ANTS2 toolkit

Interface to Geant4, event analyzer and semi-automatic detector optimization

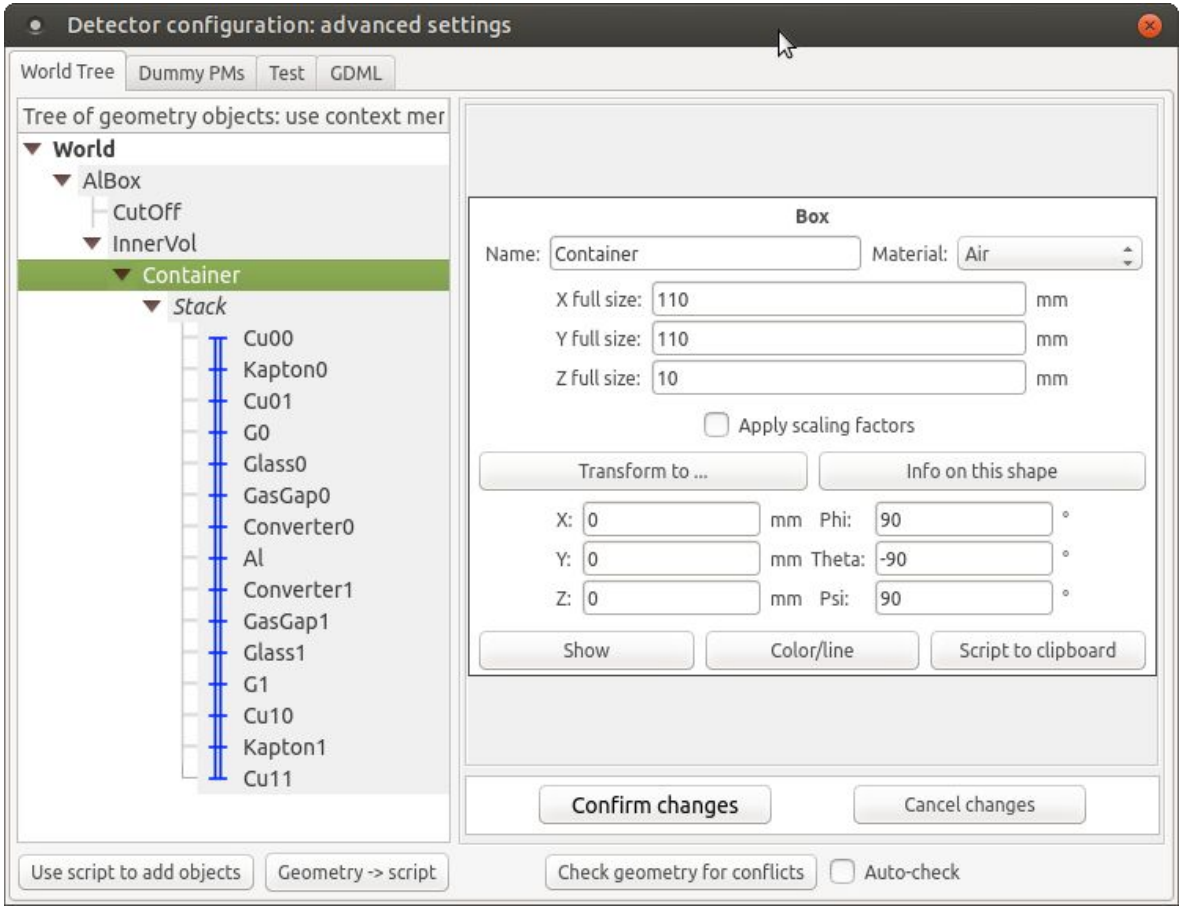
A. Morozov and V. Solovov

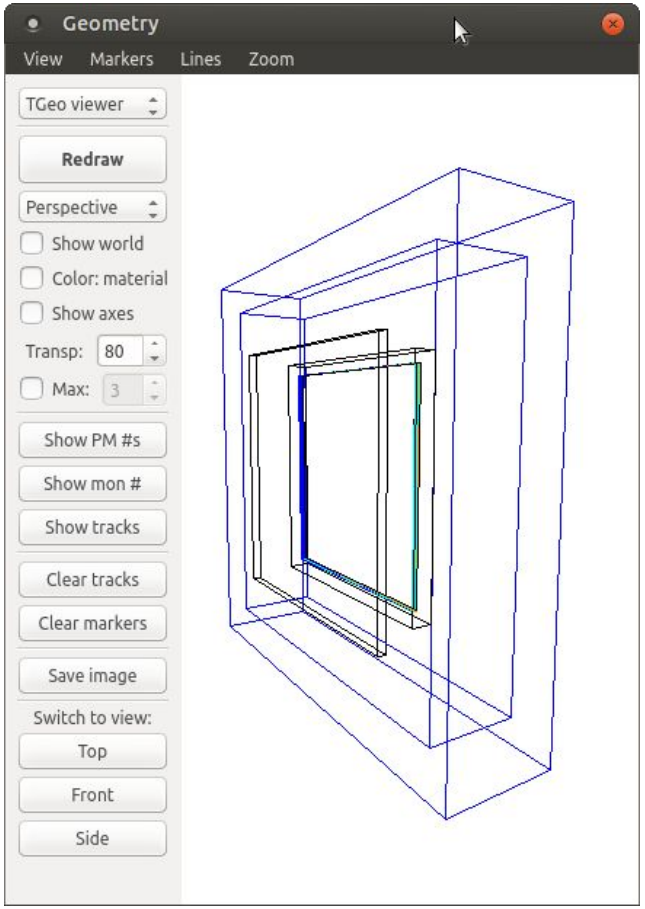
LIP, Lisbon, 24.10.2019

Contents

- Geometry: advanced GUI tools
- ANTS2 - Geant4 interface
- Event viewer and history analyzer
- Semi-automatic detector optimization

