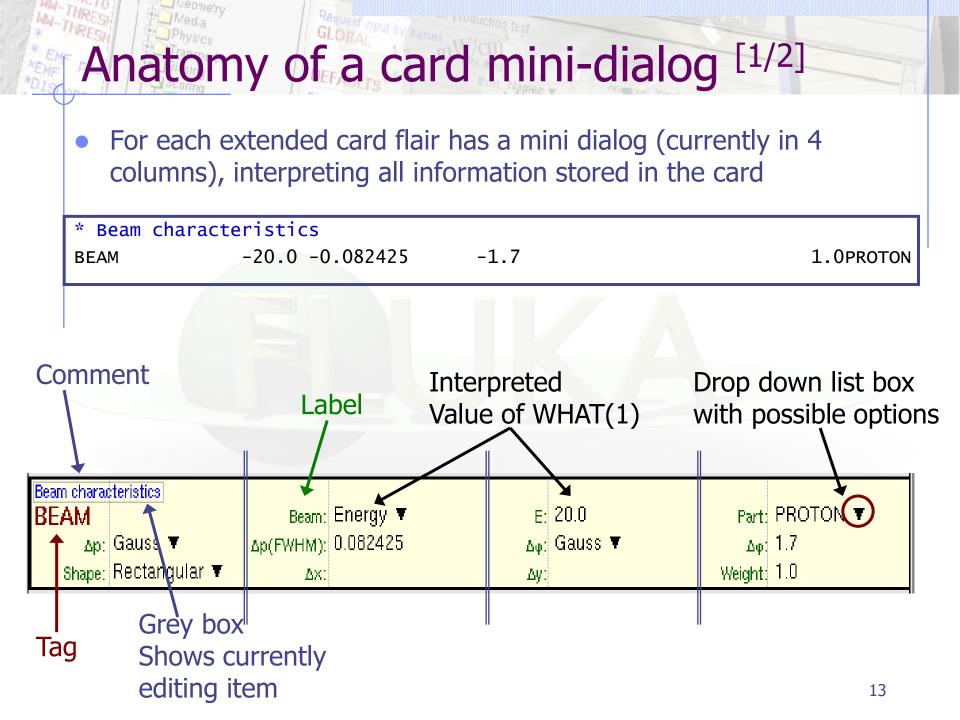
Concepts: Extended Cards [2/2]

- The region definition in the in geometry is emphasized by the presence of a card named "REGION";
- All the COMPOUND cards related to one material are joined in one card;
- Cards are edited with the flair editor through the use of the minidialogs, forcing the user to enter the *correct* information (default);
- The user can nevertheless gain full control of the card using the Edit dialog (*Ctrl-E*);
- Flair will try to find the best floating point representation of each number, to ensure the maximum accuracy; number of digits that fits in the specific width (10 for the fixed format, 22 for the free format).
- Function evaluation: a field value starting with = will force flair to evaluate its content as a function e.g.

BEAMPOS x: =2*10+length

Flair will create a valid fluka input containing the evaluation of the formula and keep the formula inside the comments as

*@what.1 = 2*10 + length



MW-THRE _ ceometry Request nput by names Anatomy of a card mini-dialog [2/2] * Energy deposition in 3D binning 10.0 ENERGY -50.0 45.0 36.0EneDep USRBIN 54.0 -45.0 -54.0 -33.0 100.0 100.0 100.0& USRBIN USRBIN Unit: 50 BIN 🔻 Name: EneDep Type: X-Y-Z 🔻 NX: 100.0 Xmin: -45.0 Xmax: 45.0 Part: ENERGY 🔻 Ymin: -54.0 Ymax: 54.0 NY: 100.0 Zmin: -33.0 Zmax: 36.0 NZ: 100.0 Polypyromellitimide Polyimide, Kapton * chamical \$

* Chemical	O = C H - C C =	: 0		
* Formula		Н-С – С-Н	Н-С - С-Н	
*	/ \C C \			
*	N	N - C C - O	- C C	
*	\ /c c /			
* C H N O		H-C = C-H	H-C = C-H	
* 22 10 2 5	O = C H - C C	= 0		
MATERIAL		1.43	Poly	/imid
COMPOUND	10.0 HYDROGEN	22.0 CARBON	2.0 NITROGENPOly	/imid
COMPOUND	5.0 OXYGEN		Poly	/imid
MATERIAL	Name: Polyimid	#	_{p:} 1.43	
Z:	Am:	A:	dE/dx:	
COMPOUND	Name: Polyimid 🔻	Mix: Atom 🔻	Elements: 6 🔻	
f1: 10.0	M1: HYDROGEN 🔻	f2; 22.0	M2: CARBON 🔻	
f3; 2.0	M3: NITROGEN 🔻	f4; 5.0	M4: OXYGEN 🔻	
f5:	M5: 🔻	f6:	M6: 🔻	

