



flair advanced features

Starting flair

`flair -h`

shows the command lines arguments

Most important:

- filename doesn't need to be complete it will load the best match, e.g. `flair course` or `flair course.f`
- `-r` loads the last opened project
- `-l` shows the last 10 projects

```
Usage: flair [options] [filename]...
```

```
Options:
```

```
-l          Load the first flair file in the folder
--compile  Compile executable
-d/-D      Activate/Deactivate the beta-development features
--data #   Process/Merge the data files of all or specific
           runs (no default). Accepts patterns or comma
           separated names e.g. * or foo* or foo,bar*
--exe file  Fluka executable. (default: {FlukaDir}/bin/fluka)
-h|-?|--help print this help page
-i/--ini file          Alternative configuration file
                    (default: $HOME/.flair/flair.ini)
-l | --list List recent projects
-p/-P      enable/disable profiling
-m #       Open a new project in mode
--plots    Do all plots and save the files
-r|--recent Load most recent project
-R #       Load recent project (number 1..10 or filename)
-s         Skip About dialog
-t # | --type # force import file type to load if different
           than .flair. Accepted types: flair fluka gdml
           mcnp moira penelope pickle
-u | --update          Recalculate and save input file variables
-v/-V | --verbose     Increase/Decrease verbosity level
--noansi    Disable ansi coloring on dumps
```

```
filename[s] flair project, input, imported files or directory
automatically detect type from file extension
```

```
extensions supported: <none|.flair>, .inp, .fluka, .moira, .mac[ro],
.gdml, .pickle, .mcnp
```

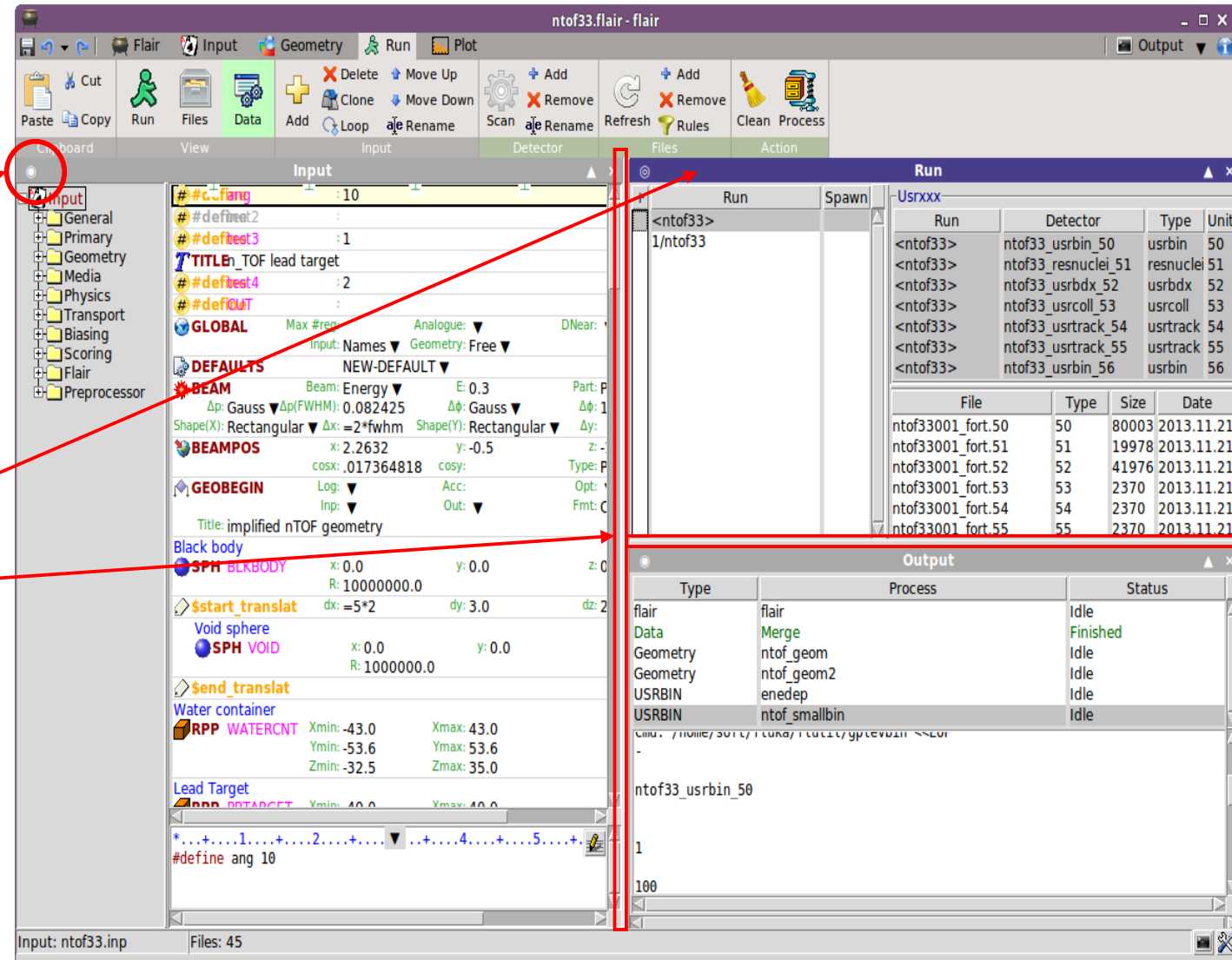
Interface customization – Docking & Tiling

Tiling

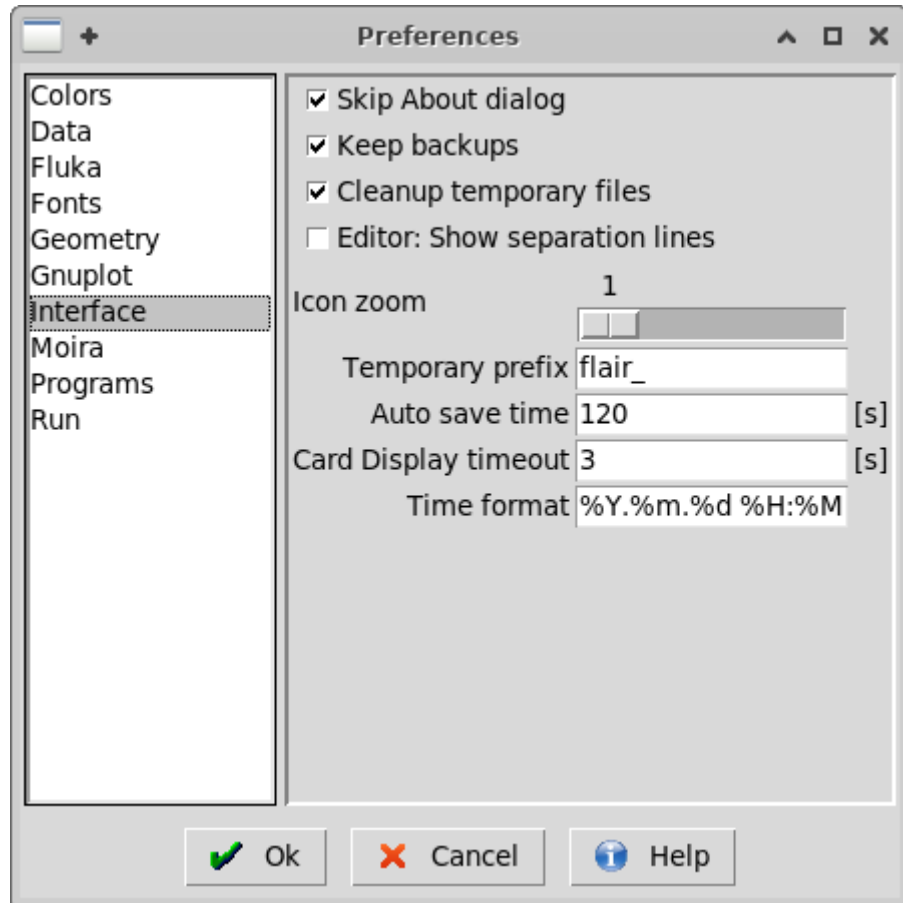
- Multiple pages can be displayed at the same time.
- Pin the page(s)
- Select a new tab
- Drag the tag title bar to rearrange
- Resize the separation
- flair will remember the layout

Undocking

- Drag the “tab” outside flair to open the tab in a new window

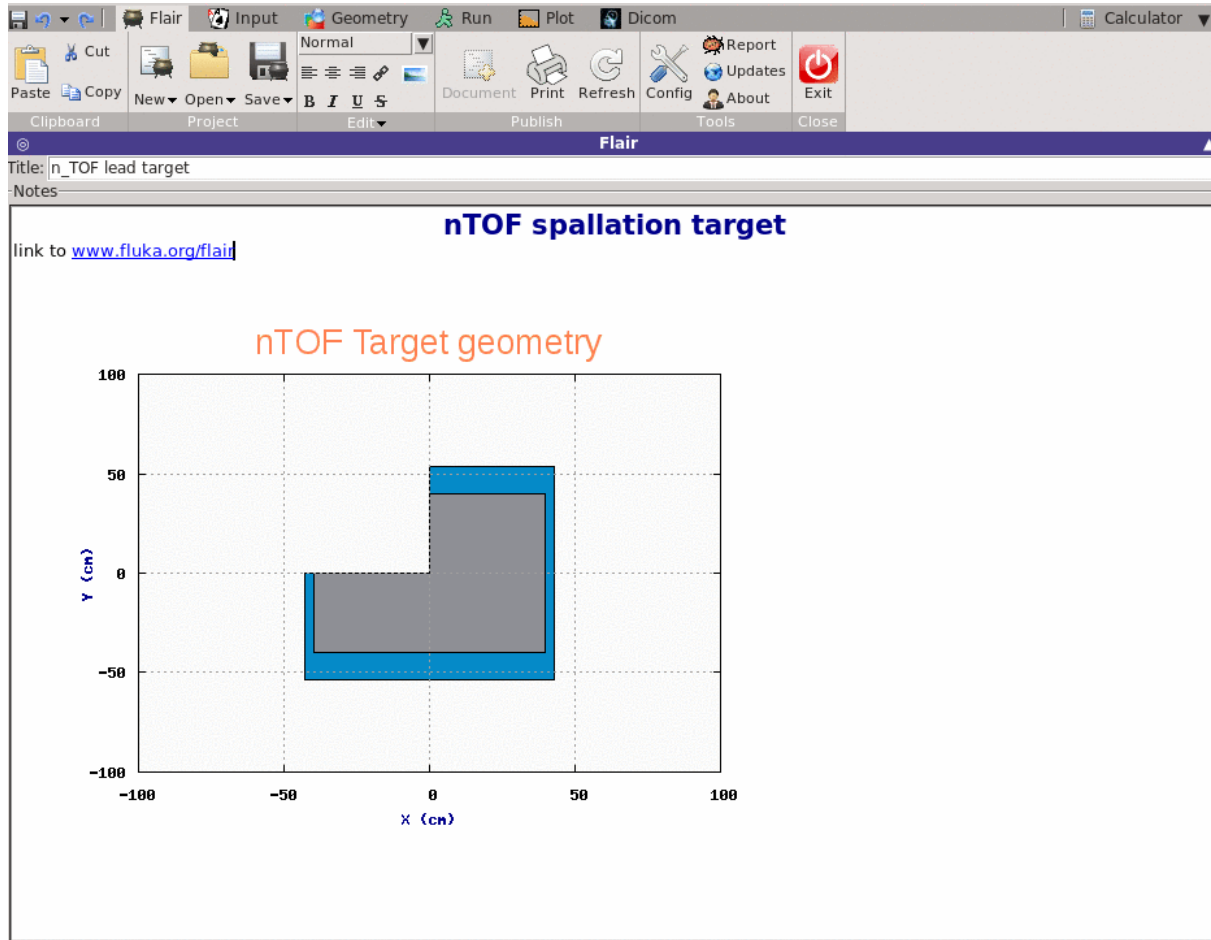


Configuration



- Almost everything is configurable:
 - Colors all colors used by flair
 - Data processing programs and filters
 - Fluka Fluka programs configuration
 - Fonts all flair fonts (except Geometry)
 - Geometry CPUs and precision
 - Gnuplot global commands and terminal
 - Moira ... to come ...
 - Programs editor, terminal, debugger...
 - Run Spawning rules & Queues

Project notes page



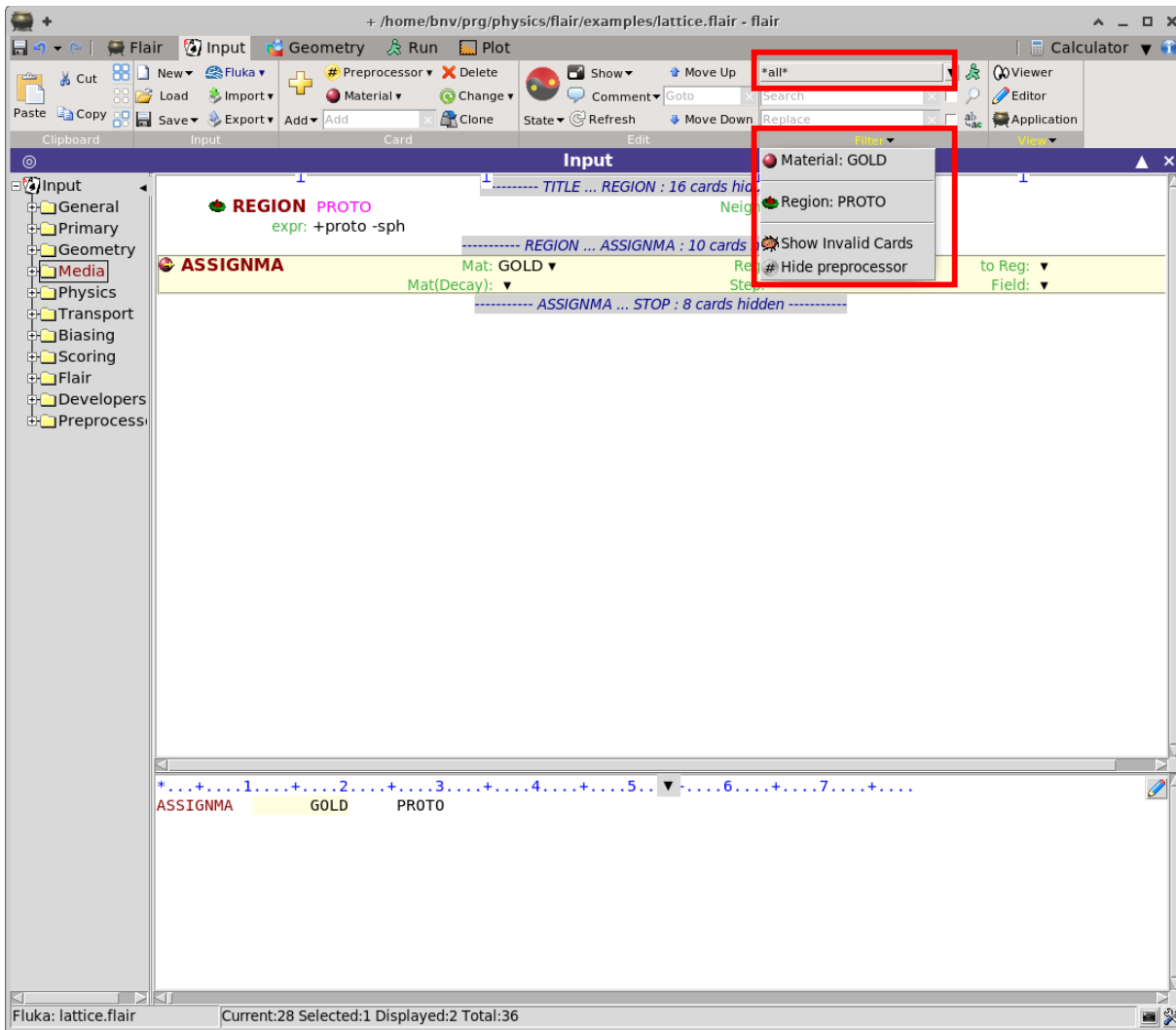
- Notes editor provides some basic formatting capabilities
- geometry viewport can be exported to notes
- flair Plots can be inserted here
→ *updated automatically when recreated*

Input Editor: multi-card editing

GEOBEGIN		Accuracy:	Option:	Paren:
Title:		Geometry:	Out:	Fmt: COMBNAME
SPH	blkhole	x: 0.0 R: 1000000.	y: 9999	z: 0.0
SPH	void	x: 0.0 R: 100000.	y: 9999	z: 0.0
RPP	proto	Xmin: -5. Ymin: -5. Zmin: -5.	Xmax: 5. Ymax: 5. Zmax: 5.	
SPH	sph	x: 0.0 R: 2.0	y: 9999	z: -0.05
XYP	cutz	z: 0.0		
\$start_transform		Trans: -trans		
RPP	replica	Xmin: -5. Ymin: -5. Zmin: -5.	Xmax: 5. Ymax: 5. Zmax: 5.	
\$end_transform				

- Selecting multiple cards:
 - Shift + Left-mouse-button
 - Control + LMB
 - Shift + <Up/Down/PgUp/PgDn>
- Any change in the active card will be propagated to all SIMILAR cards (same tag) for the same WHAT.
- Two-undo commands will be registered
 - first for all other cards
 - second for the current active

Input Editor: Filtering



- Filter → Run
 - Filter cards by:
 - All cards
 - Active cards based on current defines
 - Active cards for a specific run
- Ribbon → Filter
 - Right-click on a card
 - Enables to filter cards by:
 - Search string
 - Body
 - Region
 - Material
 - Transformation
 - Detector
 - Cards with Errors

Input Editor: expressions

- flair expressions offer the possibility to make parametric runs
- a field value starting with = will force flair to evaluate its content as a function e.g.
BEAMPOS *x*: =2*10+length
- Expressions will be saved in the .flair file as well as comment inside the input
!*@what.1 =2*10+length*
- The input CARD will have the evaluation of the expression!
Don't manually change it → it will be overridden by flair the next time

Useful expressions:

- Units predefined: e.g.: *MeV, mm, ms...*
- Constants e.g: *fwhm, c, qe, ...*
- Particle masses: *Mp, Me, m[FLUKANAME]* like *mMUON m4HELLIUM*
- Card reference functions
what(n)
body(name,what)
card(tag,sdum/id,what)
- All common mathematical functions
- Some physics functions

Running – spawning of jobs

Create Loop of runs:
varying a variable
over a range

The screenshot shows the 'Run' window in the flair advanced interface. The main window is titled '+ /home/bnv/prg/physics/flair/examples/ntof/Target3RG.flair - flair'. The 'Run' panel is active, displaying a list of runs with columns for 'Run', 'Spawn', and 'Override'. The 'Override' panel is also visible, showing a list of defines with checkboxes and their values. The 'Progress' panel at the bottom shows the status of the current run: 'Status: Finished OK', 'Input: 20Fe2Bo/20Fe2Bo', 'Dir:', 'Started:', 'ETA:', 'Time/prim:', 'Elapsed:', 'Cycle:', 'Run:', 'Cycles:', 'Primaries:'. The status bar at the bottom indicates 'Fluka: Target3RG.flair Running 0 out of 6908'.