Geometry: advanced GUI tools

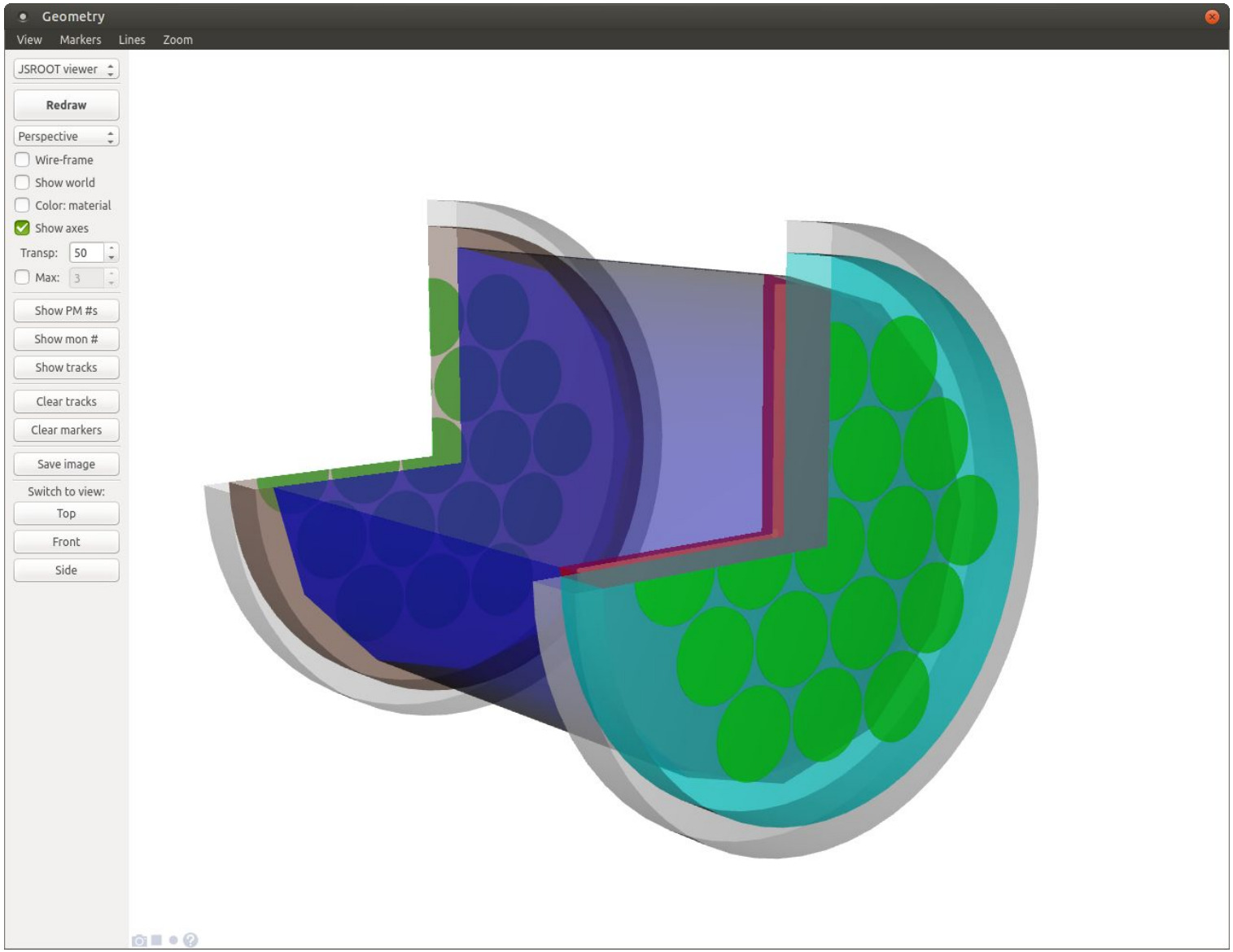




- Tree view of the detector volume hierarchy
- Context menu to add / remove / duplicate / disable volumes
- Drag-and-drop to move or reorder objects
- Property edit box for the selected object