Request input by names Input Editor 1,

seonety

#define	BIAS			
TITLE	Biasing test			
GLOBA	\L	Max #reg:	Analogue: 🔻	DNear: 🔻
		Input: Names 🔻	Geometry: Free V	
DEFAU	LTS	NEW-DEFA 🔻		
BEAM		Beam: Energy V	E: 0.005	Part: NEUTRON V
Δp	Flat 🔻	Δp:	∆¢: Isotropic ▼	
Shape	Rectangular 🔻	Δx:	Δy:	Weight:
BEAMP	POS	X:	у:	Z:
		COSX:	cosy:	Type: POSITIVE V
GEOBE	GIN	Log: 🔻	Acc:	Opt: 🔻
	-	Inp: 🔻	Out: 🔻	Fmt: COMBNAME V
Title	:			
Black bo	· ·			
SPH	blkbody	x: 0.0	y: 0.0	z: 10
		R: 10000000.0		
Void sph	ere			
SPH	void	x: 0.0	у: 0.0	z: 10
		R: 1000000.0		
Cylindric				
RPP	target	Xmin: -100.	Xmax: 100.	
		Ymin: -100.	Ymax: 100.	
		Zmin: -100.	Zmax: 100.	
Black ho				
	BLKBODY		Neigh: 5	Volume:
Expr	+blkbody -void			
Void aro	und			
REGIO			Neigh: 5	Volume:
Expr	+void -target			

SPH blkbody 0.0 0.0 10. 1000000.0

highlight differences during editing

Input Editor - 2

- Drag'n'drop from the TAG of the cards
- Double click on card TAG to select all similar cards
- Editing multiple cards: select cards and modifying the value in one card will propagate the change to all similar selected cards
- Ctrl-Double-Click Show/Hide selected cards
- #if..#endif, \$transform, \$translat or \$expand flair will enclose the selected cards with the #if #endif, or \$start_xxx, \$end_xxx transformation cards
- Popup Balloon tooltip displays short help:
 - for every option on every card
 - body description in the REGION expression
- Right-click: shows popup-menu
 - Quick filtering by REGION, MATERIAL, scoring etc...
- Easter Eggs: AWARI by Double-Right-Click on dialog showing the card representation as text at the bottom of the screen

Input Editor -3

- Automatic indentation of nested #if..#endif and \$start..\$end directives.
- To refresh the display press Ctrl-R
- Each REGION can be split into many cards if needed to be used with preprocessor commands.
- Use as a name "&"

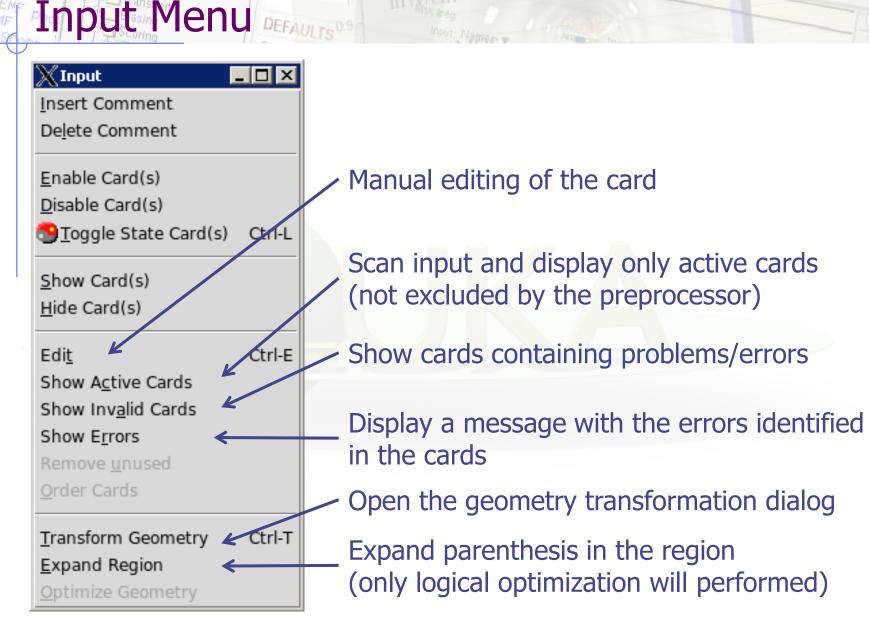
Void around REGION VOID	Neigh: 5	Volume:
	Neight 5	volume.
Expr: +void -target		
#if BIAS ▼		
REGION &	nt: -bias	
#endif		

Input Card Filtering 💥 Filter Cards - 🗆 × Cards-Units-Materials Category *all* 01 - RANDOMIZ DEFAULTS ALUMINUM General GEOBEGIN ARGON 05 - GEOBEGIN GEOEND 09 - LOW-NEUT Primary BERYLLIU 11 - GEOBEGIN Geometry GLOBAL BLCKHOLE ...Bodies OPEN 13 - EMFFLUO CALCIUM Media RANDOMIZ CARBON 15 - GEOBEGIN Physics REGION 17 - DETECT COPPER Transport RPP 19 - DPMJET GOLD SPH 30 - OPEN HELIUM Biasing Scoring START 50 - USRBIN HYDROGEN -Bodies-Regions Particles-Detectors 3-HELIUM blkbody BLKBODY Flux target TARGET 4-HELIUM void VOID ACTIVITY @LASTREG ACTOMASS AKAONZER ALAMBDA ALAMBDC-ALL-CHAR ALL-NEGA ALL-NEUT Ok Cancel Reset