Select job queue

default: =nohup

local: attached to flair

nohup: detached from flair

tsp: task spooler (simple task scheduler on a single machine)

Custom queues can be created in flair.ini

[Batch]

queue = /path/command

Override some run parameters

Activate / Deactivate defines or modify its default value

Jobs can be spawned in multiple CPUs
→ enter the number of required CPUs

Spawned name fully configurable:
Configuration → Run → SpawnRunName

e.g. $V_{i2} = \text{input}_{01} \text{input}_{02}$
 $i2/N = 01/\text{input}, 02/\text{input}$

Data processing and custom rules

The screenshot displays the Flair software interface. The main window shows a 'Run' window with a list of runs and a 'Detectors' window showing a table of detector data. A 'File Selection Rules' dialog box is open, showing a list of rules and a syntax section. A red arrow points from the 'Filter' button in the 'Run' window to the 'Rules' dialog box.

Run	Spawn	Detectors
<ntof33>	2	
1/ntof33	8	1/ntof33
2/ntof33	8	1/ntof33

Run	Type	Output
1/ntof33	usrbin	1/ntof33_50.bnn
1/ntof33	resnucle	1/ntof33_51.mc
1/ntof33	usrbdx	1/ntof33_52.bnx
1/ntof33	usrcoll	1/ntof33_53.cll
1/ntof33	usrtrack	1/ntof33_54.trk
1/ntof33	usrtrack	1/ntof33_55.trk
1/ntof33	usrbin	1/ntof33_56.bnn
1/ntof33	usrbin	1/ntof33_57.bnn
1/ntof33	usrbin	1/ntof33_58.bnn

Rules

- +\\d\\d\\d_fort_\\U
- +\\d\\d\\d_\\Ue

Syntax: [+][filename | ^regexpr\$]

Special Characters:

Character	Description	Default
\\I	Input name	\\U Unit number
\\T	Type (usrtrack...)	\\t Short type (t,x,...)
\\e	Default extension	
.	Any character	* 0 or more char
+	1 or more char	? 0 or 1 match of char
\\d	Digit	\\D Non Digit

Each run (also spawned ones) can be processed separately

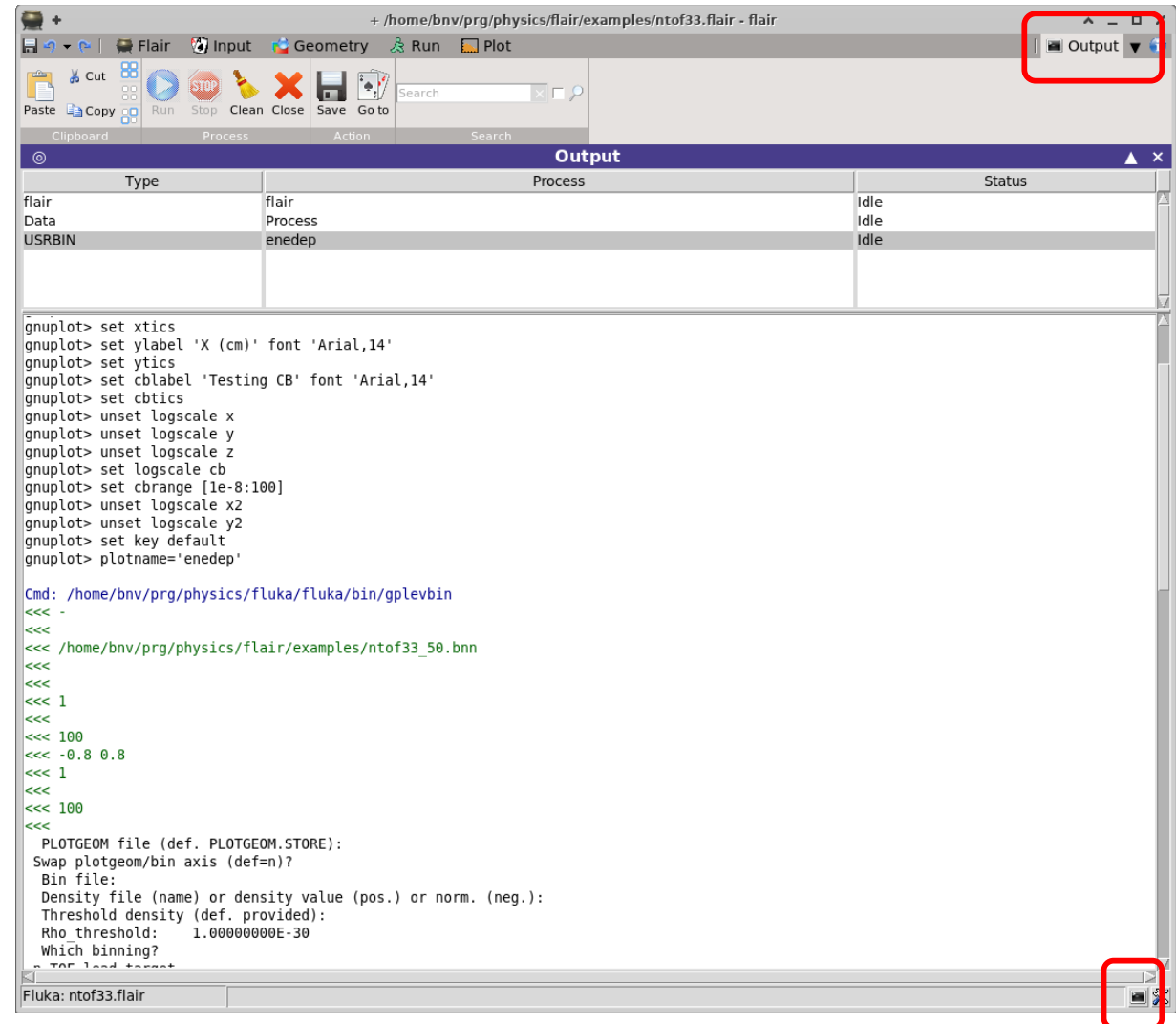
Custom processing commands can be created in flair.ini

[Data]
cmd.N = name : command
arg.N.M = arg : type : default : label
inp.N.L = type : default : label

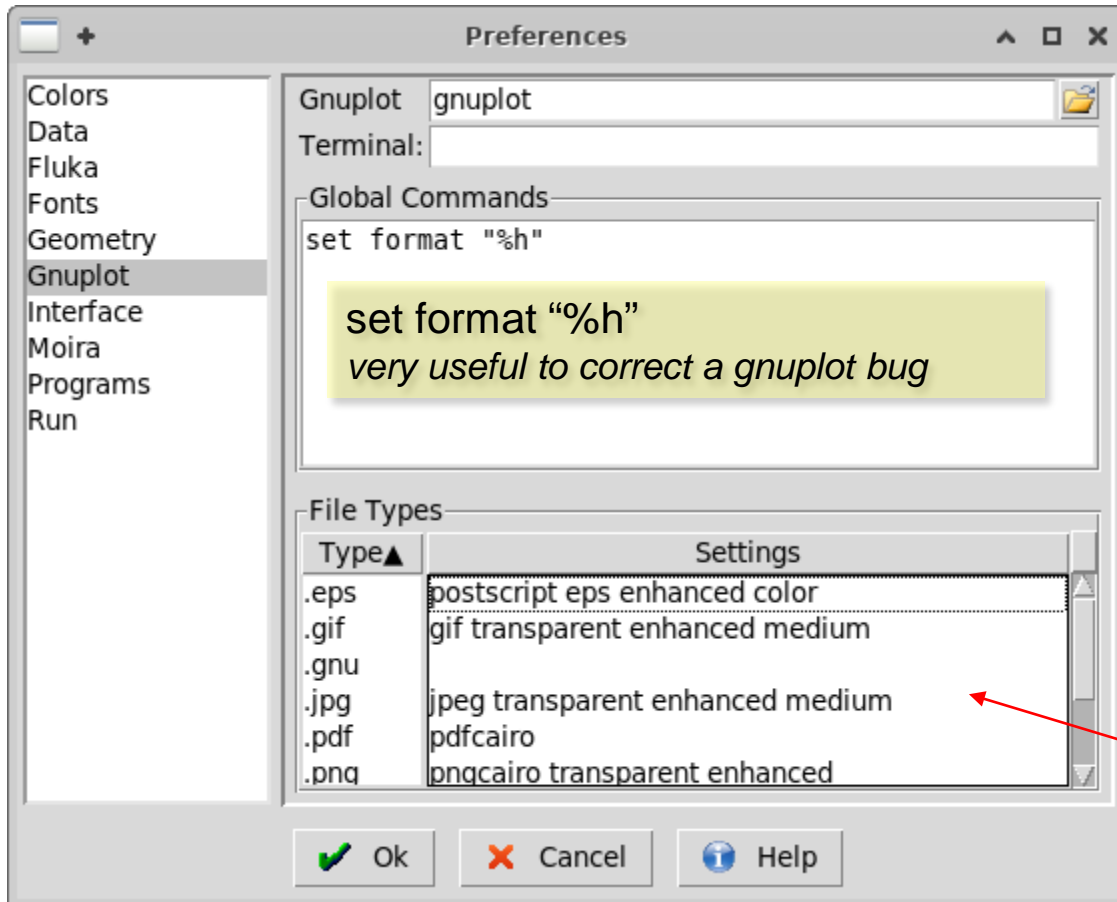
look the default flair.ini for examples

Output page

- useful source of information:
 - Every command executed by flair will be displayed there
 - Classified by the process that started the command
- Various commands:
 - Clean the display
 - Save the output as text
 - Goto highlight specific card having an error Either on Input or in Geometry Editor
 - Searching in the text
- **ERRORS/WARNINGS**: are clickable going to the faulty card



Plotting – configuration

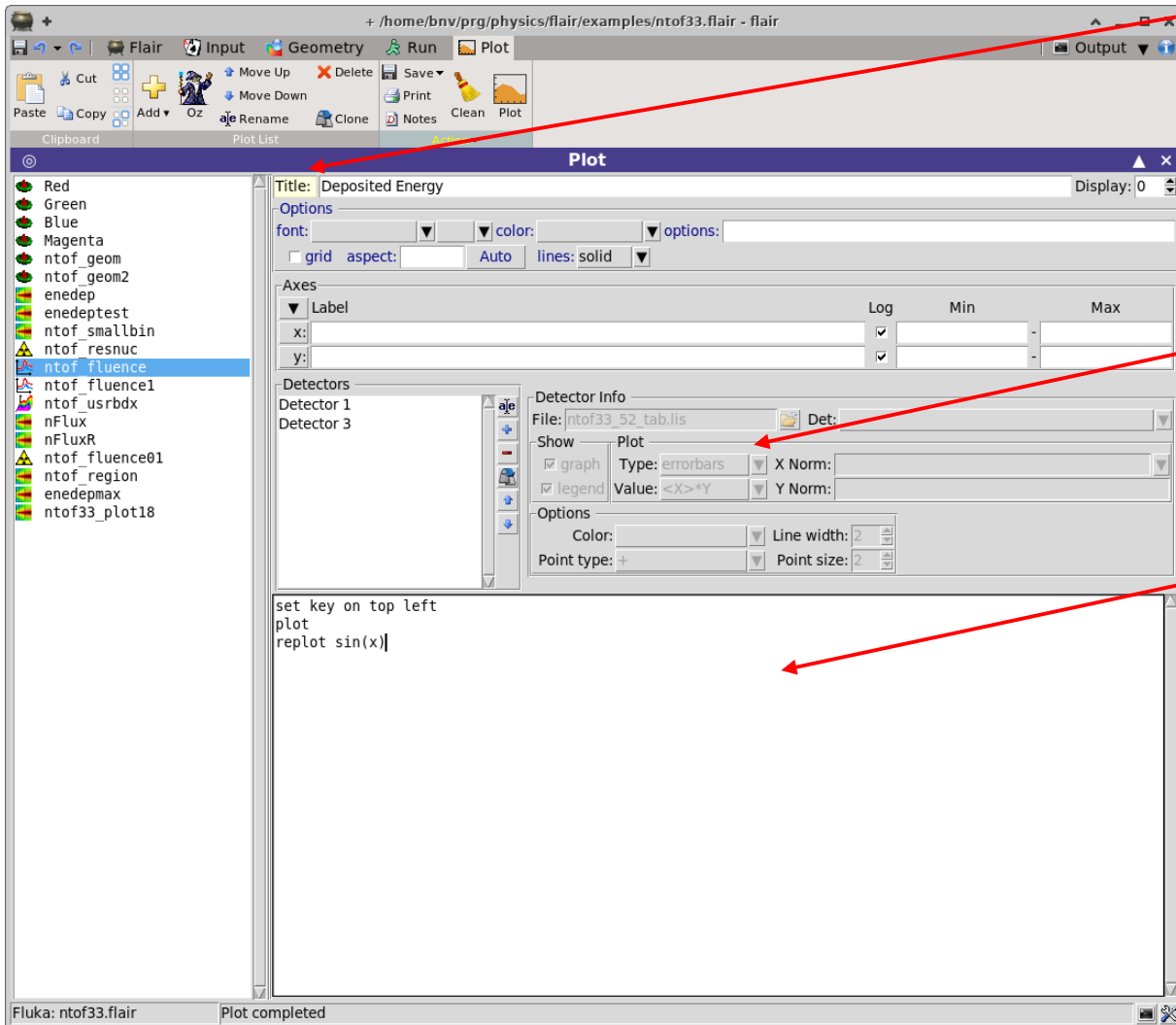


- Configure → Gnuplot

- select the gnuplot program path
- select the favorite gnuplot “terminal”
- Set some global commands for gnuplot to be executed on every plot
Similar to $\${HOME}/.gnuplot$
- Customize export file type, or even create new ones (right-click)