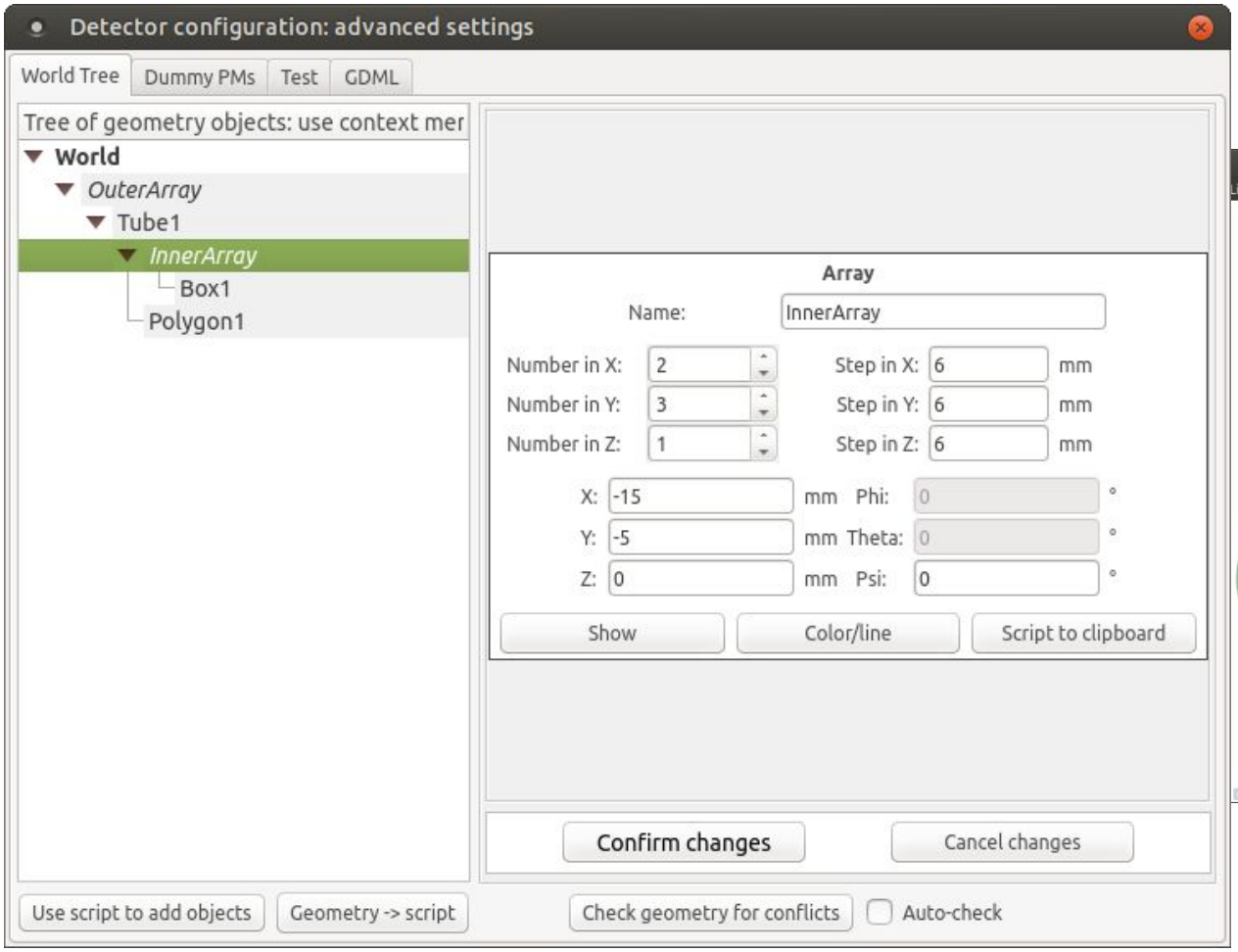
Geometry: advanced GUI tools

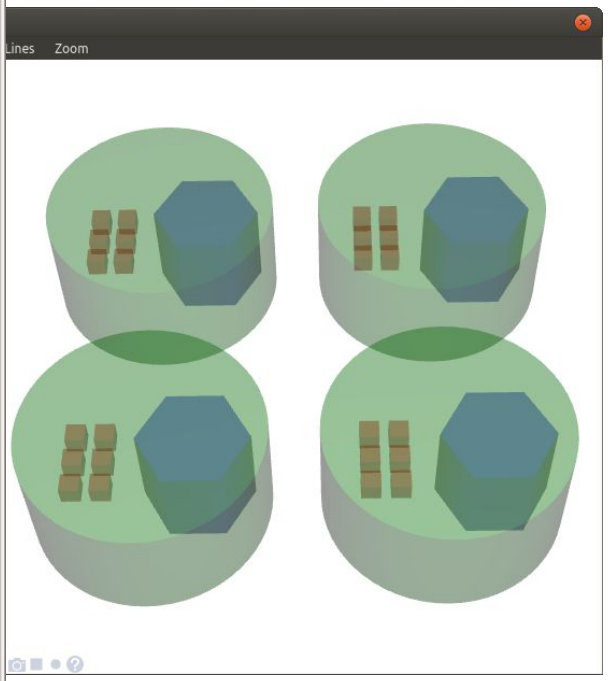
JSROOT visualization



Geometry: advanced GUI tools

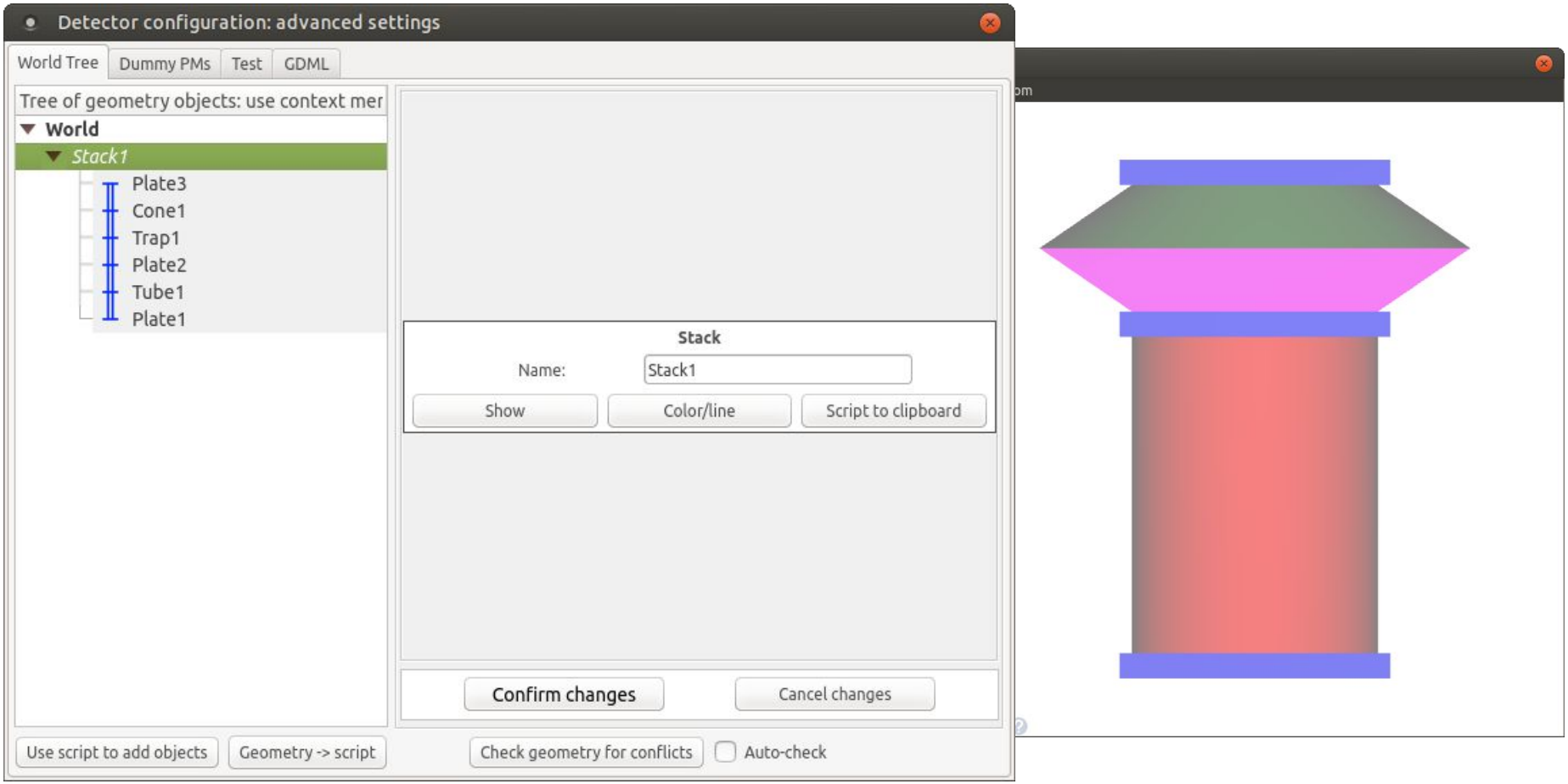
Arrays of objects





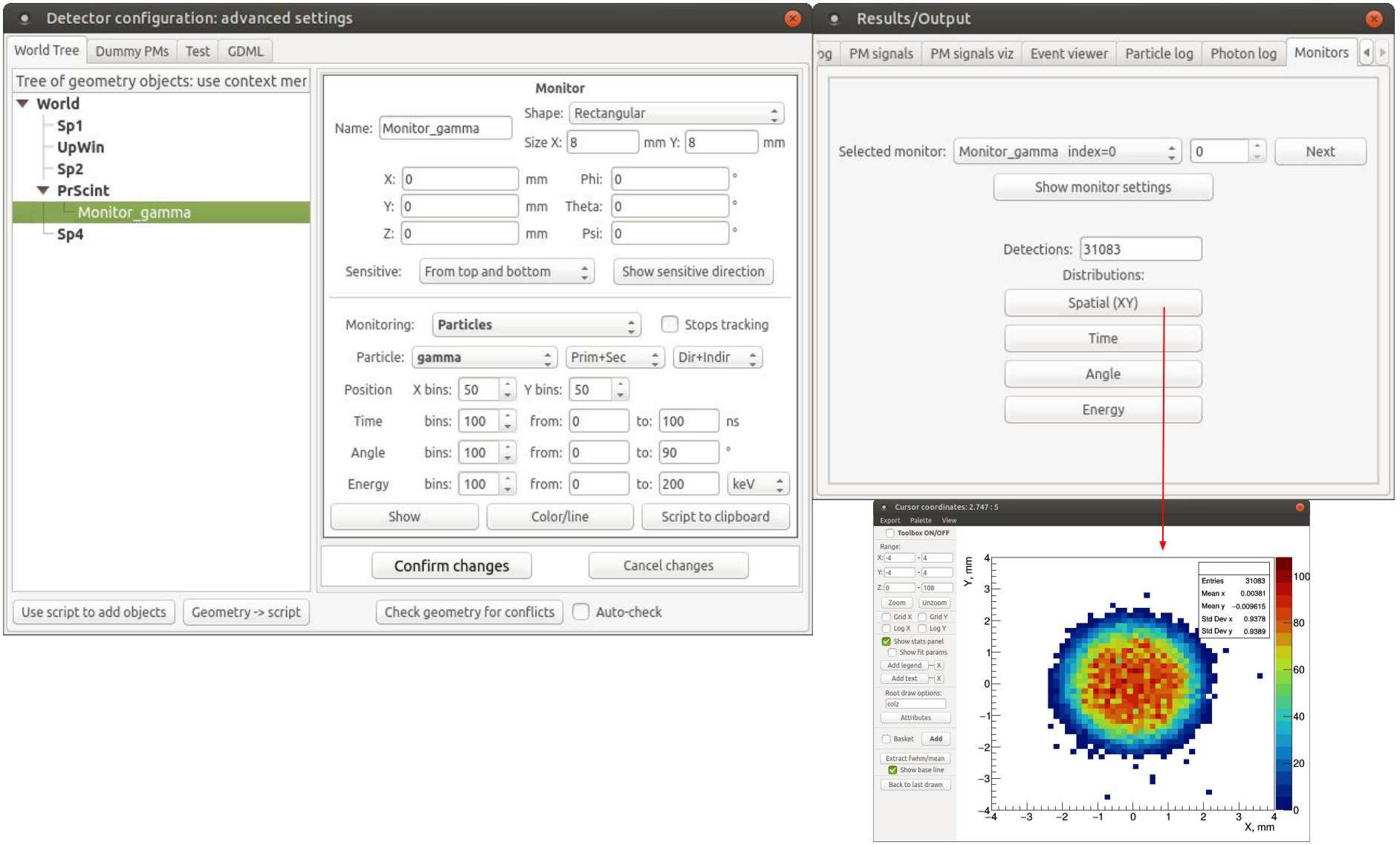
Geometry: advanced GUI tools

Stacks of objects



Geometry: advanced GUI tools