Request input by/name

Ceonetry

 Filter Cards dialog allows a more advanced selection of cards to be displaced, by showing only the cards that match the selected options



Request input by name

Manual Card Editing

Kedit Card						_ 🗆 ×
Comment:	Primary particle def	initi	on			
Card:	BEAM	▼Lin	es: 1 -	sdum:	NEUTRON	
		<u> </u>				<u>-</u>
	005	<u> </u>	2:	 	10000.0	_
4:		V	5:	▼ 6:		▼
Extra:						
	1					

Accessible: Ctrl-E or right-click \rightarrow Edit, Menu \rightarrow Input \rightarrow Edit

Lines: Number of lines the card extends

- Extra: additional information for a card like title string for TITLE, or region expression for REGION
- Dropdown box: shows with categories all items defined in the input (bodies, regions, materials, particles...)

Bodies Transformation

X Transform t	oodies	- 🗆 ×
ROT-DEFini:	▼ Add to Input Get from Input	
Туре	Value	- +
RZ	90.0	
Zero: 1e-10	Accuracy: 15	-
Infinite: 1000	00.0 🔽 Use QUA (instead of RCC/RE	C)
	Transform	Close

Transformation Types:

Ttranslate along a vectorTX TY TZtranslate along axisRX RY RZaxis rotation (degrees)Sscaling

- Applies a user transformation to the selected bodies on the input editor.
- Convert transformations to/from ROT-DEFini cards
- Zero: limit below which to be considered as zero
- Accuracy: Numeric digits
- Infinite: infinite bodies when converted to which size to use
- Use QUA: convert infinite cylinders to infinite QUAdrics

Remember:



When transforming bodies for

use with LATTICE card, use the maximum precision

Color Palette

🔀 💿	Pale	ette	\odot	\otimes
Material	Global	Local	Alpha	1
ADTISSUE		#FFE1C8	0	4
AIR	#E0F4FF		100	
ALUMINUM	#949399		0	-
ARGON	#65EDFE		100	
BERYLLIU	#CFBEBA		0	
BLCKHOLE	#404040		0	4
BONECOMP	#E1C498		0	
BONECORT	#CEA178		0	
CALCIUM	#CCCCCC		0	
CARBON	#606060		0	
CONCRETE		#A0A0A0	0	:
COPPER	#B87333		0	-
GOLD		#FFD700	0	
HELIUM	#40E0D0		100	
HYDROGEN	#40E0D0		100	
IRON	#925239		0	
KAPTON	#FDDC13		0	
LEAD	#909098		0	
MAGNESIU	#9AA387		0	
MERCURY	#CC9B59		0	
MUSCLESK	#DA755B		0	z
		Ok	Canc	el

Accessible: Menu \rightarrow View \rightarrow Palette

- Edit colors used for material display in Geometry plots and GeometryEditor
- Global colors are saved inside flair.ini and are shared between all projects
- Local colors are initially randomly assigned and saved inside the project file
- Alpha channel for setting transparency on materials. Used on the 3D raytracing plotting.

0 = opaque

100 = transparent



Comp	oile Exec	cutable +	+ =
File	Size	Date	
usimbs.f	4856	2010.09.24 18:57	
	5751	2010.09.24 18:40	
oauxfi.f	6059	2010.09.24 18:49	

automatic selecting needed routines from usermvax/

Request nput by nen

DEFAULTE

🗙 FLUKA Us	ser routines		
File▲	Size	Date	Desc
pshckp.f	1274	2005.06.02 13:16	
queffc.f	1605	2005.03.24 10:40	quantum efficiency (for optical photons)
rflctv.f	1469	2005.03.24 10:40	reflectivity (for optical photons)
rfrndx.f	1469	2005.03.24 10:40	refraction index (for optical photons)
soevsv.f	2507	2005.06.17 16:13	saving source events
source.f	7327	2009.09.09 16:08	to generate any distribution for source particles
stupre.f	4223	2005.03.24 10:40	set user variables (electrons and photons)
stuprf.f	1981	2005.07.25 13:43	set user variables (hadrons, muons and neutrin
ubsset.f	5585	2005.03.24 10:40	to override input biasing parameters
udcdrl.f	2425	2005.03.24 10:40	decay direction biasing
usimbs.f	3262	2008.10.30 11:56	user-defined importance biasing
usrein.f	1553	2005.03.24 10:40	event initialisation
usreou.f	1480	2005.03.24 10:40	post-event output
f	1050	2006 00 11 15-20	
Link: Ifluka	3	🔻 Exe: biastest 🛛 🛃 💥	✓ Default main:
tions:			🗆 D Line 🔽 Bound Check
			<u>A</u> <u>B</u> uild <u>C</u> ompile <u>C</u> lean

Filetypes accepted:

- Fortran: .f, .F, .for, .FOR
- C/C++: .c, cpp, .cxx, .cc
 - Libraries: .a, .so

Automatic scanning of necessary user routines and copying them to project folder. Build: behaves like a "makefile" compiles based on files timestamp when are newer