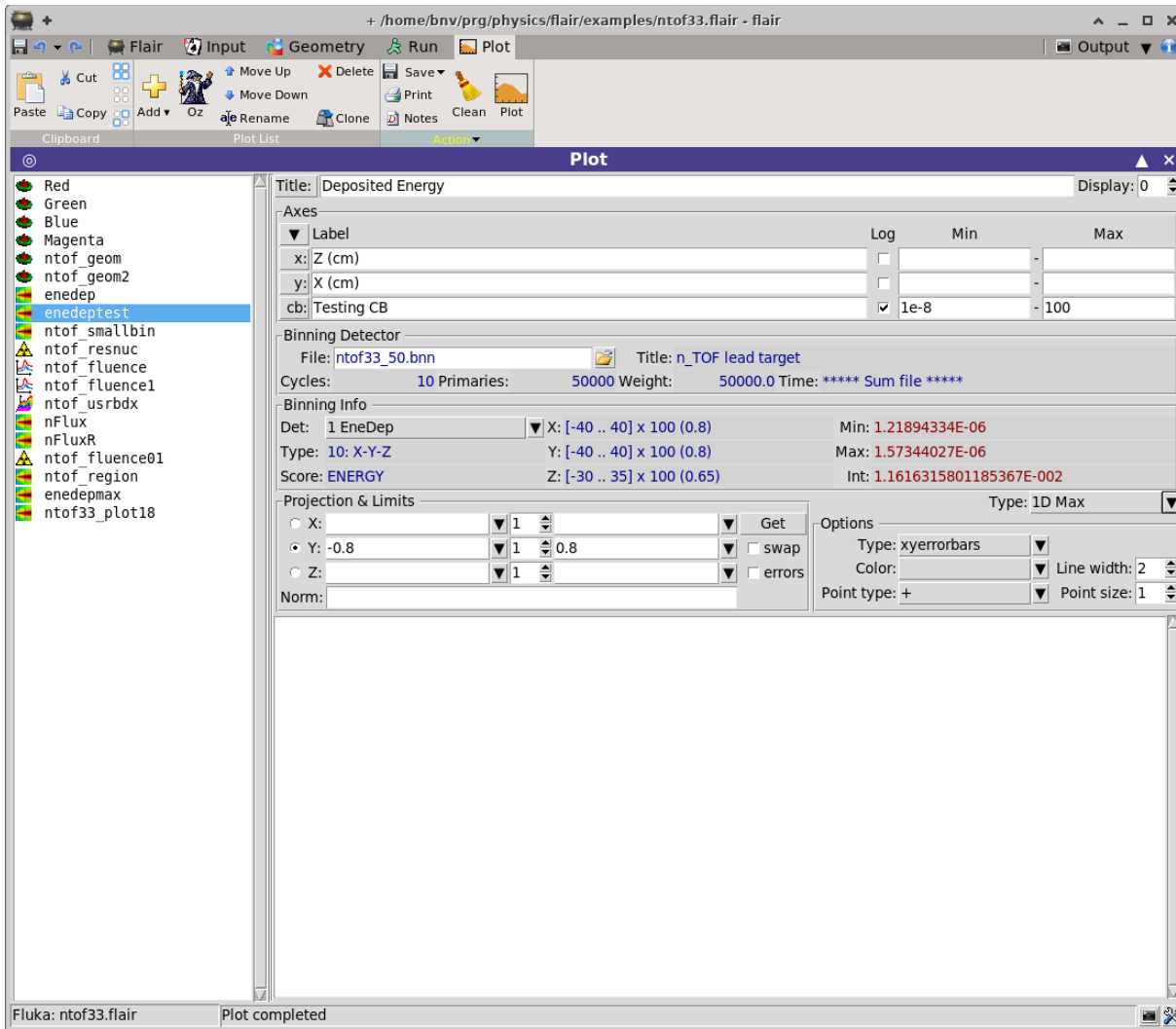
Right click to
add/delete/clone

Plotting: advanced features



- **Label Button:** expand more advanced features
- Selecting multiple detectors/plots common fields are disabled “greyed”
Right-click to enable/disable them
Edit a value and it will be updated in all detectors/plots
- Command entry allows manual commands to gnuplot
special commands:
 - plot: execute at this line the plot command
 - replot: append extra plot commands e.g. experimental data


Plotting: usrbins

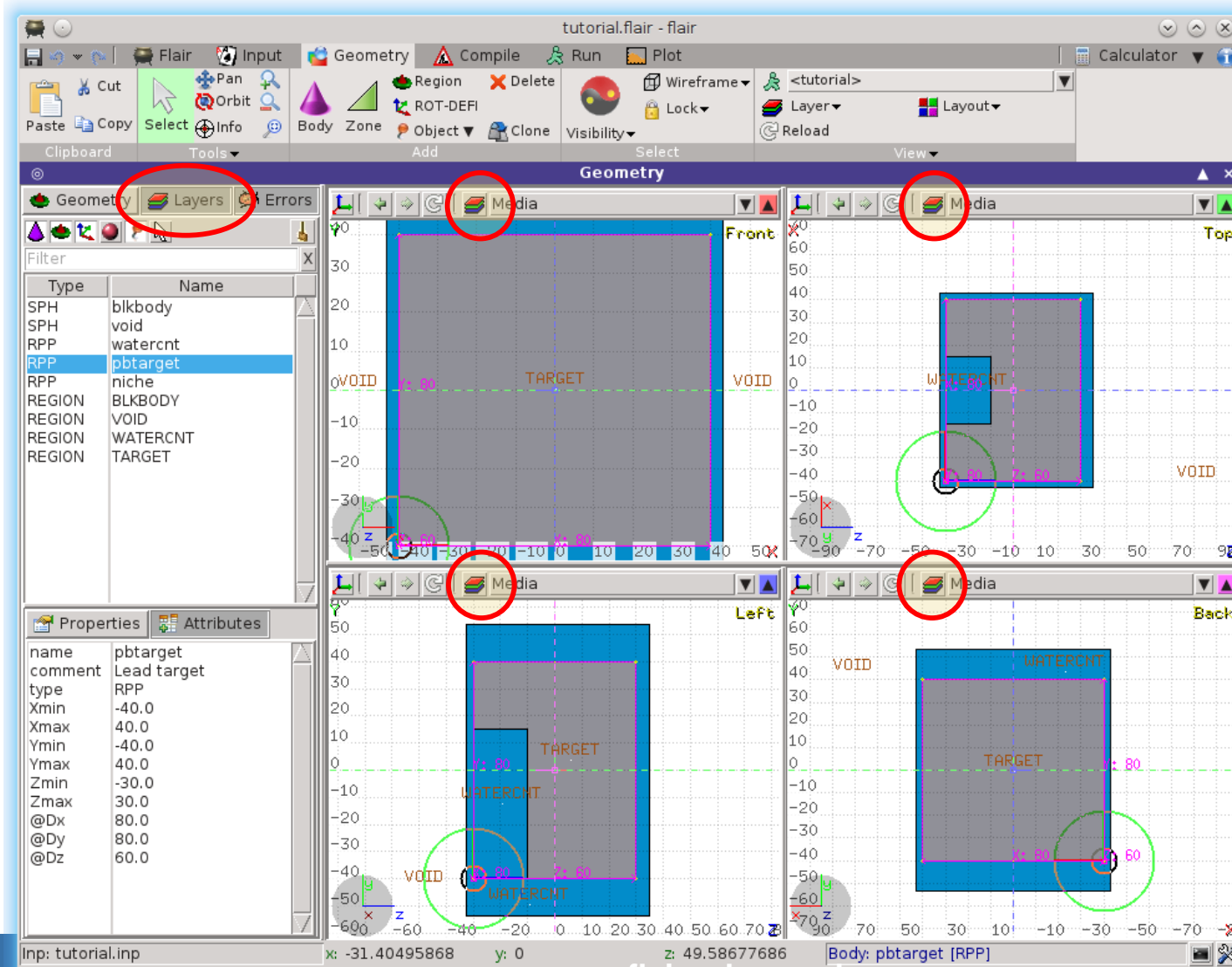


- Each plot generates a “*plotname.dat*” file that can be used in other plots e.g.
 - Prepare several usrbins with 1D projection or Max
 - Create a USR-1D plot adding the plots as detectors

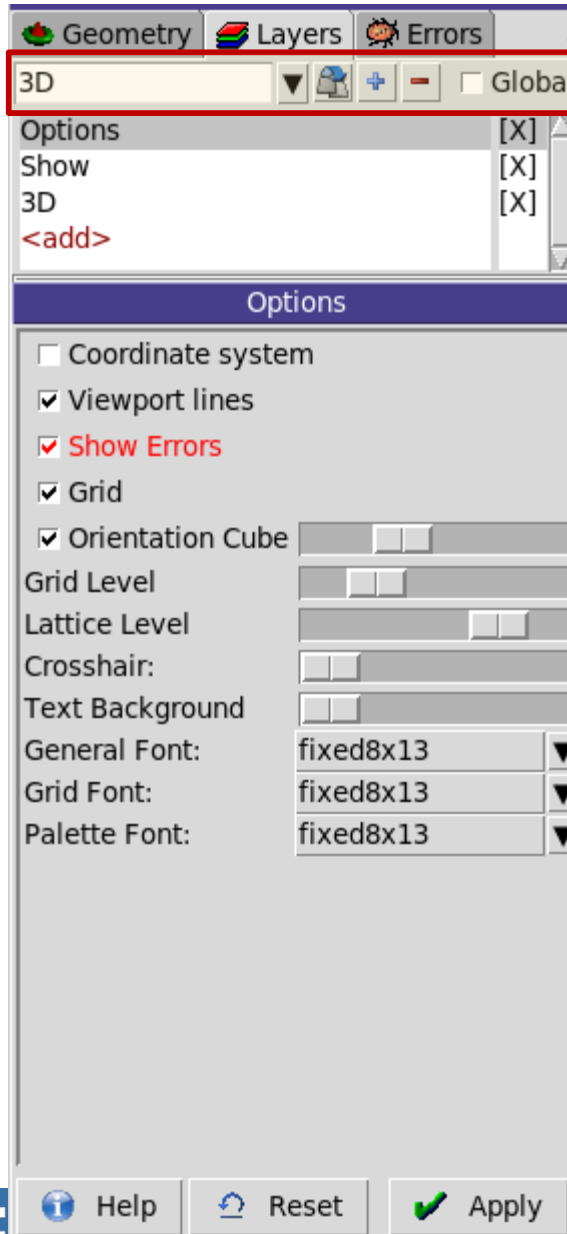
Geometry Editor

Geometry Layers [1]

Custom Layers can be specified in the “Configure Layer menu” ()



Geometry Layers [2]



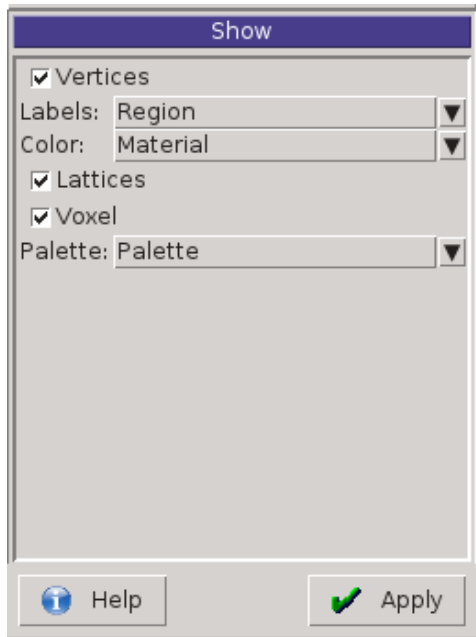
Toolbar:

- **Add/delete/rename/clone layers.**
- **Global:** when ticked the layer will be shown on every project, else it will be used only in the current project

Options:

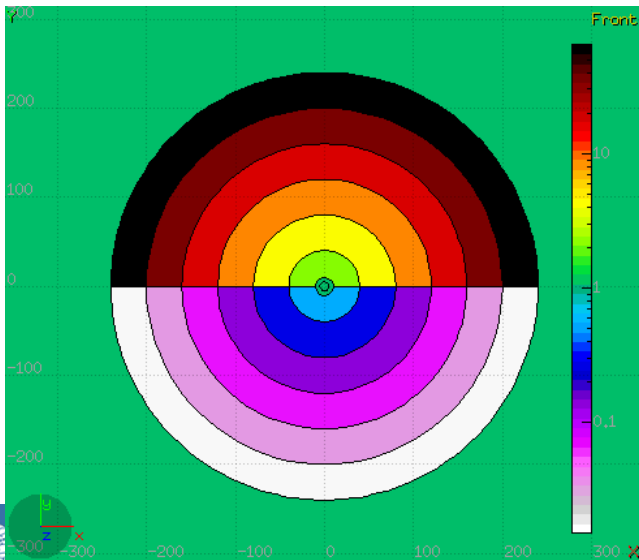
- **Enable/Disable:** Title, Coordinate system, Viewport lines, Vertexes and Grid;
- **Adjust:**
 - **Grid level** (set gridline intensity);
 - **Lattice level** (set lattice hash line intensity);
 - **Crosshair** (dimension of the crosshair in the center of the project);
 - **Orientation Cube** (size);
 - **Fonts**
 - ...
- All layers can be combined together e.g:
 - USRBIN and 3D
 - Custom color values (EMFCUT) with 3D
 - Image and USRBIN
 - ...

Geometry Layers [3]



Show: (2D drawing, and color filling options)

- **Bodies**: display the boundaries of bodies;
- **Vertices**: display the intersection of bodies;
- **Enable/Disable**: Lattice and Voxel;
- **Associate Region Colors to:**
 - Regions
 - Materials
 - Density
 - Importance Biasing
 - Splitting
 - Corrfactor
 - Deltaray
 - Thresholds
 - ...



Geometry Layers [4]

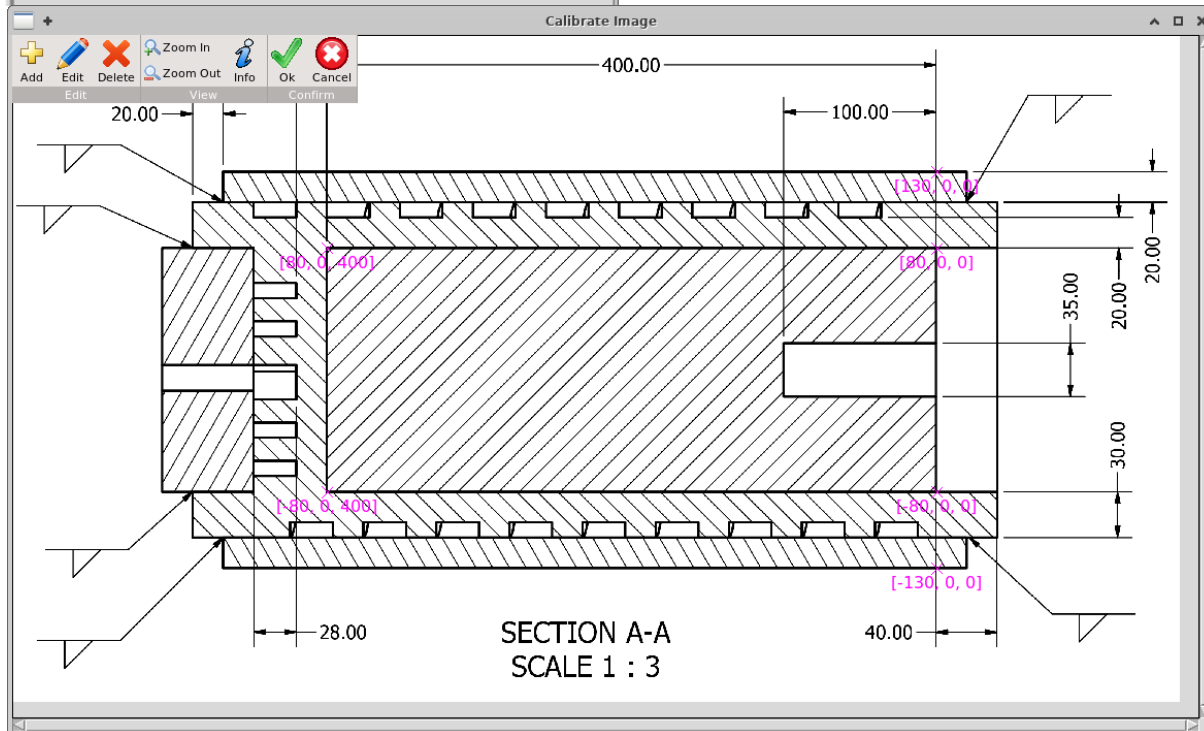
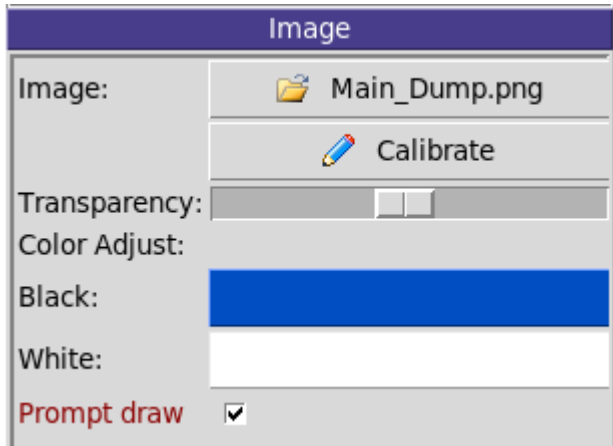
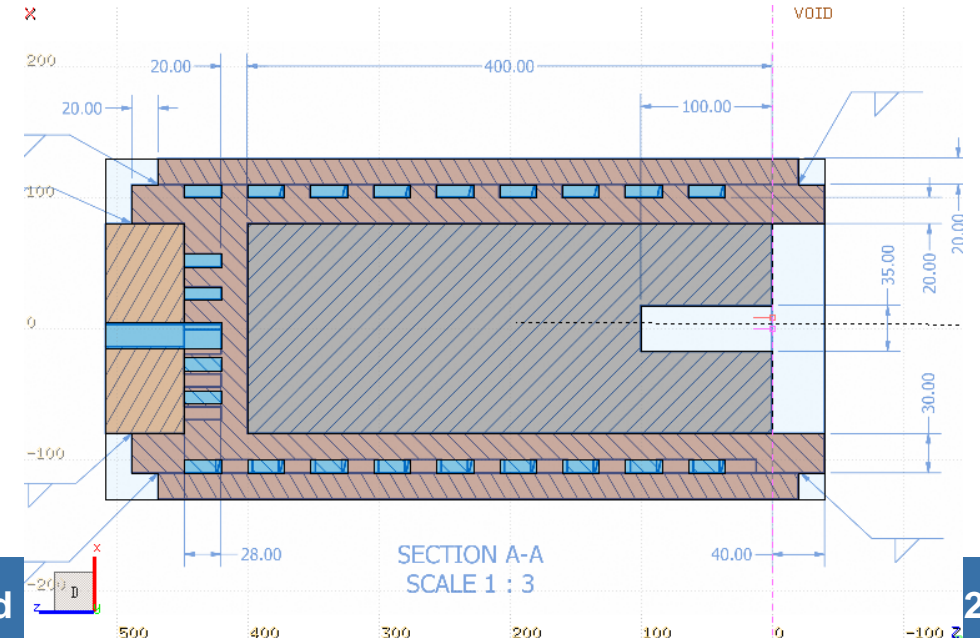
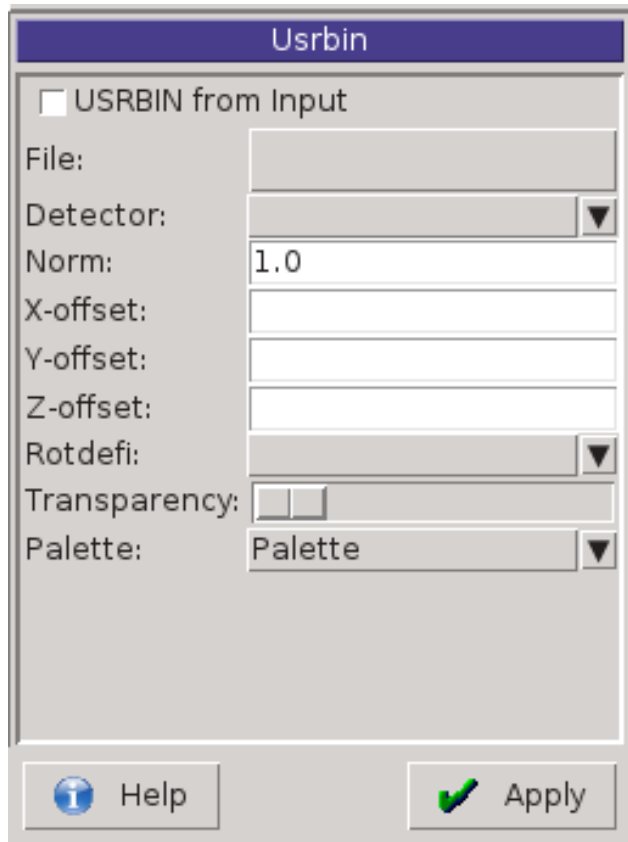