Monitor objects



ANTS2 - Geant4 interface

Goal:

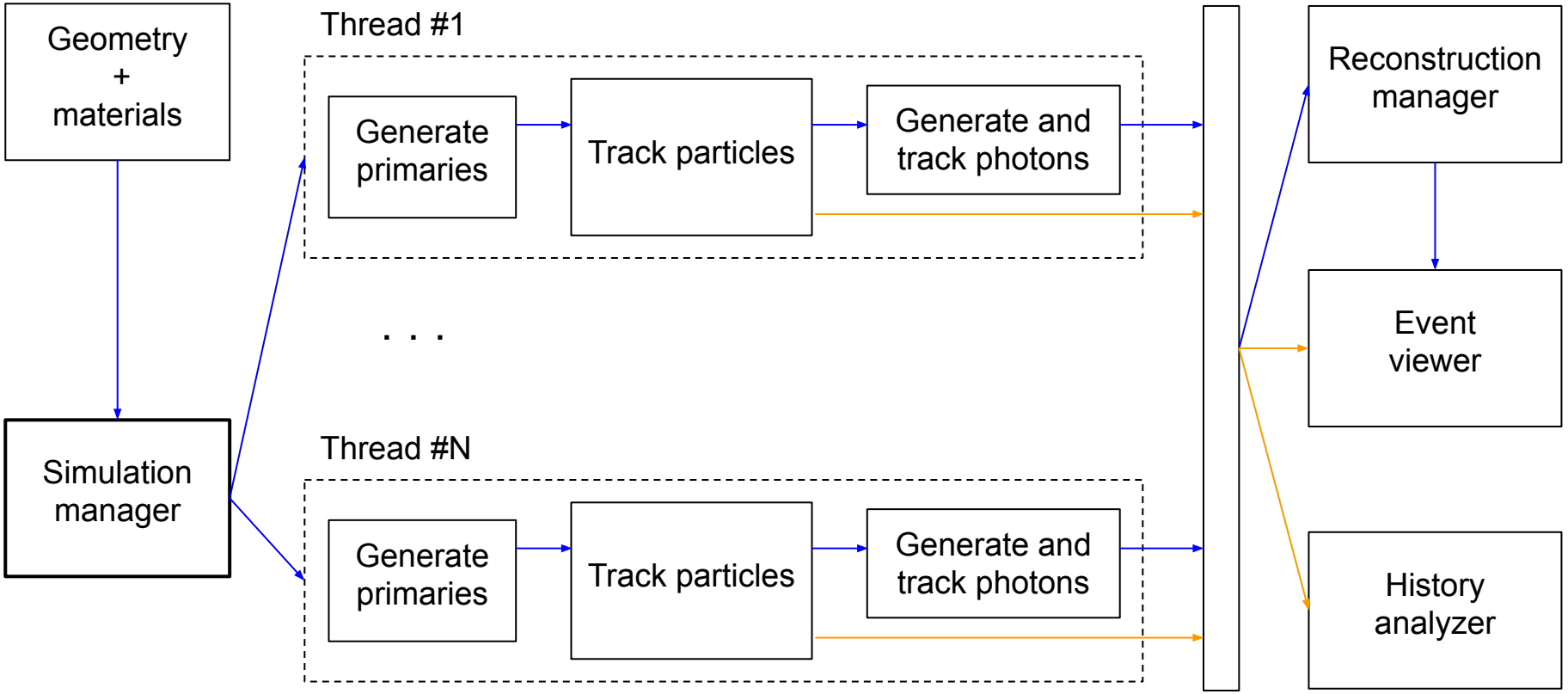
Delegate particle simulations from ANTS2 to Geant4 and import back the energy deposition data (optionally also data from monitors / tracking history)

Challenges:

- ANTS2 uses TGeoManager 3D navigator from CERN ROOT, so detector geometry is configured in TGeo format
  - ANTS2 also uses custom system of materials
- ANTS2 have many options for primary particle generation (e.g., models of radioactive sources and script-based generator)
- Need to configure Geant4 simulation without C++ code re-compilation
- Need compatibility with ANTS2 multithreading and simulations on ANTS2 grid