Compile: Forces compile of the selected files

Clean: cleanup of all produced files

When you are unsure, click on "Clean" before "Build"



			Run Fluka			+ = ▼ ▲	
-Run / Input		Override	Options				/in
<rdsource></rdsource>	4	Title					<in< td=""></in<>
rdsource_first rdsource_second	-	Primaries	0	Rnd 0			inp
		Time	0	Exe rdsource		🛛 🗙 📂	
	<u>.</u>	Defines	Default Defines				edit
	aje	Sel	Name		Value		
			FIRST			\square	Cre
			SECOND				00
	-		ANALOGUE ENERGY	=10*MeV			on
							ove
		C	ustom prepro	ncessor	#defines		Ove
		Č					
			even with w	/alue/fu	nction		
				anacyra			
		I				/	•
Cycles: Continue Previou	ıs 0	🖨 No	. Cycles 5 🔮 Last 5				• F
🔏 Run Stop Cycle	2	Stop Run	Kill Attach <u>R</u> efresh	Queue *Defau	lt 🔻		
Progress							• 5
Status: Finished OK		Inp	ut: rdsourcePrecision_secon	d	Dir:		
Started:		E	TA:		Time/prim:		• E
Elapsed:		Cy	cle:		Run:		
Cycles:							• E
Primaries:							

Request oput by name

DEFAULTE

<inputname> refers to the input file AS IT IS in the input editor.

Create additional runs based on the same input file by overriding:

Title

- Preprocessor definitions
- Random number seed
- Starting particles
- Execution timeout
- Executable
- Monitors the status of the run by inspecting the FLUKA output files. If timeout occurs try to re-Attach to the running process.
- The timeout is user-definable in the Preferences dialog

Running: How to use multicore CPU's

- Create clones of the current input e.g. test.inp named: test1.inp, test2.inp, test3.inp ...
- Assign a different random number seed on each run (Rnd entry)
- Select all in the listbox and click Run

Multiple Selection:

- To modify many runs at the same time, select them in the listbox
- The options will be "disabled"
- Right-click on the options you want to enable and modify them
- Modify the filters in Data processing for summing up all cycles from all runs (see later)

	Output Files			+ = ▼ .
Run		Cycles		
<rdsource></rdsource>		/ 001		7
rdsource_first		002		
rdsource_second		003		
rdsourcePrecision_first		004		
rdsourcePrecision_second		005		
		006		
		compile		
		data		
		input		_
		plot		
		temporary		
Filter: *.out	Files patt	ern		
File	Туре	Size	Date	
dsource first001.out	FLUKA out	138608	2012.09.05 16:29	
dsource_first002.out	FLUKA out	138608	2012.09.05 16:31	
rdsource_first003.out	FLUKA out	138608	2012.09.05 16:33	
dsource_first004.out	FLUKA out	138709	2012.09.05 16:36	
dsource_first005.out	FLUKA out	138703	2012.09.05 16:38	
nohup.out	FLUKA out	22766	2012.09.05 16:42	
dsource_second001.out	FLUKA out	148916	2012.09.05 16:40	
dsource_second002.out	FLUKA out	148916	2012.09.05 16:40	
dsource_second003.out	FLUKA out	148916	2012.09.05 16:41	
dsource_second004.out	FLUKA out	148916	2012.09.05 16:41	
rdsource_second005.out	FLUKA out	148916	2012.09.05 16:42	
nohup.out	FLUKA out	22766	2012.09.05 16:42	
dsourcePrecision_first001.out	FLUKA out	139076	2012.09.05 15:37	
rdsourcePrecision_first002.out	FLUKA out	139170	2012.09.05 15:43	
rdsourcePrecision_first003.out	FLUKA out	139076	2012.09.05 15:49	- L.
rdsourcePrecision_first004.out	FLUKA out	139177	2012.09.05 15:55	

Output Files

Delete selected files

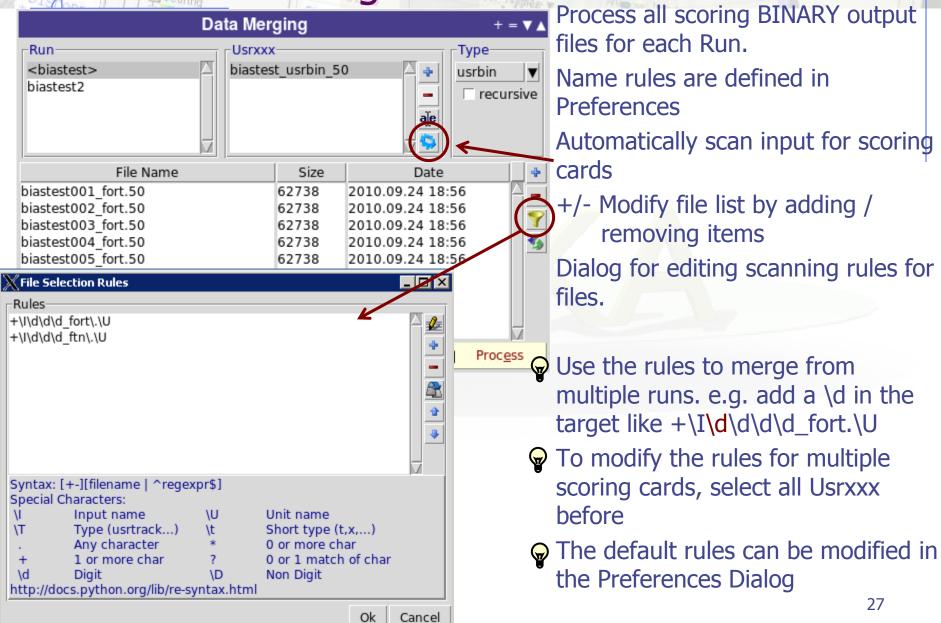
Inspect Output files generated by FLUKA classified per: Run/Cycle As well special output files from compilation data processing plotting and temporary