Image: set a background image to the geometry (i.e. a CAD-drawing);

- **Image**: load an image file (.png, .gif or .jpg);
- **Calibrate**: calibrate the image. Define a set of points (min. 3) on the image and specify their coordinate;
- **Alpha**: blending of the image
- **Color Adjust**: readjust the **black** and **white** colors of the loaded image.
- **Prompt draw**: immediate drawing of image (slower) or when display is idle. For editing is good to activate it.



Geometry Layers [5]



USRBIN:

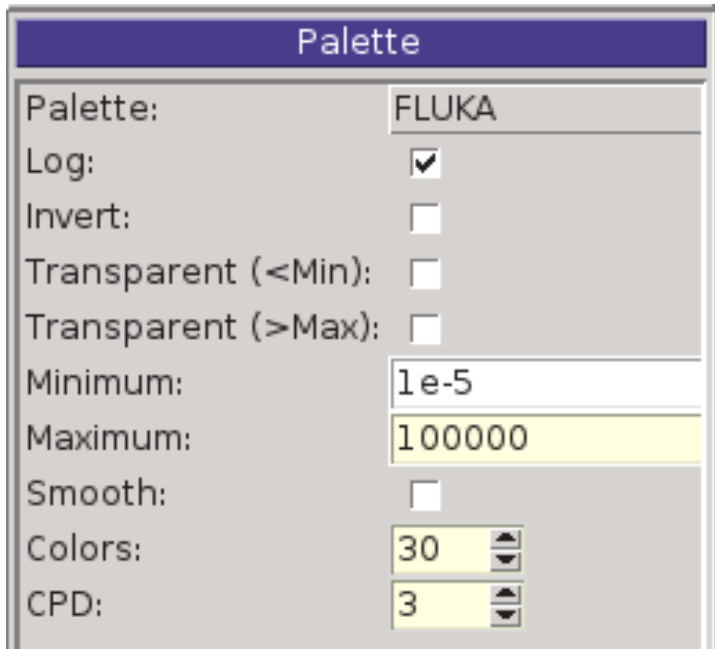
- USRBIN from input: To select a USRBIN card from input and displayed with a checker pattern
- Load **USRBIN file** (see SCORING lecture);
- Select a **detector** (or URSBIN) among the ones present in the file;
- **Normalization** constant;
- Associate a **ROT-DEFI** transformation;
- Alpha blending between USRBIN colors and materials colors



USRBIN should be combined with the Palette to define the color limits

- Up to 10 Usrbins can be displayed overlaid

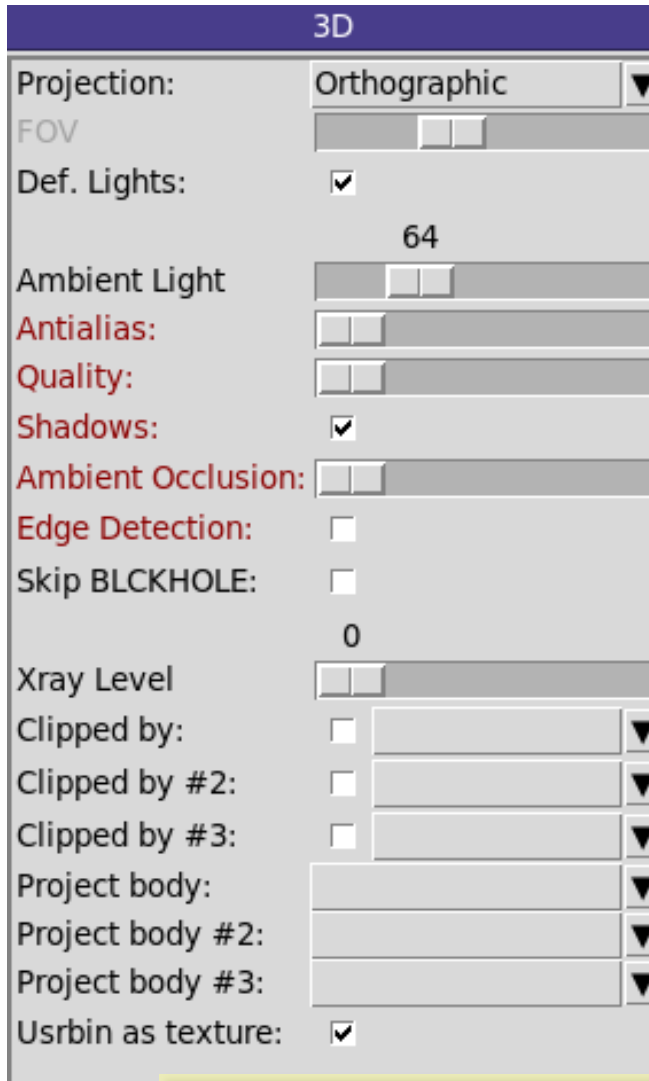
Geometry Layers [6]



Palette: enable/set color band properties

- Change the default color **Palette**;
- **Enable/Disable Log** scale;
- **Set: Maximum, Minimum** and color **steps**.

Geometry Layers [7]



Red labels: means time consuming

3D: enable 3D rendering

- Projection: Orthographic/Perspective/Combination;
- OV: Set camera **aperture** angle;
- Def. Lights: enable/disable use of default lights
- Ambient Light: Intensity of ambient light;
- **Antialias**: adaptive super sampling (slow rendering);
- **Quality**: improve quality by using more an more advanced illumination features
- **Shadows**: Enable/Disable casting of shadows
- **Ambient Occlusion**: advanced shading technique
- **Edge detection**: draw black edges on region boundaries
- Xray – automatic transparencies;
- Clipped by: setting a clipping body [tick] for negative body
- Project body: body used to project possible usrbin information
- Usrbin as texture: surface map usrbin coloring on visible regions

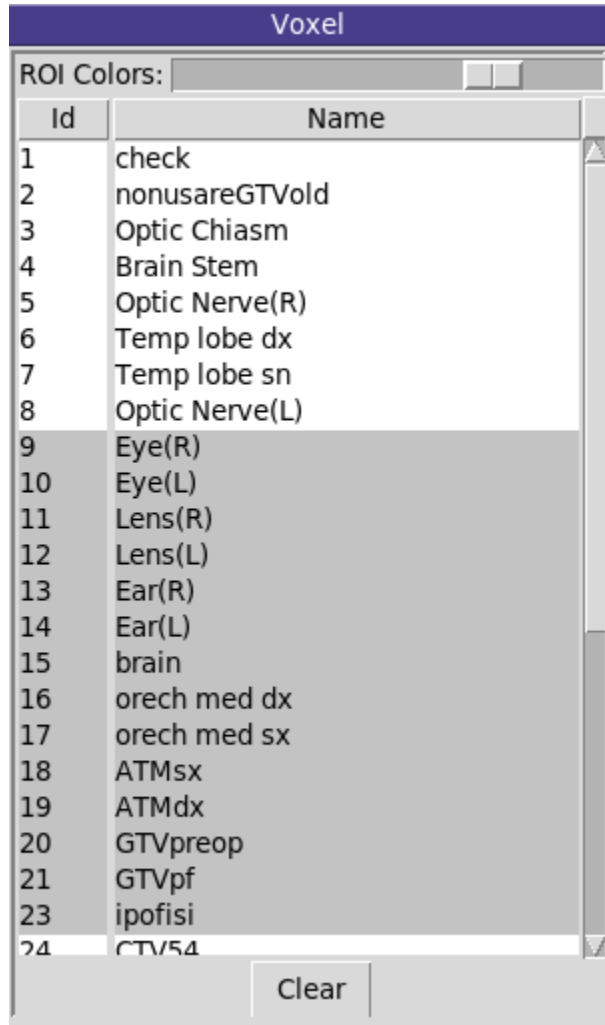
Geometry Layers [8]

Userdump				
Userdump file:		cylinder001_dump		
Start:		0		
# Events:		100		
	Particle	Emin	Emax	Color
[]	4-HELIUM			#FF8C00
[]	3-HELIUM			#FF1493
[]	TRITON			#B03060
[]	DEUTERON			#9932CC
[]	HEAVYION			#0000FF
[]	OPTIPHOT			
[]	RAY			
[X]	PROTON			#FF0000
[]	APROTON			
[X]	ELECTRON			#00FF00
[X]	POSITRON			#00FF00
[]	NEUTRIE			#708090
[]	ANEUTRIE			#708090
[]	PHOTON			#FFB90F
[]	NEUTRON			
[]	ANEUTRON			
[]	MUON+			#00FFFF
[]	MUON-			#00FFFF
[]	KAONLONG			
[]	PION+			#8B4513

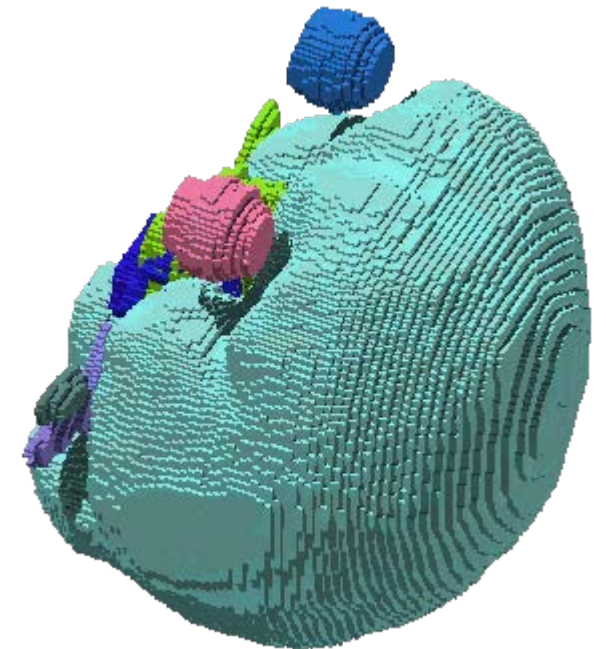
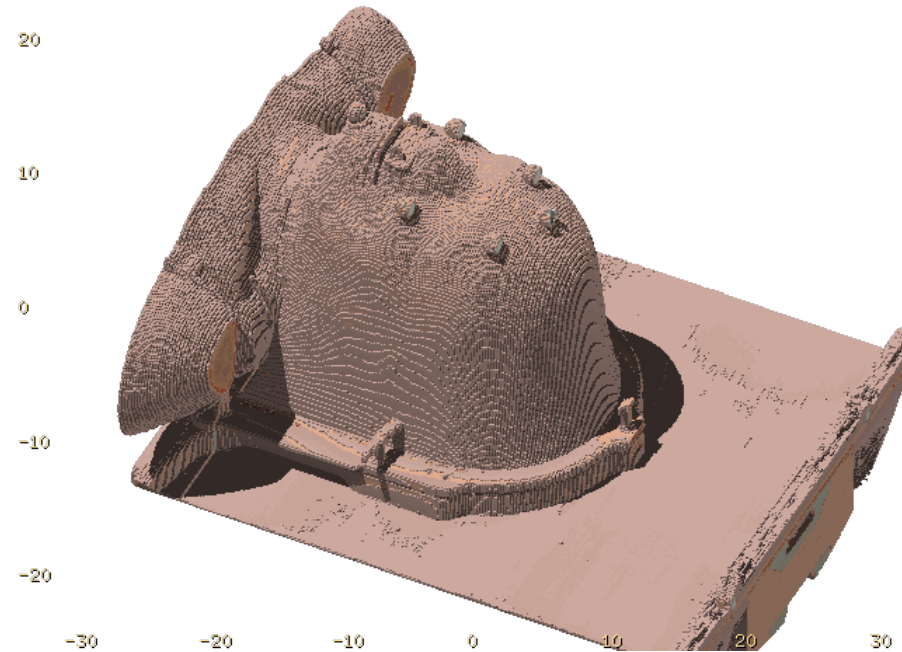
Userdump: plot particle tracks from USERDUMP

- Userdump file: file to be loaded
- Start: starting primary event
- # Events: total number of events to plot
- Select the particles you want to display
- Define energy limits if needed
- Colors of the track

Geometry Layers [9]

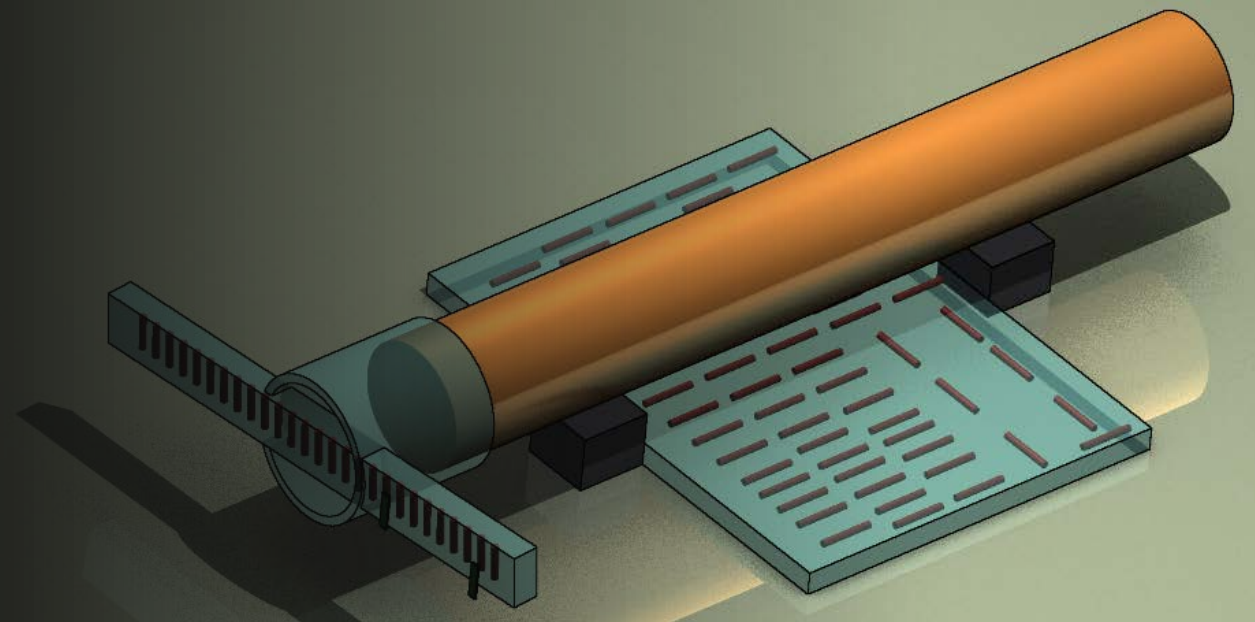


Voxel: Filter Region of Interests (ROI) in voxel
if the voxel has embedded ROIs then select to filter the display with the needed ROIs