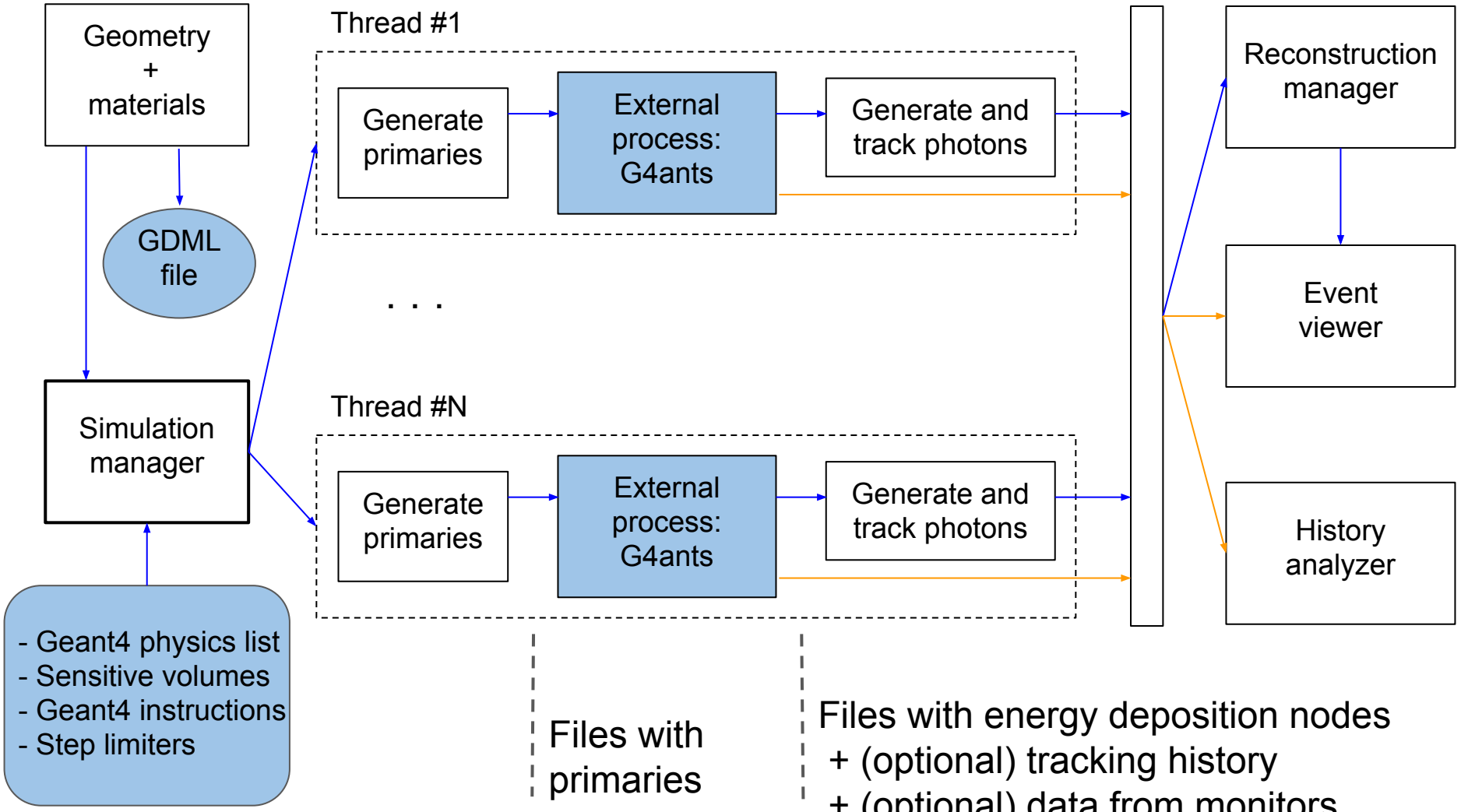
ANTS2 - Geant4 interface

ANTS2 without Geant4:



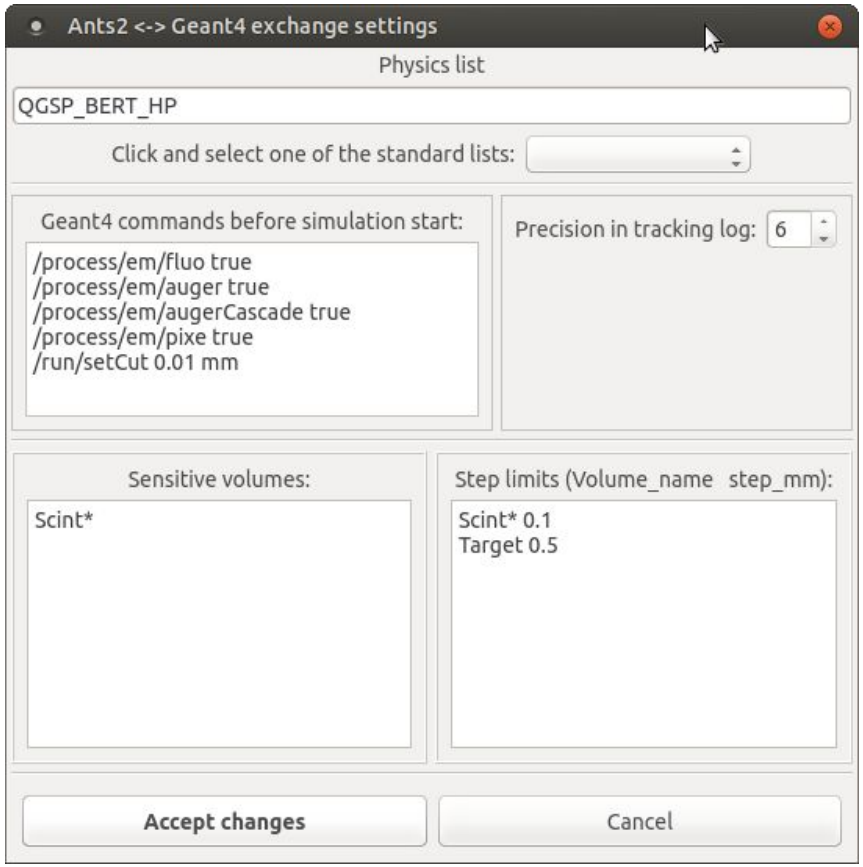
ANTS2 - Geant4 interface

ANTS2 with Geant4:



ANTS2 - Geant4 interface

ANTS2 GUI: configuration of Geant4 simulation settings

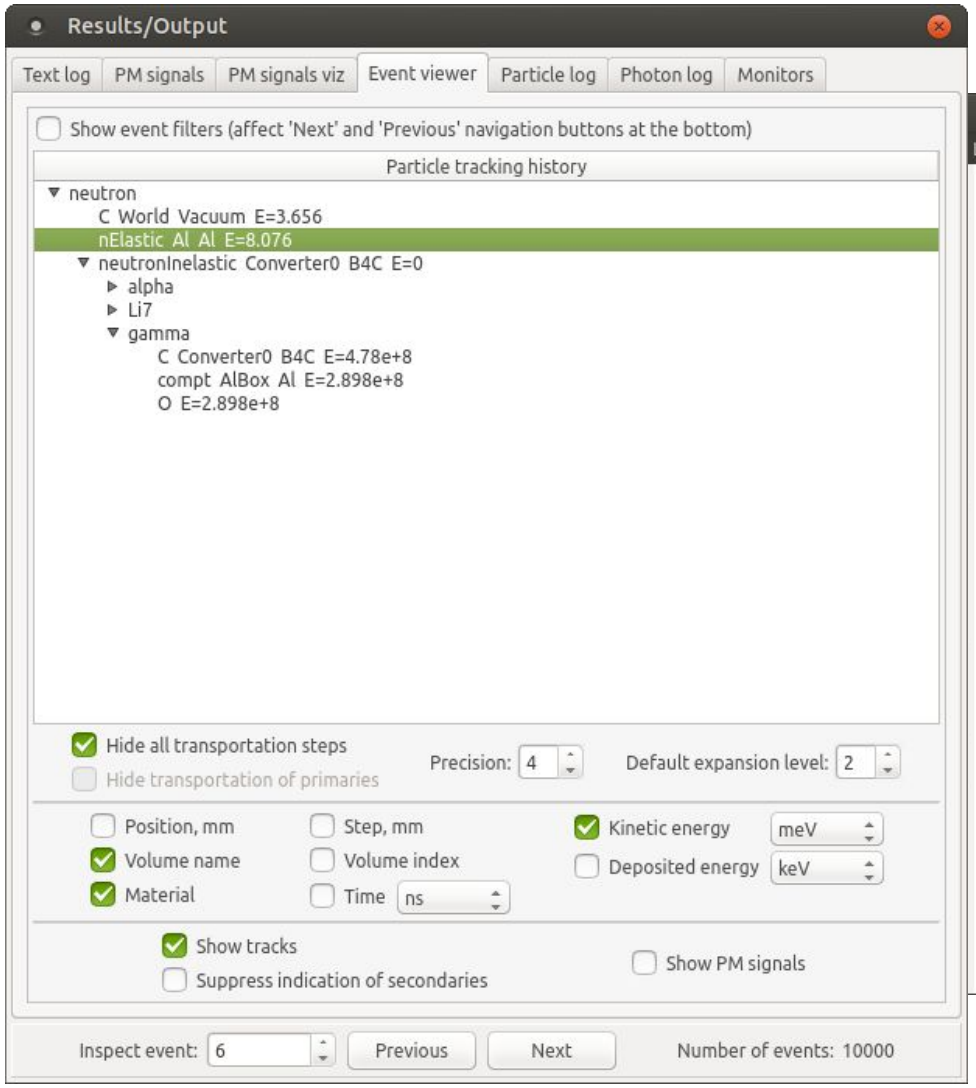


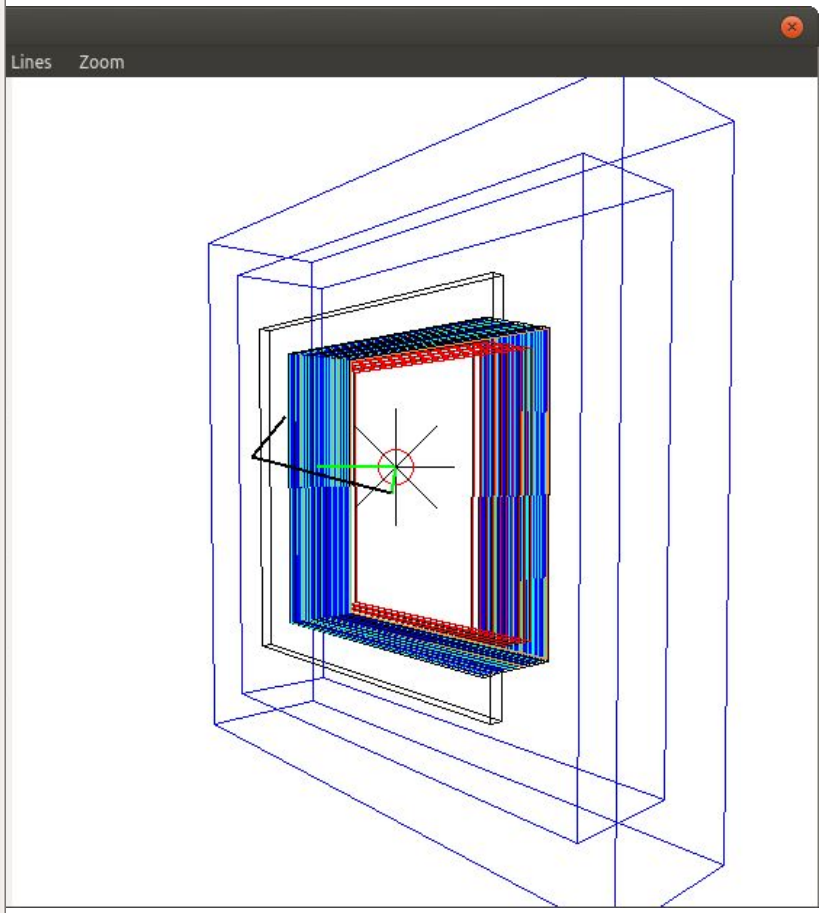
G4ants executable:

- Reads configuration file
- Generates Geant4 detector geometry and materials described in the GDML file
- Configures physics list and sets step limiters
- Creates Geant4 SensitiveDetectors
  - for each sensitive volume: to collect energy deposition data
  - for each monitor object (if present)
- Configures SteppingAction (if tracking history collection is activated)