Double clicking opens:

- Files in the file Viewer
- coredumps in debugger

Right click can convert USRBIN's from formatted to unformatted

Data Processing





	Plot List		
File geometry enedep fluence resnuc	Title nTOF Target Geometry Deposited Energy Particle Fluence Resitual Nuclei	Type Geometry USRBIN USR-1D RESNUCLE	 Plots can be created in the list frame. Either Add new or Clone from existing on It is important to set a un filename for each plot. The filename will be used for auxiliary file that the plot (the extension will chang) The Filter button creates automatically one plot for processed unit (From the input file) Hit Enter or click the Edit display the plotting dialog.
			 Fast Double click on item

Request input by names

DEFAULTE

Plot Types

- Geometry For geometry plots
 - USRBIN For plotting the output of USRBIN
- USR-1D To plot single differential quantities from cards USRBDX, USRTRACK, USRCOLL, USRYIELD
- USR-2D To plot double differential from USRBDX
- To plot 1d or 2d distributions of RESNUCLEi RESNUCLE
- **USERDUMP** To plot the output of USERDUMP. Useful for visualizing the source distribution (ToDo)

he "Plot" w plots nes. inique

- his every t needs ge)
- or each e default
- t icon to g
- n to open the corresponding dialog
- Slow Double click to modify the value

100	Less Single Silleren	and Print	
Plot Title: Particle Fluence			
Opt: font 'Times,20'	Header File: flu	ience	
Axes Labels		Set	Size / Multiplot
X: Energy	Opt: font 'Helvetica,14'	📕 grid	👅 auto 🛛 🗙
Y: Fluence (dn/dlnE/7e12p)	Opt: font 'Helvetica,14'	👅 legend	ratio: Y:
Axes Range			
👅 log X:	log X2:	-	⊒ show <u>Get</u>
📕 log Y: 🛛 🚽 🚽	💷 log Y2:	-	show Reset
Hada	(industry)ala		
	And the second second second		10
	New II when it hands	State State	
tere i i i i i	a hardel	d vite	100 C
110	Participation and an and		
1.4	THE REP. 12 No.	nan (C. S. K.	- 1 m
	man and the second s	an i the	10 T
	Augusta 11	and the second	
Gnuplot commands			
	Footer		Plot
			🚽 .eps 🔻

otting Frames

All plot types share some common fields:

Title + options, Filename, Axis Labels, Legends (Keys) and Gnuplot Commands.

Plot button (Ctrl-Enter) will generate all the necessary files to display the plot, ONLY if they do not exist.

Re-Plot will force the creation of all files regardless their state

Check the gnuplot manual to provide additional customization commands: e.g. To change the title font to Times size=20, add in the Opt: field the command: font 'Times,20'

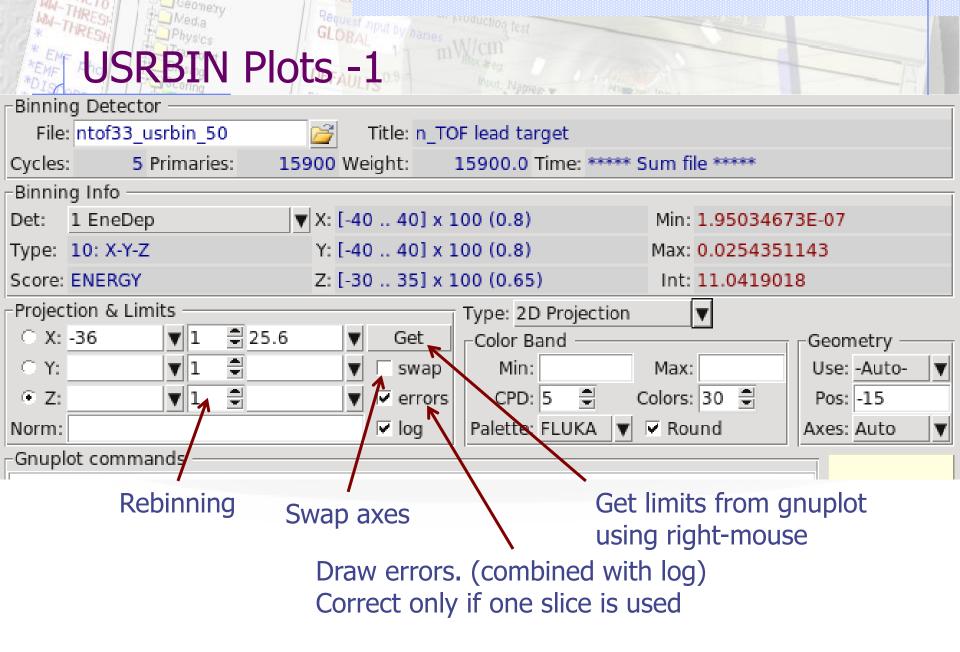
Look in the flair manual for a short reference of gnuplot commands