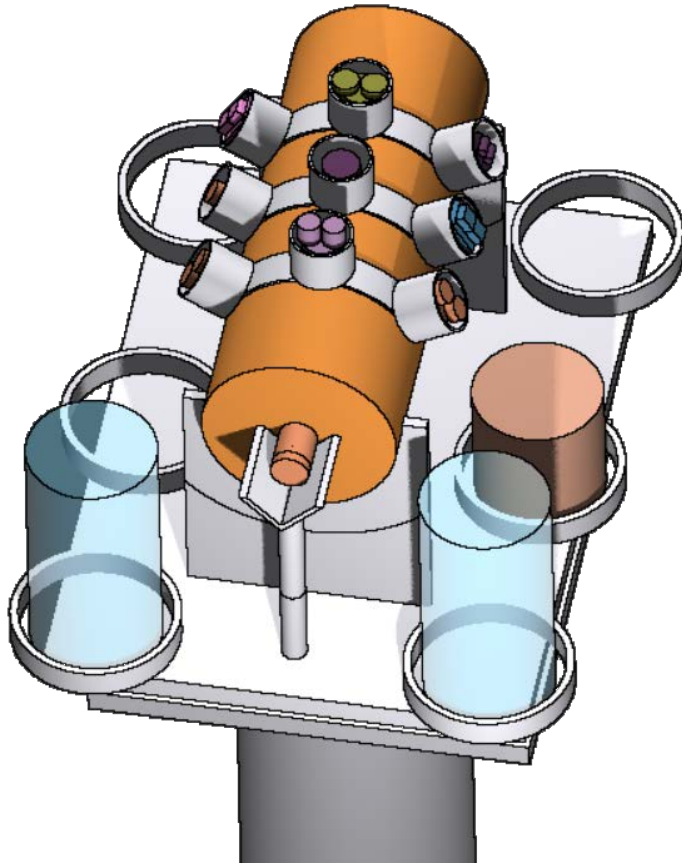
FARM_{3D} – Flair advanced render module

C. Theis, V. Vlachoudis

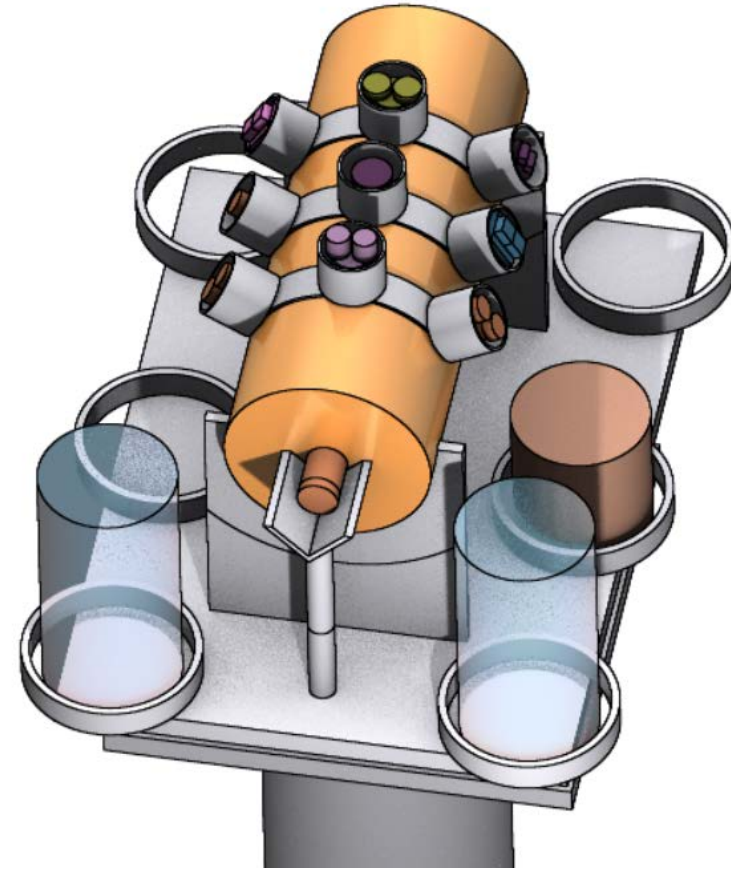


FARM

- Two render modes available
 - Speed – optimized for 3D previews during geometry construction
 - Quality – optimized for high-quality photorealistic renderings for presentations & publications



Speed – interactive rendering



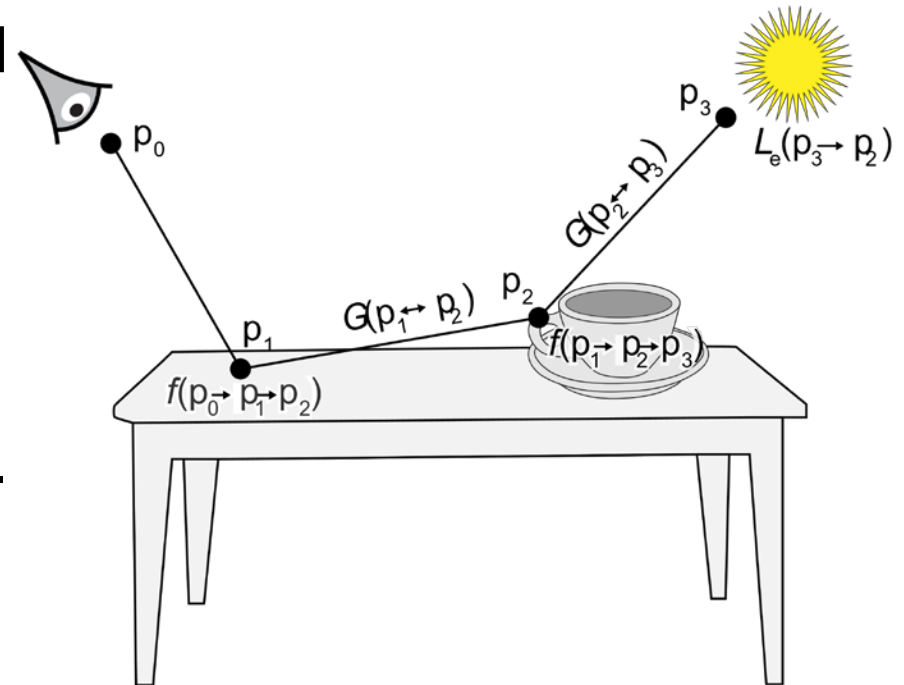
Quality – 15 s to render

FARM

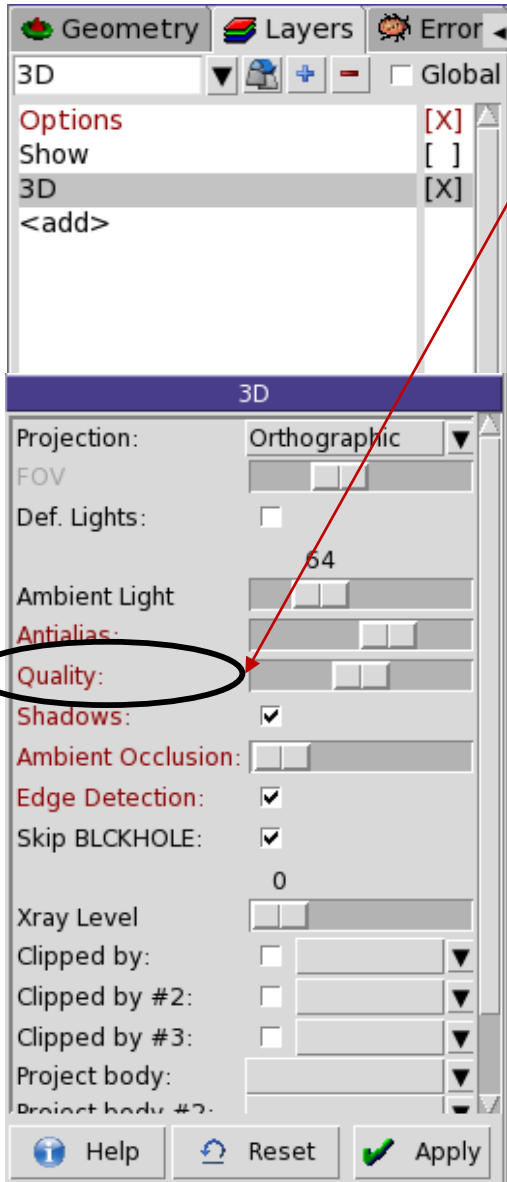
FARM (Flair advanced rendering module) is a physically based ray-tracer which solves the light transport equation. This is done by integrating so called “**bi-direction scattering functions**”, which model the interaction of photons with material, using Monte Carlo or analytical methods.

In the “**Speed**” mode it falls back to a relative simplified unphysical lighting model.

When activating the “**Quality**” mode numerous different light interaction models can be selected which are partially based on measured scattering data.



Render - mode



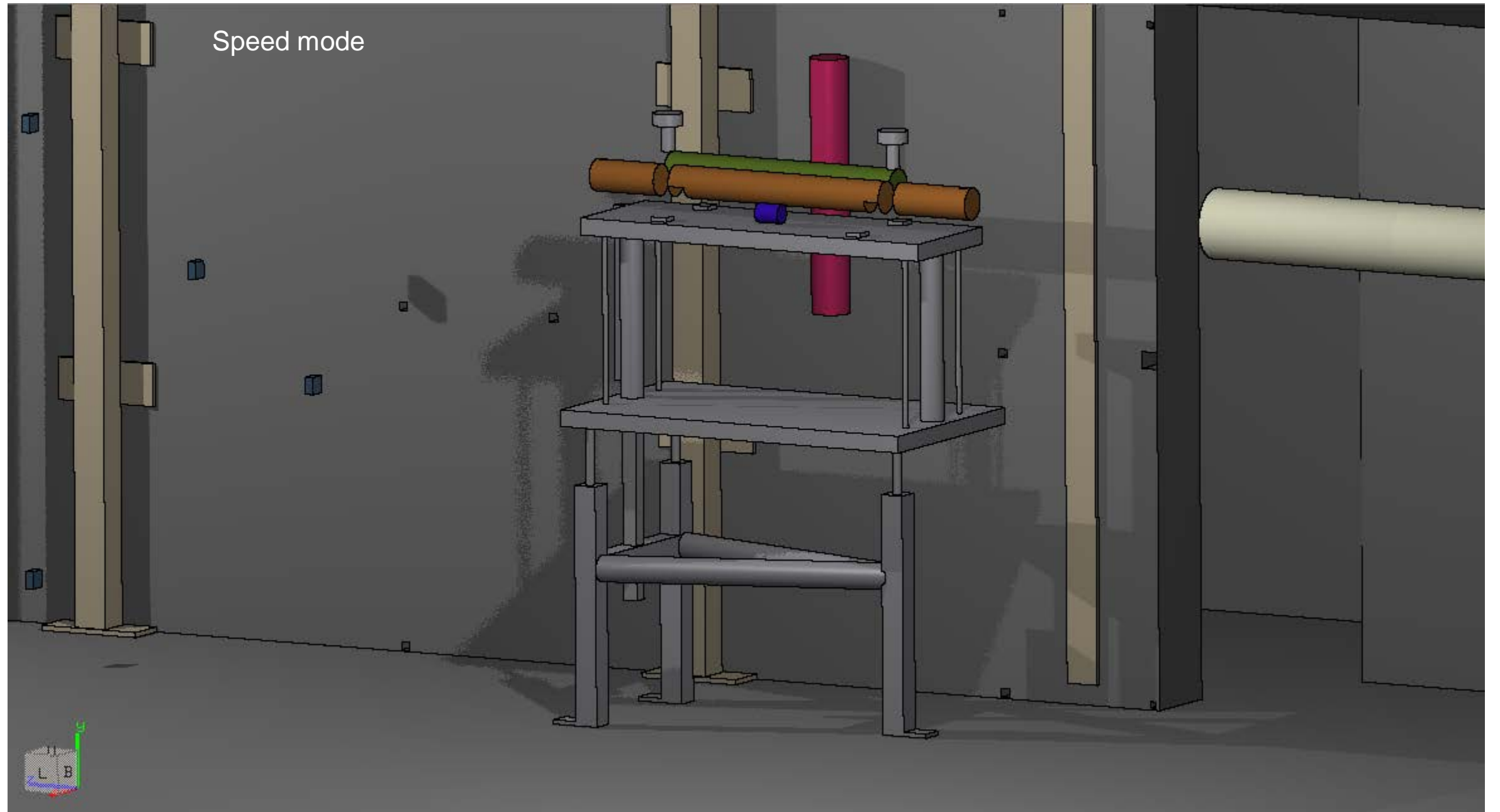
- By default “Speed” mode is activated.
- “Quality” mode can be activated by moving the slider in the 3D layer options to the right

Once “Quality” is enabled the following effects will be taken into account:

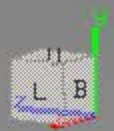
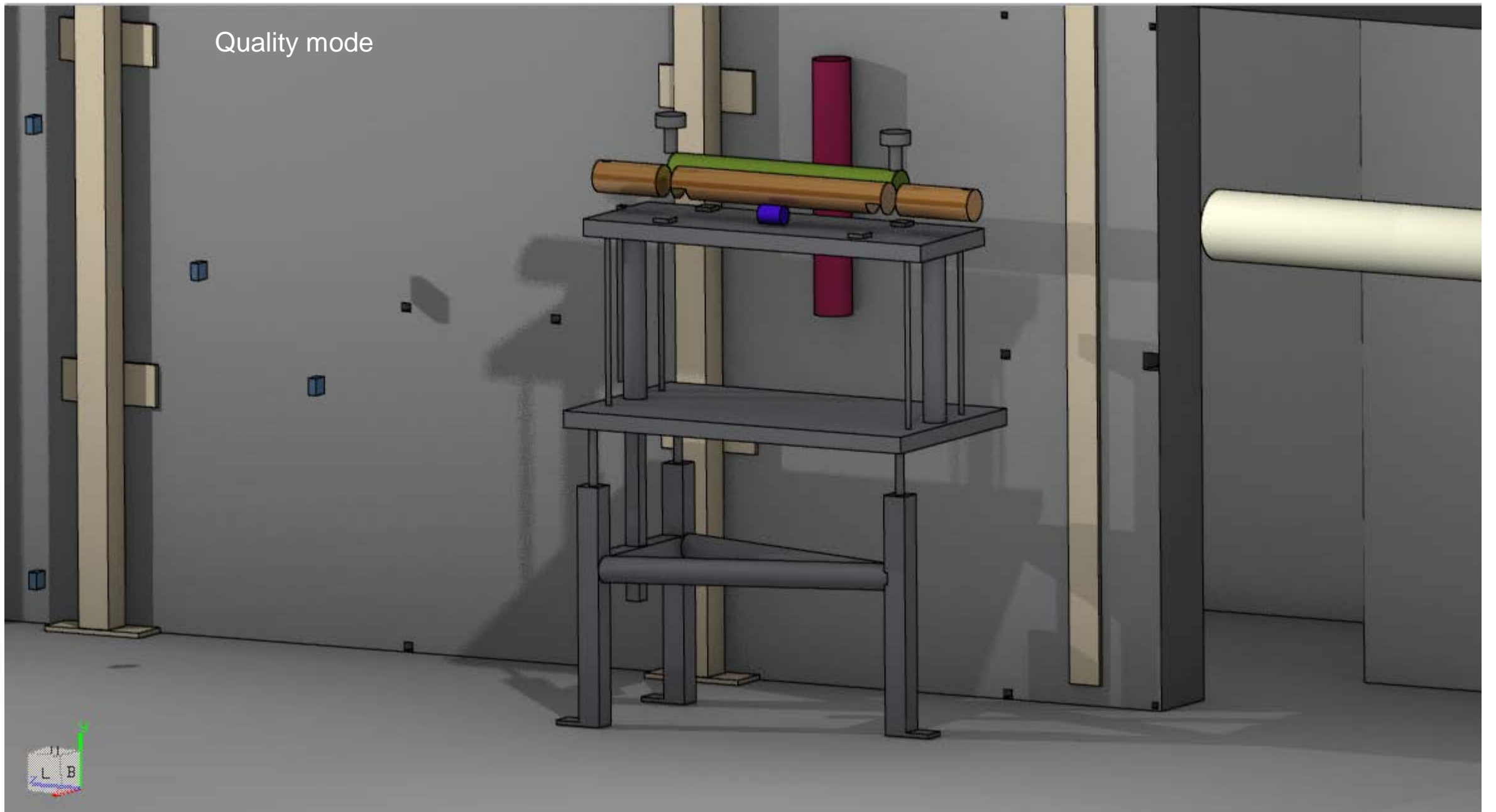
- Material dependent scattering & reflection models
- Reflection & refraction of light
- Fresnel reflection for conductors & di-electrics
- Dispersion
- Beer’s absorption
- Microfacet distribution for polished metals