In the Configuration Dialog you can set global commands to execute before or after any plot

- The output window displays all the commands that are sent to gnuplot. As well as the errors. In case of problem always consult the output window!
- In the Gnuplot commands you can fully customize the plot by adding manually gnuplot commands:
- Special commands:

General Tips

- plot, splot with no options, defines the order where flair should insert the plot or splot command.
- replot <plot-cmd> append extra plots to the one generated by flair



WW-THRESH	Physics	Request nput by names	roduction test			
*EMF USF	RBIN Plot	SAUL 2	Max and Input Natirez			
Binning Detect	or					
File: ntof33_u	usrbin_50	🎽 🛛 Title: n_TO	F lead target			
Cycles: 5	Primaries: 159	00 Weight:	15900.0 Time: ****	* Sum file *****		
Binning Info —						
Det: 1 EneDe	> ▼	X: [-40 40] × 1	00 (0.8)	Min: 1.9503467	'3E-07	
Type: 10: X-Y-Z		Y: [-40 40] x 1	00 (0.8)	Max: 0.0254351	.143	
Score: ENERGY		Z: [-30 35] x 1	00 (0.65)	Int: 11.041901	.8	
Projection & Lin	Projection & Limits Type: 2D Projection					
○ X: <u>-</u> 36	▼ 1 🛢 25.6	▼ Get	Color Band		Geometry —	
• Y:	V 1 🕄	🔻 🗆 swap	Min:	Max:	Use: -Auto-	
• Z:	V 1 🗘	🔻 🔽 errors	CPD: 5 🚔	Colors: 30 🛢	Pos: -15	
Norm:		🔽 log	Palette: FLUKA	🔽 Round	Axes: Auto	
-Gnuplot comm	ands ———		1			

Normalization could be used as:

- number or expression evaluating in a number 65e-3/2.7
- function with x as variable. e.g E2T(x*65e-3/2.7)-293 with the function defined in the Gnuplot commands E2T(x) = ((3.00629e-08*x-0.000108436)*x+1.01097)*x+311.839

MW-THRES Request input by names **SRBIN** Plots - 3 hout Names w Binning Detector File: ntof33 usrbin 50 Title: n TOF lead target 2 5 Primaries: 15900 Weight: 15900.0 Time: ***** Sum file ***** Cycles: Binning Info ▼ X: [-40 .. 40] x 100 (0.8) Min: 1.95034673E-07 Det: 1 EneDep Y: [-40 .. 40] x 100 (0.8) Type: 10: X-Y-Z Max: 0.0254351143 Score: ENERGY Z: [-30 .. 35] x 100 (0.65) Int: 11.0419018 Projection & Limits Type: 2D Projection ₹ 25.6 ○ X: -36 **v** 1 Get Color Band Geometry - \bigcirc Y: **v** 1 Min: Max: Use: -Autoswap V Colors: 30 🚔 • Z: **v** 1 CPD: 5 Pos: -15 errors ¥ Palette: FLUKA Round log Axes: Auto Norm: T -Gnuplot commands

Normalization could be plotted:

- 2D projection, 1D projection
- Trace of the maximum
- Full width at half maximum

WM-THRESH Physics Request nput by names	production test
USRBIN Plots - 4	Nour Netree - Anapus Inc.
Binning Detector	
File: ntof33_usrbin_50 🛛 🚰 Title: n_T	OF lead target
Cycles: 5 Primaries: 15900 Weight:	15900.0 Time: ***** Sum file *****
Binning Info	
Det: 1 EneDep ▼ X: [-40 40] x	100 (0.8) Min: 1.95034673E-07
Type: 10: X-Y-Z Y: [-40 40] x	100 (0.8) Max: 0.0254351143
Score: ENERGY Z: [-30 35] x	100 (0.65) Int: 11.0419018
Projection & Limits	Type: 2D Projection
⊂ X: -36 🛛 🔽 1 🚍 25.6 🖤 Get	Color Band Geometry
○ Y: ▼1	Min: Max: Use: -Auto-
• Z: ▼1 € • ✓ errors	CPD: 5 🖨 Colors: 30 🖨 Pos: -15
Norm: 🔽 log	Palette: FLUKA 🔻 🗷 Round 🛛 🗛 🛪 Axes: Auto 🔻
Gnuplot commands	

Geometry plot overlay (useful for LATTICE's):

-Auto- generates automatically from FLUKA a geometry at the middle position of the projection

otherwise you can use any existing geometry plot from the drop down list. Be carefull to proerly match the axes that you are using

Configuration Dialog: Programs

<u>ଚ</u>	Preferences		\odot \odot \otimes
Programs Interface Input Data Gnuplot Geometry Fonts Colors	Default Fluka Var Fluka Directory rfluka Executable flukahp fff USBMAX trace scan Gplevbin projection USBREA usrbin to ascii Submit Command Kill Command Kill Command Viewer Editor Terminal File Explorer Debugger Gnuplot	FLUPRO flukahp /usr/bin/nohup /usr/bin/emacs xterm konqueror gdb gnuplot	
	Ok Cancel	Help	

- Set FLUKA directory
- Override default programs to use
- Processing programs are in the "Data" section

Configuration Dialog: Interface

Programs □ Skip About dialog Input □ Show tips dialog Data □ Show icon toolbar Gnuplot □ Show status bar Geometry □ Show frame title Fonts □ Remember last frame □ Show fluka files in fluka_XXX dir □ Keep backups □ Cleanup temporary files Key time threshold 1000 (ms) Balloon delay 1500 Temporary prefix flair_ Attach timeout (s) 120 Refresh Interval (s) 15 Time format %Y.%m.%d %H:%M	Q .	Preferences	<u>></u> ×
	Interface Input Data Gnuplot Geometry Fonts	 ✓ Show tips dialog ✓ Show icon toolbar ✓ Show status bar ✓ Show frame title □ Remember last frame □ Show fluka files in fluka_XXX dir ✓ Keep backups ✓ Cleanup temporary files Key time threshold 1000 Balloon delay 1500 Temporary prefix flair_ Attach timeout (s) 120 Refresh Interval (s) 15 	

- General interface settings
- Keep backups when files are saved as (file~)
- Automatically Cleanup temporary files. Disable only if you want to inspect files after Debug or Plot when an error occurs
- Key time to reset the type-in search in listboxes
- Balloon delay time
- Time format for files (follows python&C syntax)
- Time out to attach to a running simulation
- Automatic refresh interval of information 36

Configuration Dialog: Input Editor

Q 🔾	Preferences	\odot \odot \otimes
Programs Interface Data Gnuplot Geometry Fonts Colors	 Show comment scale Insert comment Show pre-processor cards Enable drag 'n drop Auto Body Insert Sort Region and Material List Show card interpretation Pad space 0.35 Label Fraction 	
	Ok Cancel Help	

- Show alignment scale
- Automatically insert comment
- Always display preprocessor cards
- Enable drag'n'drop
- Automatic body insertion while editing the region expression
- Sort the region and material list
- Display card interpretation at the bottom of the screen

Configuration Dialog: Data

Q 💿	Prefere	nces 🛛 😒 🖄
Programs Interface Input Data Gnuplot Geometry Fonts Colors	Type▲ detect resnuclei usrbdx usrbin usrcoll usrtrack usryield +\I\d\d\d_fort\.\ +\I\d\d\d_ftn\.\U	
	USRBIN process USRTRACK proc USRBDX proces USRYIELD proce RESNUCLEI pro DETECT process	essing 222 sing 222 essing 222 cessing 222
	Ok Can	cel Help

- Define how to generate the automatic filenames
 - \I will be replaced by input
 - \T by card name
 - \t by card character

<mark>usrbd</mark> x	X
usrbin	b
usrcoll	С
usrtrack	t
usryield	У
resnuclei	r

\U the abs(unit-number)

Configuration Dialog: Gnuplot

Q .	Preferences	۲	٢	×	
Programs Interface Input Data Geometry Fonts Colors	Terminal: Global Commands File Types Type▲ Settings .eps postscript eps enhanced color .gif gif transparent medium .jpg jpeg transparent medium .pg postscript enhanced color .ps postscript enhanced color				
	Ok Cancel Help				

Terminal:

additional options to supply to default terminal

Global Commands:

gnuplot commands to be executed before any plot

File Types:

Right-click: to Add/Delete/Modify file types.

Configuration Dialog: Geometry

Q .	Preferences	\odot \odot \otimes
Programs Interface Input Data Gnuplot Geometry Fonts Colors	Laptop Mode Zero 1e-10 Infinite: 10000000000.0 Accuracy: 15	
	Ok Cancel Help	

Laptop Mode:

check to swap middle with right mouse buttons. Middle button is used in GeometryEditor for panning, zooming, rotating etc... Zero: Infinite: Accuracy: same as in the Bodies Transformation dialog

Materials Database

				Material Database		+ = 1	▼ ▲	search database
Searc	h:						9	
Grou	р—		Material	List				
Biolog Eleme Gener ICRU Impla Liquic Metal Plastic Targe	ents ral ds / G l Alloy cs / P ets	iases /s /olyme	ers Skeletal Lead Thallium Cyclobu 1-Chloro Sodium	clohexanone Muscle (W&W type 1) n tane	Density 13.546 0.9478 1.05 11.35 11.72 0.00125 0.8862 2.261	Stoichiometry Hg H-10, C-6, O-1 H-10.1, C-17.1, Pb Tl H-8, C-4 H-9, C-4, Cl-1 N-16.5, O-56.5,	÷	insert material to input add/del material edit material
		rcury						
Notes	5:				MERC		àje ⇒	add names to be used by FLUKA
	chiom		Properties			ſ		
Comp Z 80	positi A	on: <u>m</u> El Hg	ass ▼ liquid Name Mercury	▼ 13.546 Frac	Grou		Ok	Modify Stoichiometry and properties of materia

WARNING: When modifying the database a local copy will be created in ~/.flair folder!!!