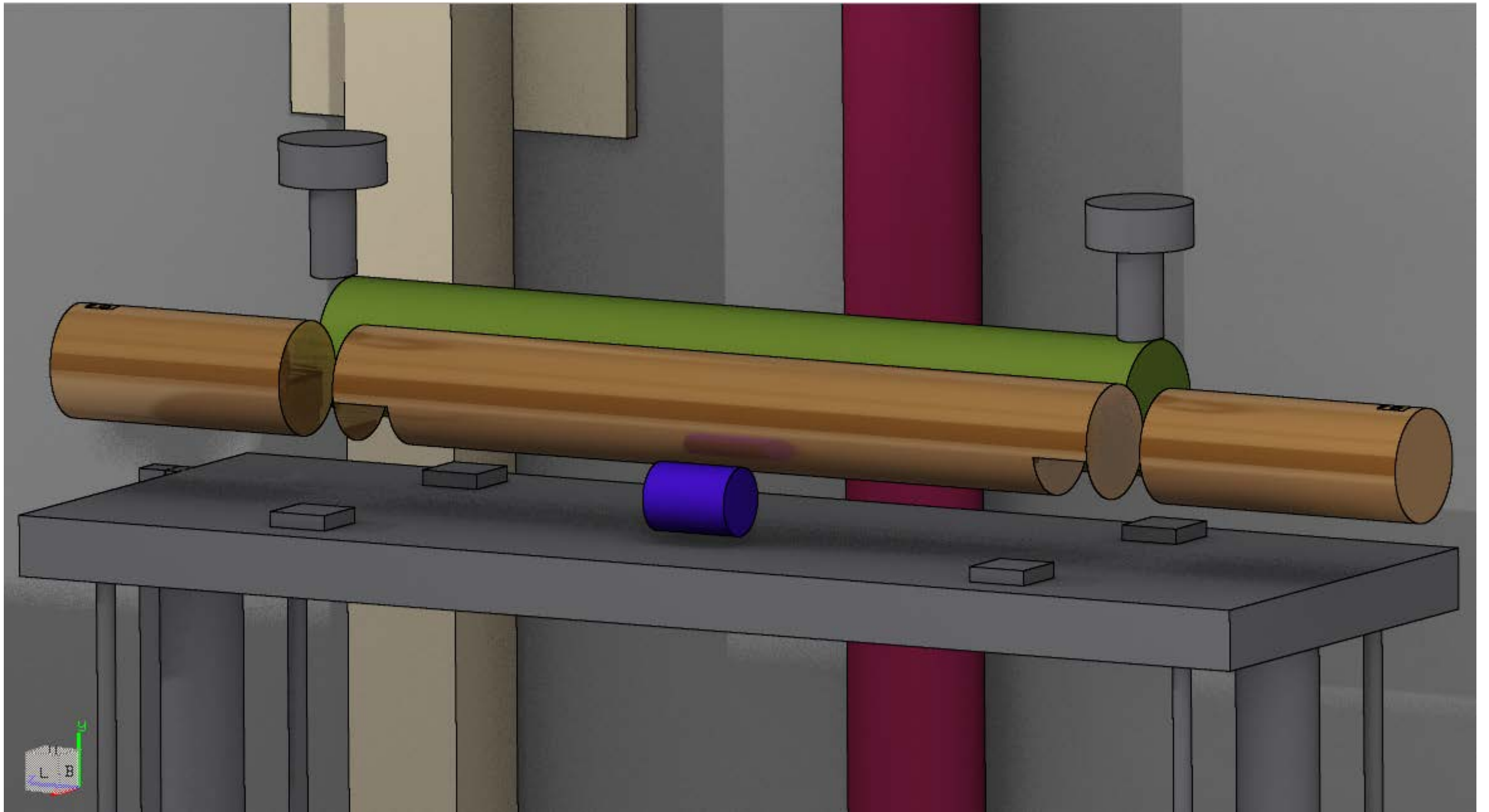
Please note that “Quality” mode is clearly slower as it invokes a physically based renderer requiring more resources to account for all these effects correctly.

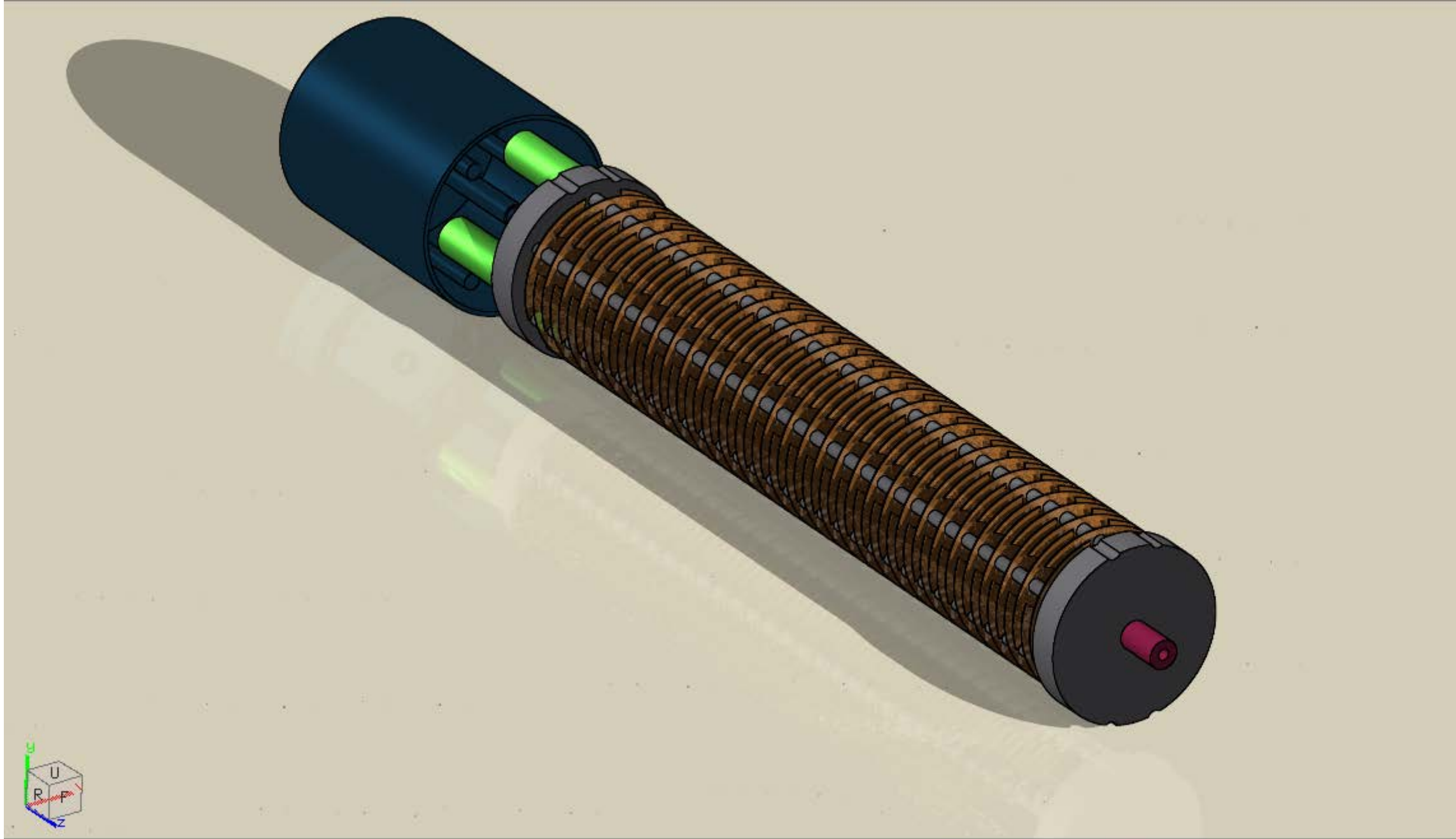
Speed mode

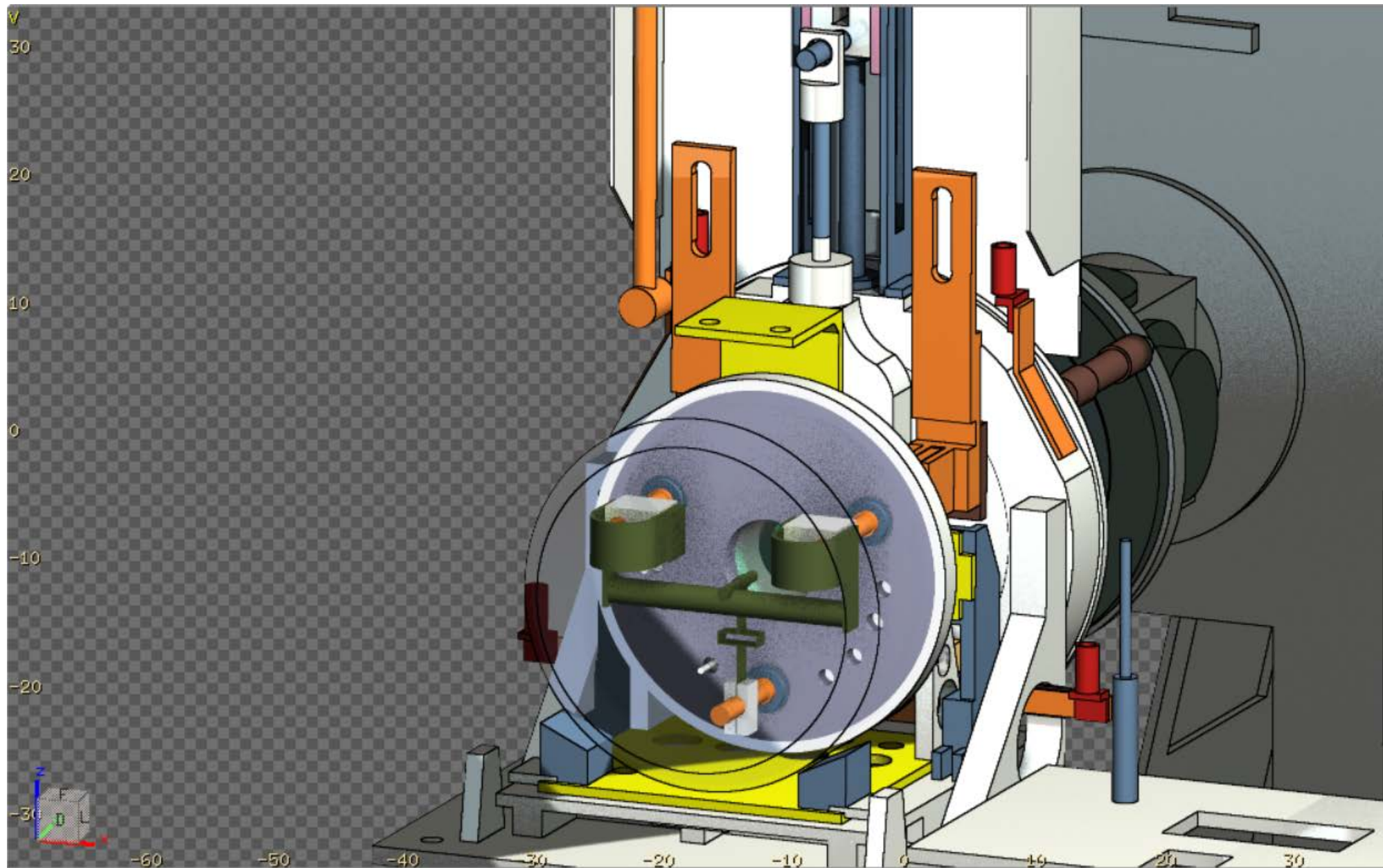


Quality mode

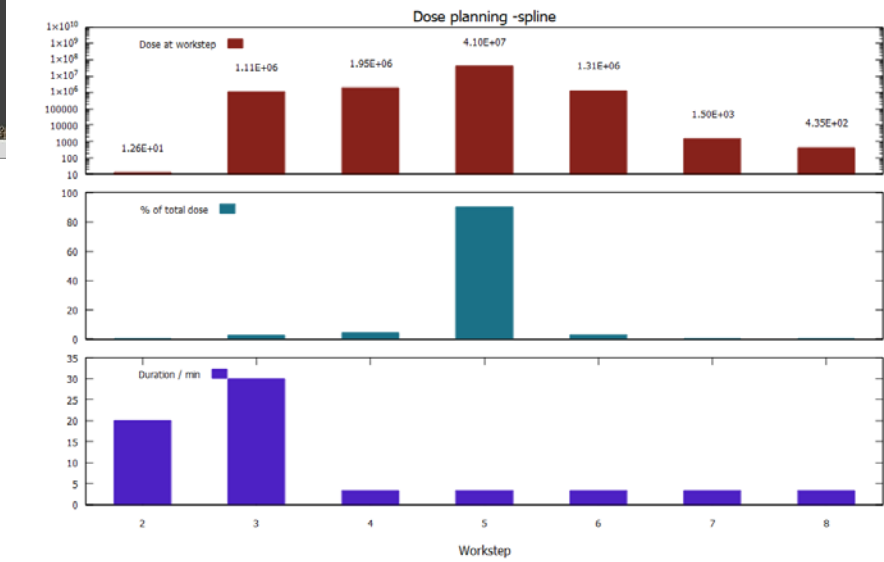
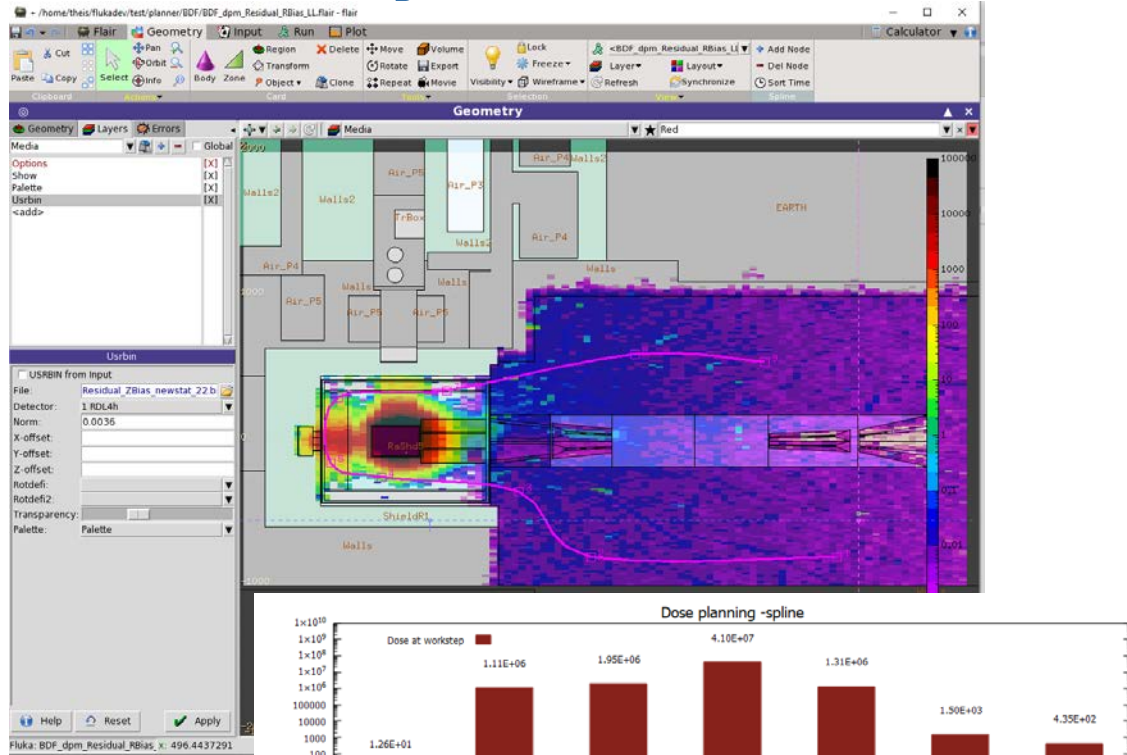








Geometry Editor: Planner



Integral dose: 4.5397e+07
Total dose while moving: 4.5397e+07 (1.0000e+02%)
Total dose while working: 1.2625e+01 (2.7811e-05%)