AM-THRESS MA-THRESS AM-THRESS Physics Periodic Table ULTS⁰¹ Market Provide Annual Content Physics Periodic Table ULTS⁰¹ Market Physics Tradictore Table ULTS⁰¹ Market Physics M

💥 Table of Elem	ients	;																	_ 🗆 ×	1
Table List																				
Group	1	2		3	4	5	6	7	8	9	10	11	12	2 13	14	15	16	17	7 18	1
Period																				
1	1																		2	
1	н																		He	
2	3	4												5 B	6	7	8	9		
	Li	Be													С	N	0	F		
3	11 Na	12 Mg												13 Al	14 Si		16 S	17 C		
	19	20	i –	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	5 36	
4	K	Ca		Sc	Ti	V	Cr	Mn	Fe	Co				-Hg Merc						
5	37	38	i	39	40	41	42	43	44	45	4							Z: 80	0	
Э	Rb	Sr		Y	Zr	Nb	Мо	Tc	Ru	Rh	P					Atomic Weight: 200.59 (2)				
6	55	56	*	71	72	73	74	75	76	77	7					Density: 13.546 c				
0	Cs	Ba		Lu	Hf	Ta	W	Re	Os	Ir	F			M		Melting: -38.83				
7	87	88	**	103	104	105	106	107	108	109	1									
	Fr	Ra		Lr	Rf	Db	Sg	Ns	Hs	Mt	<	N	/lerci	ury	>	Oxidation: +1,+2				
* Lanthanides			*	57	58	59 Da	60 Not	61	62	63	6,	4	Jπ	∆ (MeV)	Т1/2, Г	, Abunc	dance	e Dec	ay Mode
				La	Ce	Pr	Nd	Pm	Sm	Eu	<u> </u>	5		-8.2s) ms (+4	-		А	-
** Actinides			**	89	90	91	92	93	94	95	g176	5 0+		-11.80		ms (+1				L00%
				Ac	Th	Pa	U	Np	Pu	Am	C 17			-12.7		130 s (5				%,EC 15
											178			-16.32		254 s (1	9)			70%,EC
		H - g	as			Li - so	id			Br - lic	uid 180	9) 0+		-17.0s -20.2s		09 s (4) 0 s (2)				53%,EC 1%,A 49
		Non-		əle		Transi	tion Me	stale			art 18			-20.25 -20.75		6 s (3)				4%,A 36
		NOI1-	met	als		riansi		cdis			183	> 0+		-23.5s		0.83 s (6)			4.8%,A 1
		Alkal	i Me	tals		Alkali I	Earth M	/letals		Other	Me 183	3 1/2		-23.9s		4 s (7)	,			4.5%,A 2
	_										184			-26.2s		.6 s (3)				8.89%,A
											c) /	At 20 C	2.							
											-/ /									

Import / Export Importing

- Input: merge parts or entire input file with the current
- Mcnp: import mcnp geometry into FLUKA. (experimental)

Exporting

- Gnuplot: save active plot to a gnuplot script
- Makefile: create a makefile for compiling the executable
- Mcnp: save input in MCNP format: Geometry, Materials, Importances
- Povray: save geometry into povray 3D format

Programming Interface: API

There is work presently going on to decouple the functionality from the interface, some of the basic classes can be used to input processing

file: Input.py - to manipulate input files import Input Input.init([database]) to initialize the database of cards Most commonly used classes:

Card containing the description of each card Input manipulating the FLUKA input file

file: Project.py - to manipulate project files

Constructor: Input.Card(tag, what [,comment [,extra]]) what is a list starting with what[0]=sdum

Important Methods:

API: class Card

setWhat(n, value) nwhats() what(n) numWhat(n) intWhat(n) clone() setEnable(e) set value to what#n return number of whats return value of what#n return numeric value of what#n return integer value of what#n return a copy of the card enable/disable card

API: class Input

Constructor: Input Input() initialize the structure to hold an input file Important Variables:

cardlist cards

Important Methods:

read(filename) write(filename) addCard(card,pos) delCard(pos) preprocess() setEnable(e) a list with pointers to cards a dictionary with pointers to cards grouped per tag

read input from file write input to filename add card to position pos (or end of file) delete card from position pos preprocess input to check for active cards enable/disable card

API: class Project

Constructor: Project.Project() initialize the structure to hold a project file Important Methods:

clear() load(filename) save([filename]) runCmd(run) to re-initialize project load project from file filename save project to filename create run command

API: example

Read an input file and modify the random number seed

```
import Input
Input.init()
input = Input.Input()
input.read("test.inp")
try:
  rndcard = self.cards["RANDOMIZ"][0]
   rndcard.setWhat(2,5723)
except:
   print "No RANDOMIZe card found"
  sys.exit(0)
input.write("test2.inp")
```