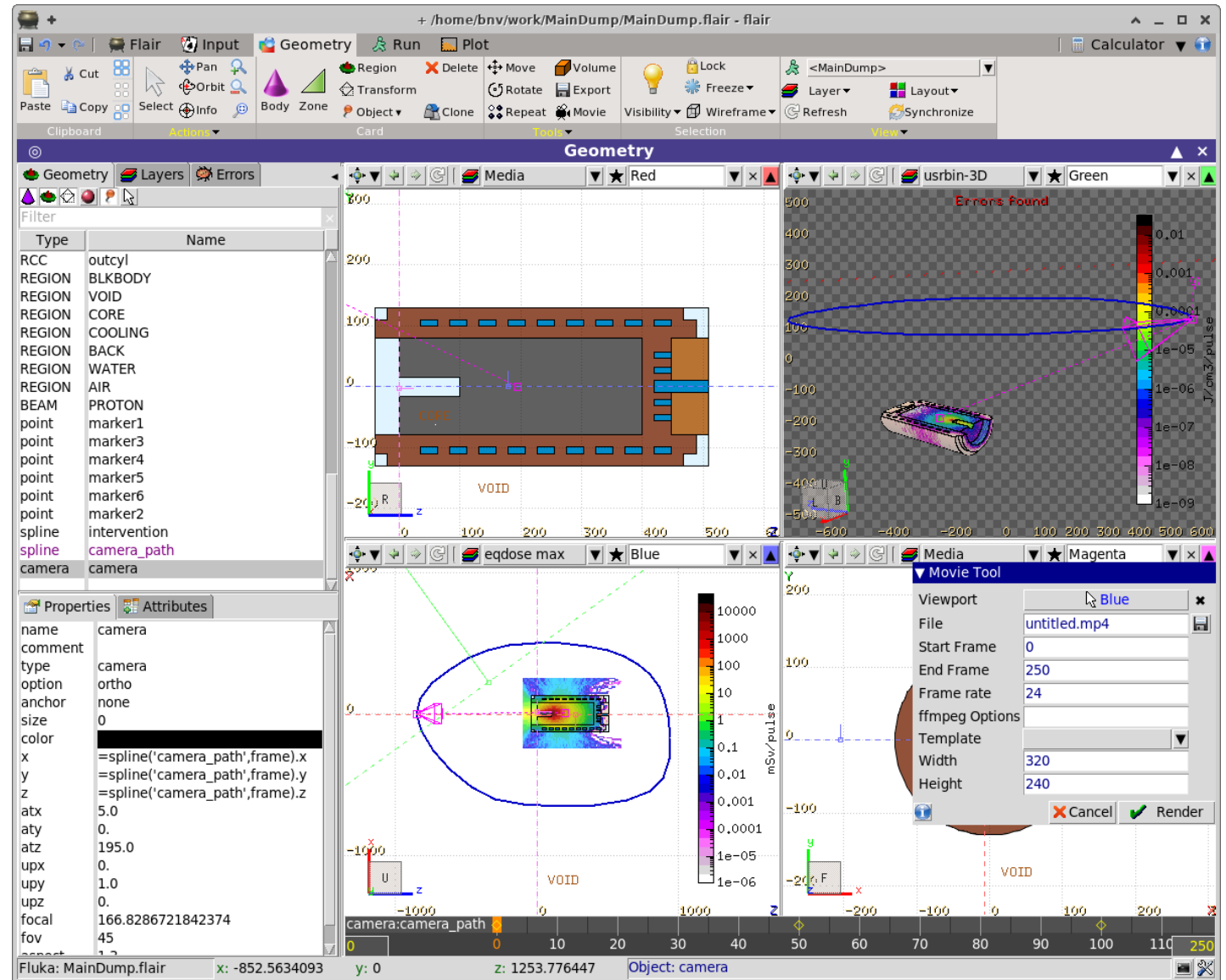
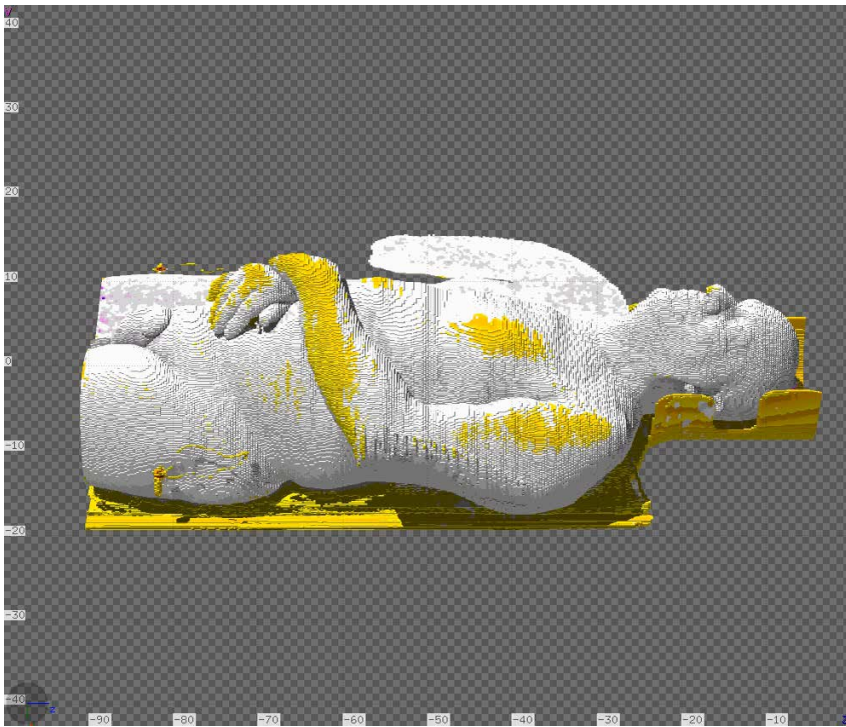
Allows for calculating the dose incurred due to interventions in an activated area, taking into account not only working a specific places but also the traversal of activated zones in order to reach the actual workplaces.

For this purpose workflows can be designed which consist of:

- trajectories
- work places
- times for traversal as well as activities

Geometry Editor: movie creation

- flair can create movies with the use of following objects:
 - A linked spline and camera
 - a viewport drawing layers

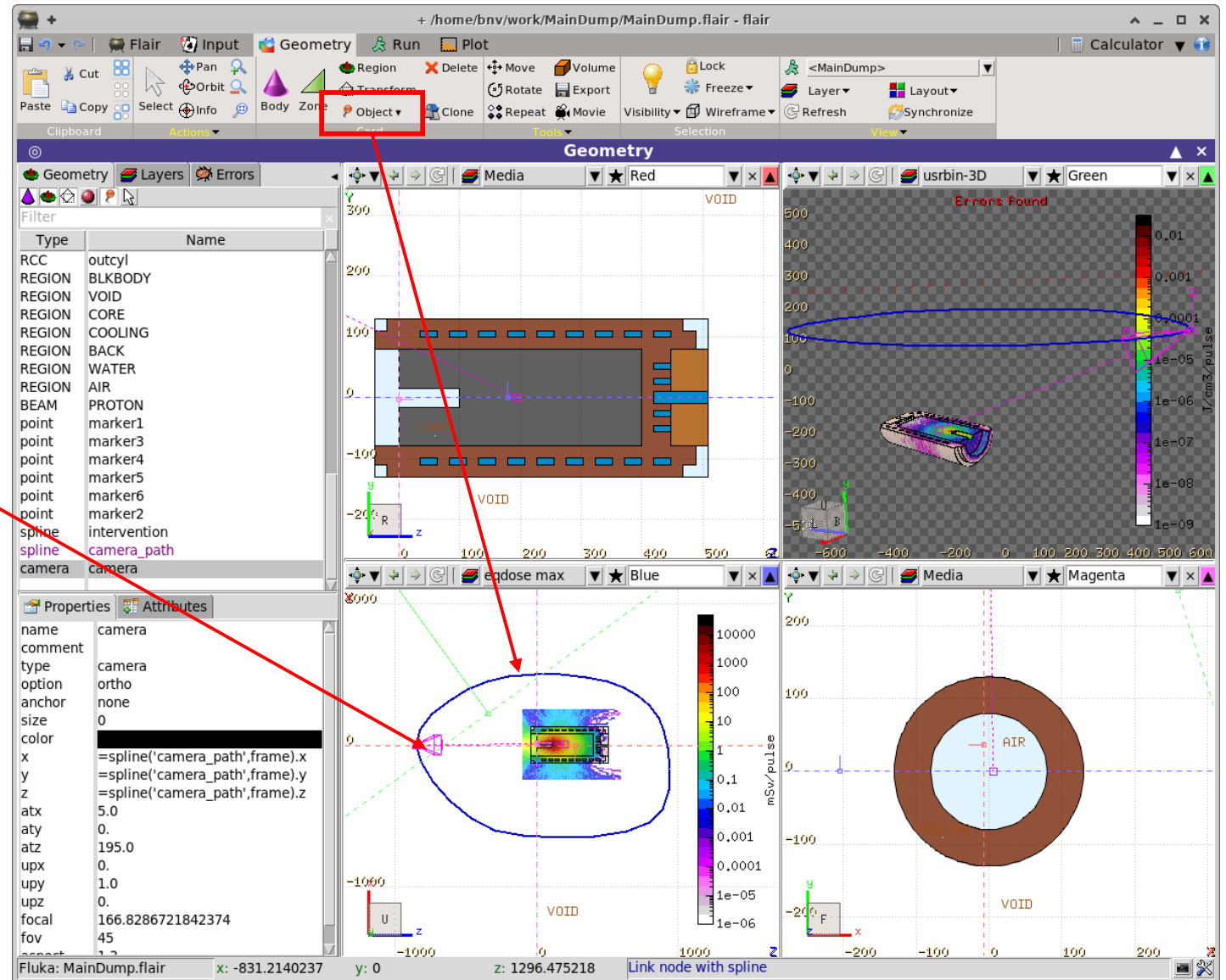
A screenshot of the flair Geometry Editor software interface. The window title is "+ /home/bnv/work/MainDump/MainDump.flair - flair". The interface is divided into several panels:

- Top Panel:** Contains various tool icons for editing geometry, such as Cut, Copy, Paste, Select, Move, Rotate, and Delete.
- Geometry Panel:** A list of objects in the scene, including RCC (outcyl), REGION (BLKBODY, VOID, CORE, COOLING, BACK, WATER, AIR), BEAM (PROTON), point (marker1-6), spline (intervention), and camera (camera).
- Properties Panel:** Shows the properties of the selected camera object, including name, type (ortho), anchor, size, color, and position (x, y, z, atx, aty, atz, upx, upy, upz). The focal length is 166.8286721842374 and the field of view is 45.
- Viewport:** A 3D view of the geometry, showing a complex structure with a color scale on the right ranging from 1e-09 to 0.01. The axes are labeled x, y, and z.
- Movie Tool Panel:** A panel for creating movies, with fields for Viewport (Blue), File (untitled.mp4), Start Frame (0), End Frame (250), Frame rate (24), and Template. It also has buttons for Cancel and Render.

Geometry Editor: movie creation

Steps

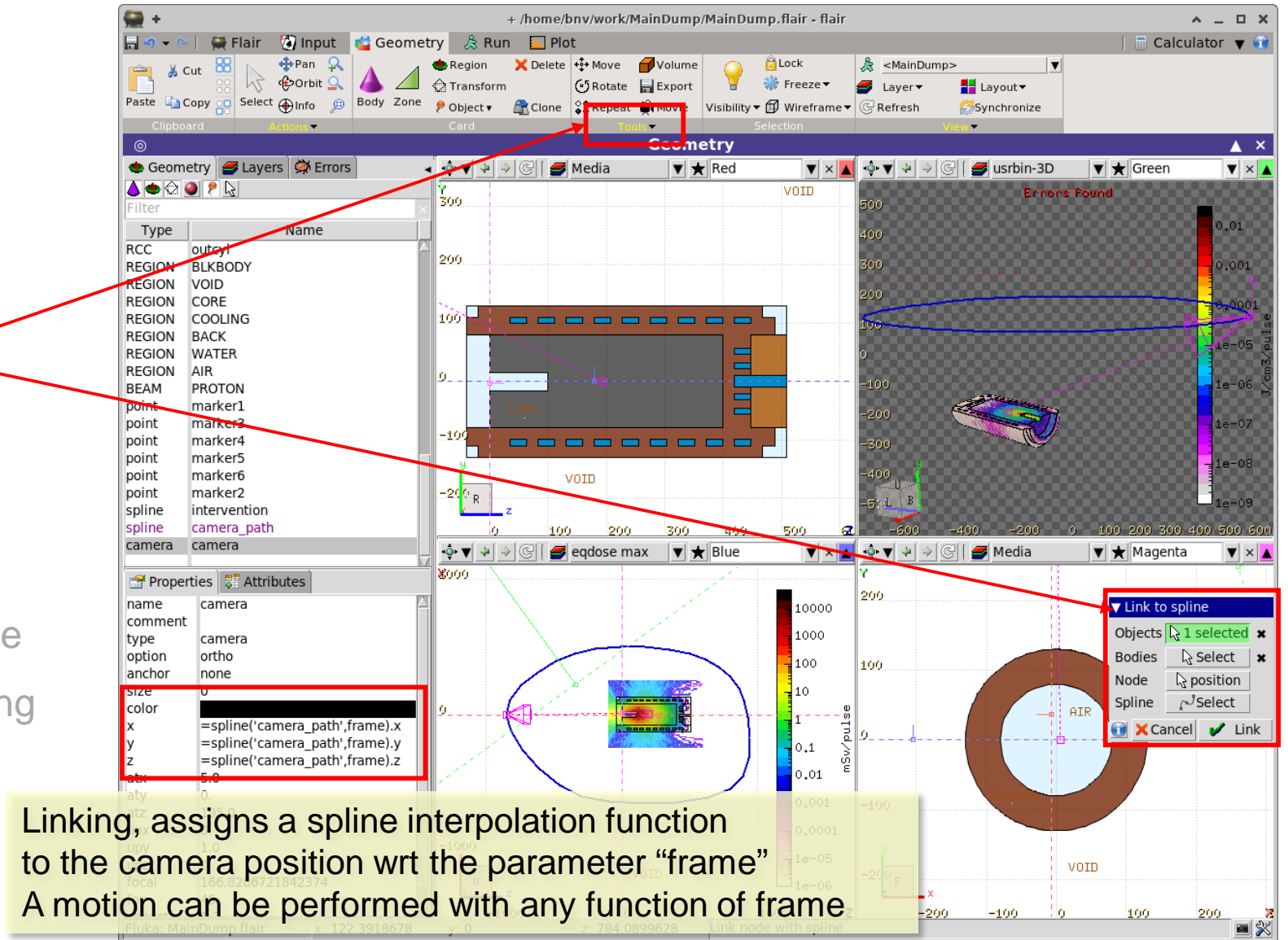
1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool



Geometry Editor: movie creation

Steps

1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool



Geometry Editor: movie creation

Steps

1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool

The screenshot displays the FLUKA Geometry Editor interface with several key components highlighted:

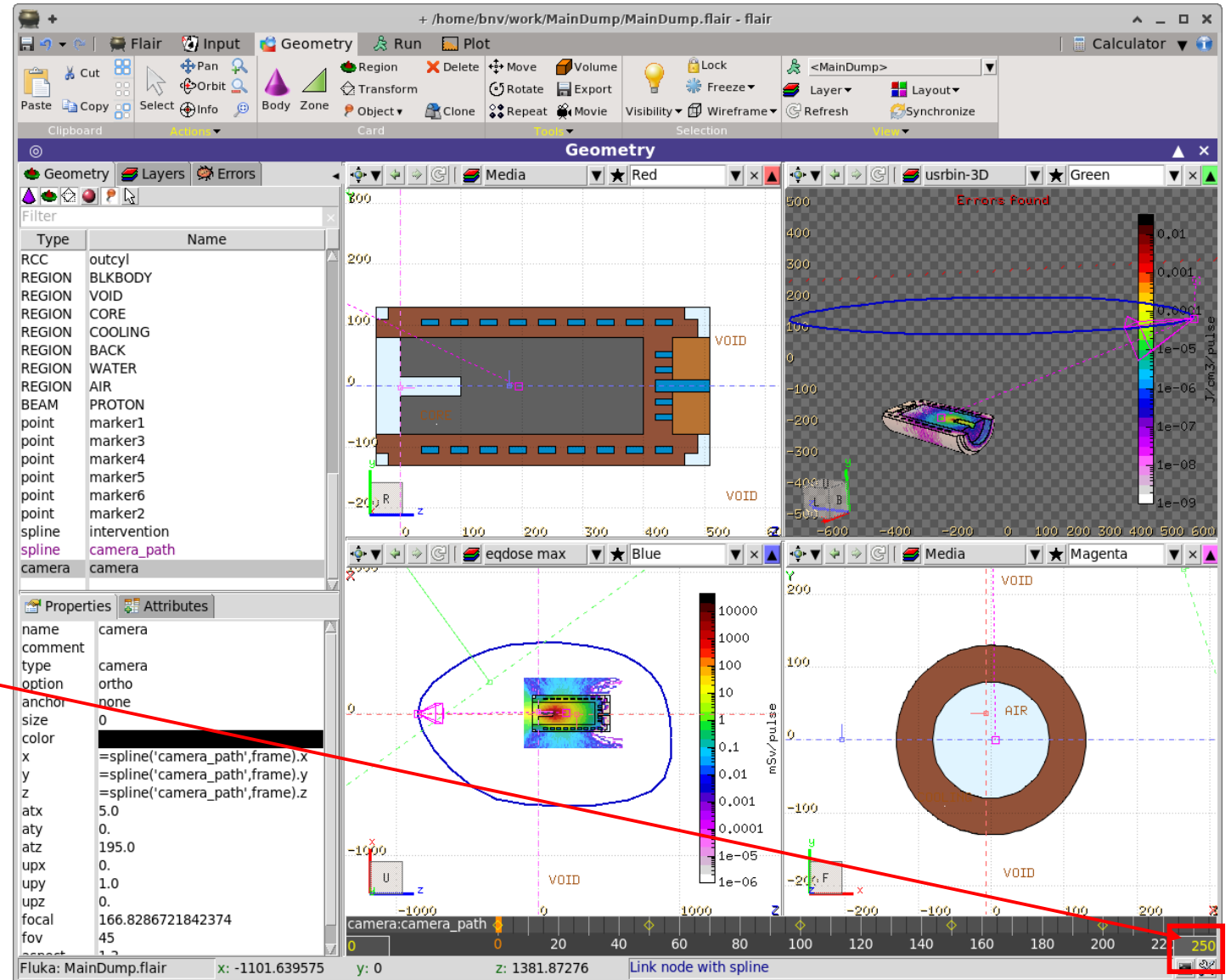
- Top Panel (Spline Properties):** A yellow box contains the following text:


```
Camera path around the target
Time refers to frames, 1s=24frames
Last entry is needed for closing the loop
~ :spline camera_path Xo: 0.0 Yo: 500.0 Zo: -850.0
                                option: Spline Closed anchor: 0 color: #0000c6
                                tension: # segments:
# node(s): 7
node:1.t: 0.0 x: 0.0 y: 0.0 z: 0.0
node:2.t: 50.0 x: -600.0 y: 0.0 z: 550.0
node:3.t: 100.0 x: -550.0 y: 0.0 z: 1650.0
node:4.t: 150.0 x: 100.0 y: 0.0 z: 1750.0
node:5.t: 200.0 x: 450.0 y: 0.0 z: 1350.0
node:6.t: 250.0 x: 450.0 y: 0.0 z: 500.0
node:7.t: 251 x: 0 y: 0 z: 0
```
- Left Panel (Object List):** Lists various regions and objects:
 - REGION VOID
 - REGION CORE
 - REGION COOLING
 - REGION BACK
 - REGION WATER
 - REGION AIR
 - BEAM PROTON
 - point marker1
 - point marker3
 - point marker4
 - point marker5
 - point marker6
 - point marker2
 - spline intervention
 - spline camera_path
 - camera camera
- Properties Panel:** Shows camera settings:
 - name: camera
 - comment:
 - type: camera
 - option: ortho
 - anchor: none
 - size: 0
 - color: [black swatch]
 - x: =spline('camera_path',frame).x
 - y: =spline('camera_path',frame).y
 - z: =spline('camera_path',frame).z
 - atx: 5.0
 - aty: 0.
 - atz: 195.0
 - upx: 0.
 - upy: 1.0
 - upz: 0.
 - focal: 166.8286721842374
 - fov: 45
 - aspect: 1.2
- 3D Viewports:**
 - Top Viewport:** Shows a 3D model of a detector with a blue spline path and a camera icon. A red box highlights the 'View' button in the toolbar.
 - Bottom Viewport:** Shows a 2D projection of the detector with a blue spline path and a camera icon. A red box highlights the 'Link node with spline' button in the toolbar.
- Timeline:** A red box highlights the timeline at the bottom, showing keyframes for the camera path at frames 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, and 250.

Geometry Editor: movie creation

Steps

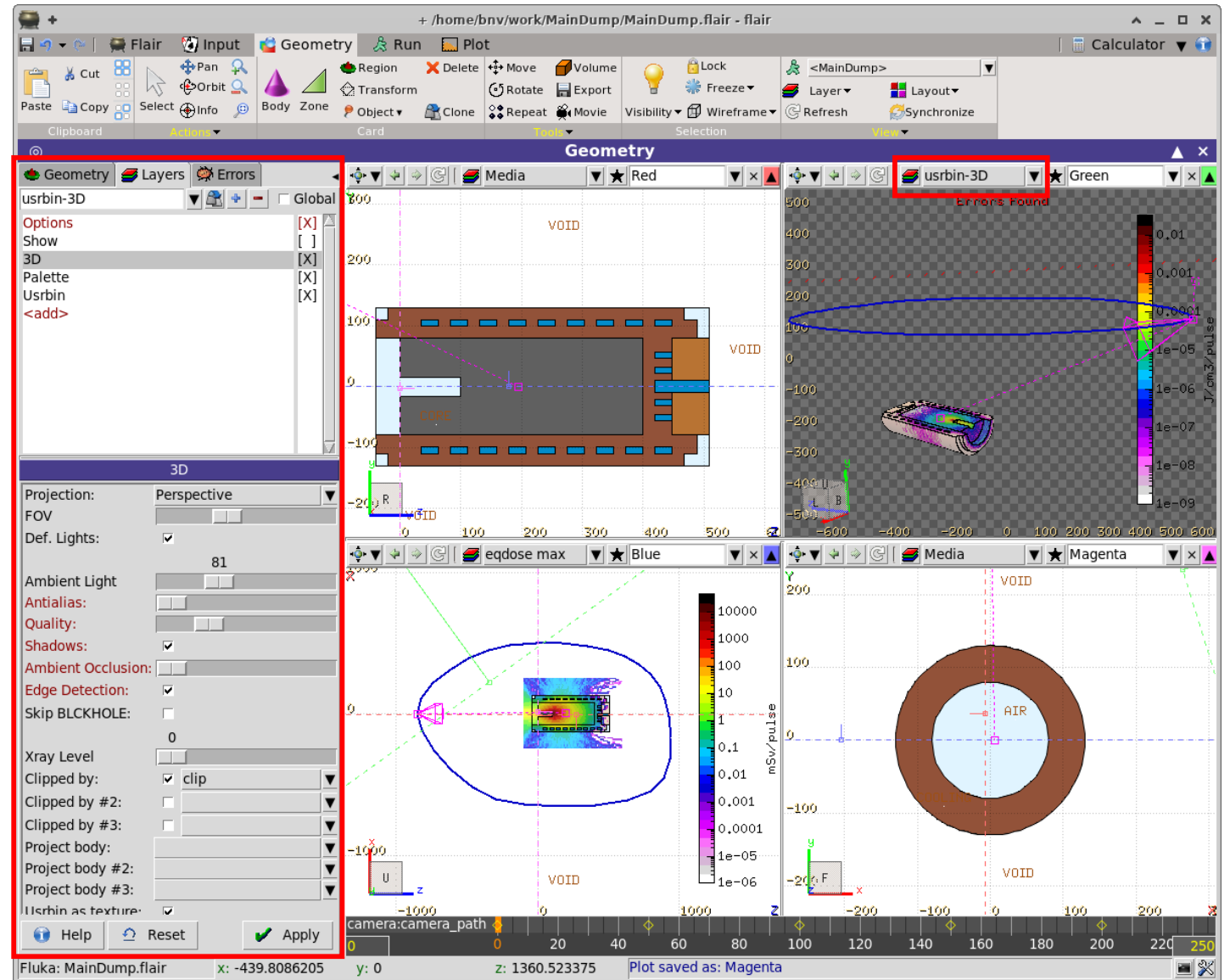
1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool



Geometry Editor: movie creation

Steps

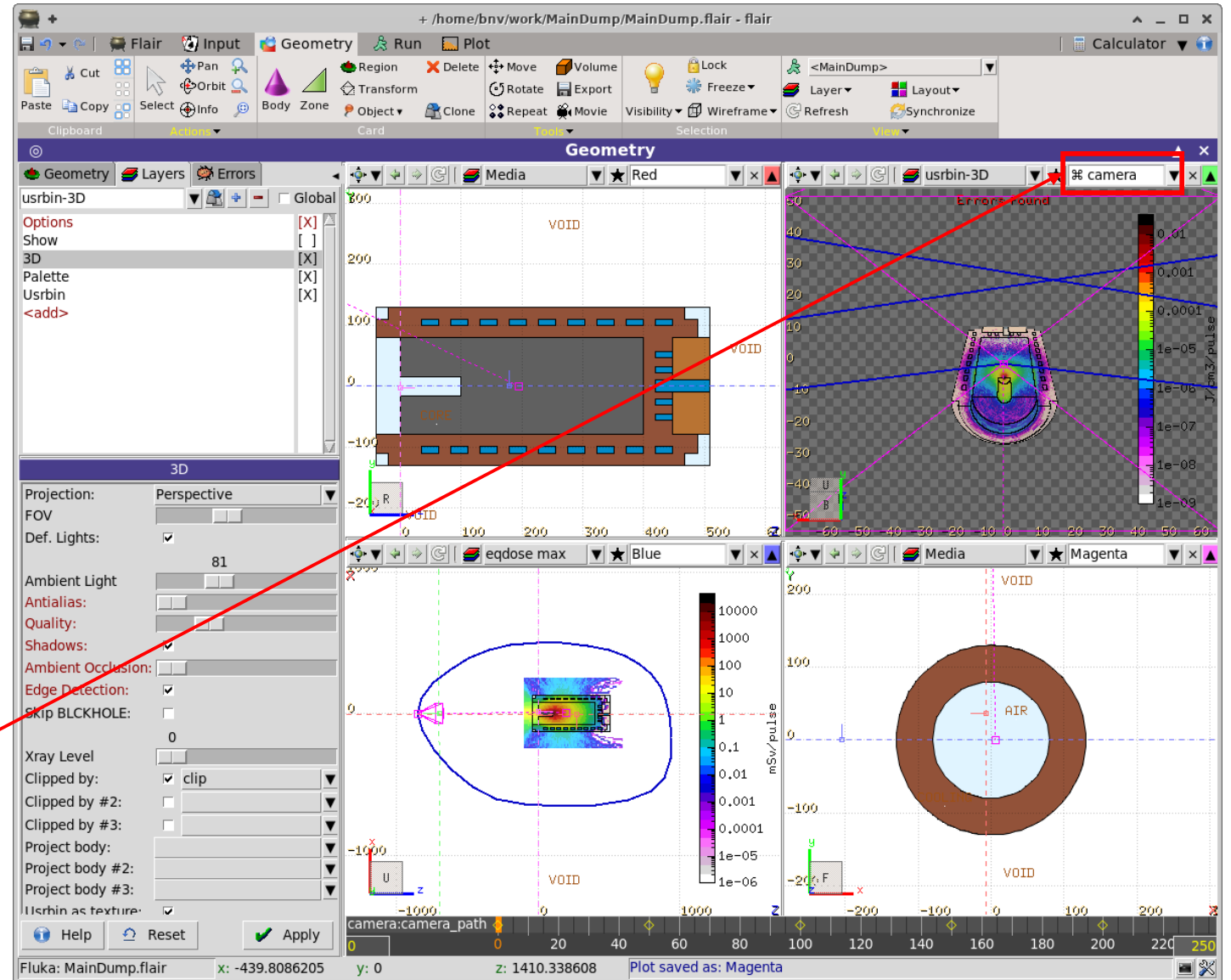
1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. **Customize a 3D layer for the rendering**
7. View the camera from on viewport
8. Open the Movie tool



Geometry Editor: movie creation

Steps

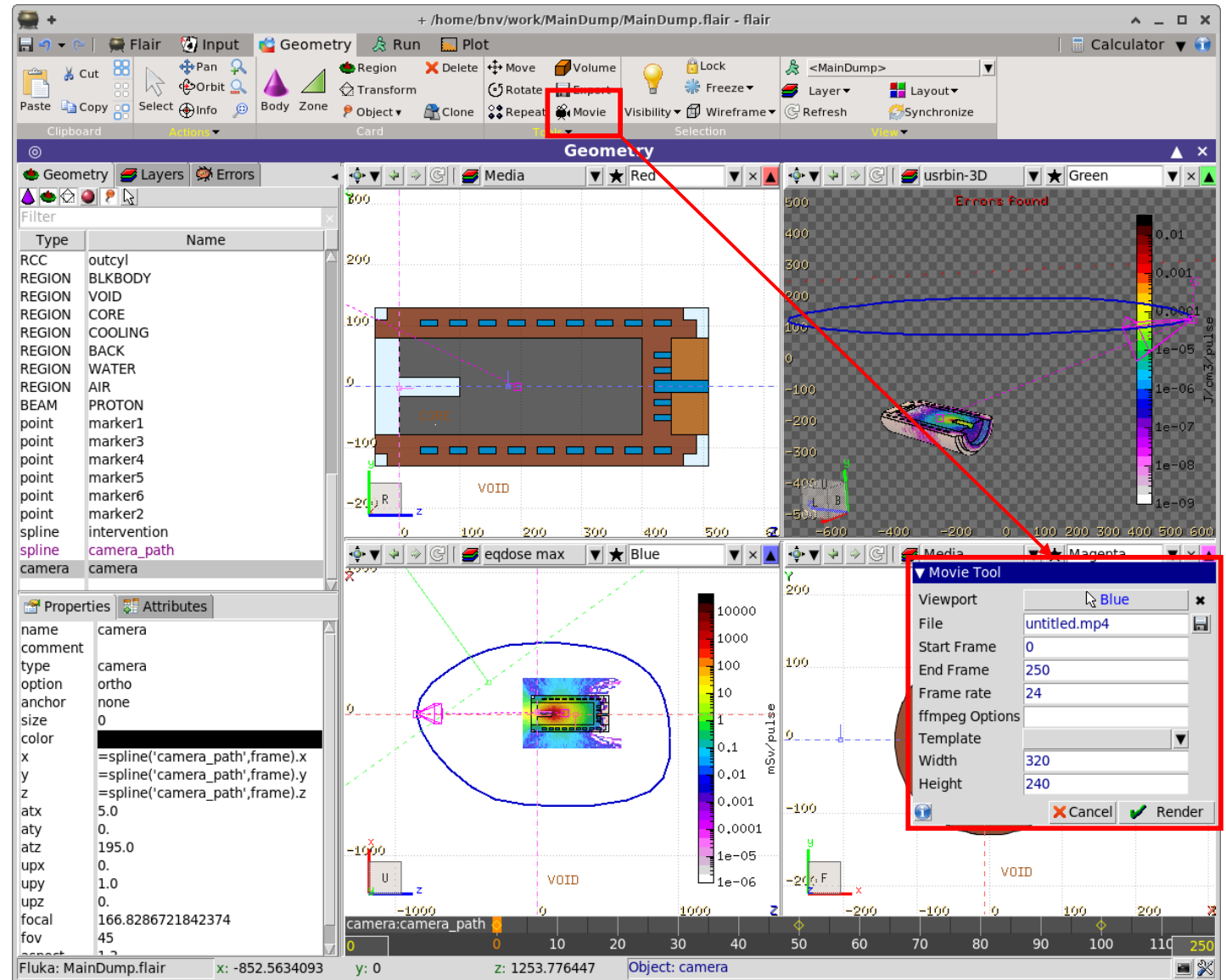
1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool



Geometry Editor: movie creation

Steps

1. Create a spline path
Object → Spline
2. Create a camera
Object → Camera
3. Link the Camera with the Spline
4. Adjust the spline keyframes in time
(EU: 1s=24 frames USA: 29.97)
 - Either manually in the Input editor
 - or with the use of the keyframe
5. Adjust the # of frames in the keyframe
6. Customize a 3D layer for the rendering
7. View the camera from on viewport
8. Open the Movie tool



Material Database

- Flair contains an internal database of ~500 predefined materials and/or compounds
- Some (~300) with the Sternheimer parameters

Please use these data as Reference only!

- Validate always the correctness of the data
- If errors found please contact the author

- The database can be edited, and populated with your own materials. In this case a local copy of the database will be made in ~/.flair directory

Custom Material database

The screenshot displays the FLUKA Materials database interface. The main window is titled 'Materials' and contains a search bar, a group selection list on the left, and a 'Material List' table. The 'Material List' table has columns for 'Material', 'Density', and 'Stoichiometry'. The 'Cyclobutane' entry is highlighted in blue. Below the table, the 'Material Properties' section is visible, showing the title 'Cyclobutane' and tabs for 'Notes', 'Stoichiometry', and 'Properties'. The 'Notes' tab is active, displaying the chemical formula C₄H₈ and a structural diagram of cyclobutane.

Material	Density	Stoichiometry
Mercury	13.546	Hg
728 Cyclohexanone	0.9478	H-10, C-6, O-1
Lead	11.35	Pb
Thallium	11.72	Tl
Cyclobutane	0.00125	H-8, C-4
1-Chlorobutane	0.8862	H-9, C-4, Cl-1
Sodium nitrate Na_N_O3	2.261	N-16.5, O-56.5, Na-27
Thulium	9.321	Tm
478 Hexene	0.673	C-6, H-12
Butane C4_H10	0.0024934	H-17.3, C-82.7

Material Properties
Title: Cyclobutane
Notes: Chemical Formula
C H
4 8
Names: Cyclobutane

- Create your own custom material database to be shared among flair projects.
- Either edit manually the materials or
- Import them from existing inputs