- Executes additional Geant4 configuration instructions given by the user
- For each event, reads records of the primaries from the file and simulates them one by one

Event viewer and history analyzer

Event Viewer: tree view of all transportation and interaction history. Flexible configuration to show only relevant information. Track view with step indication.





Event viewer and history analyzer

Event analyzer: “In volumes” tab

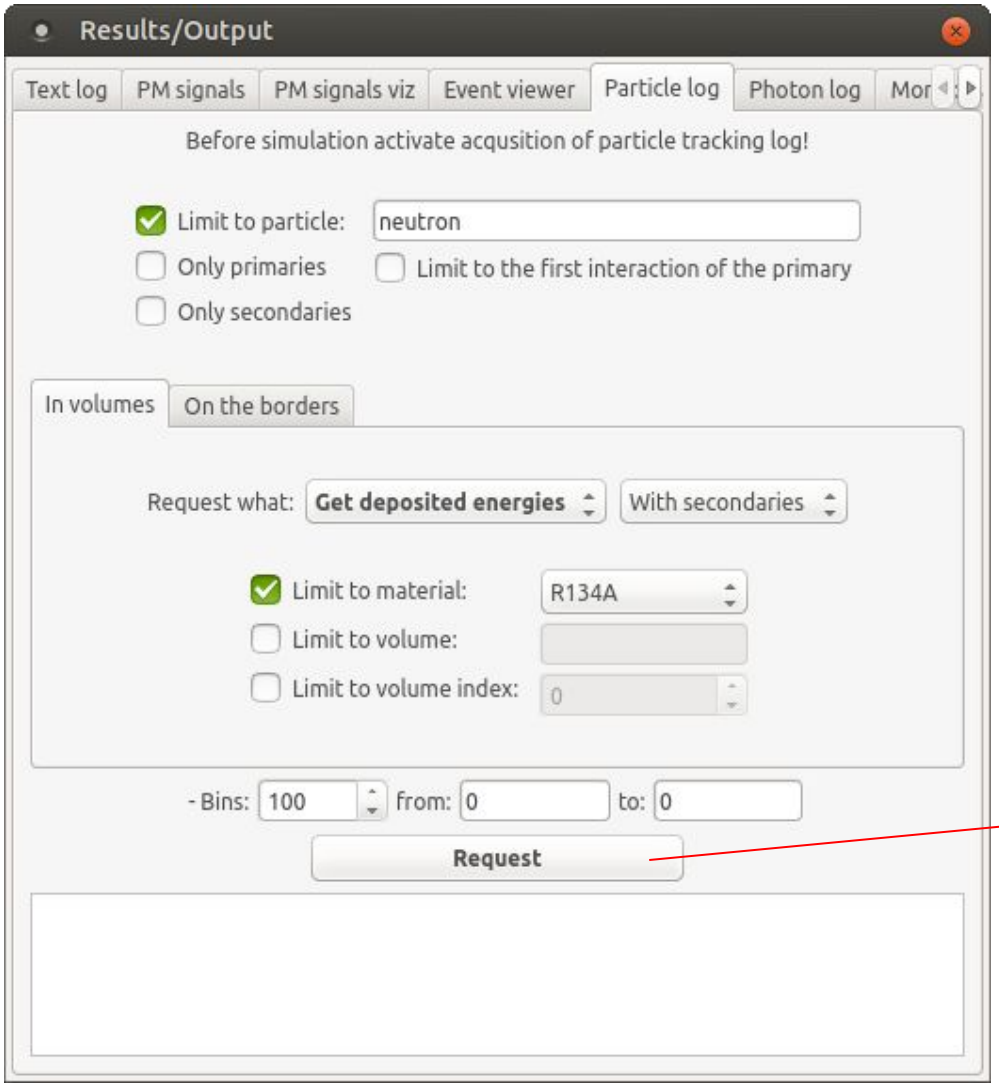
Statistics on:

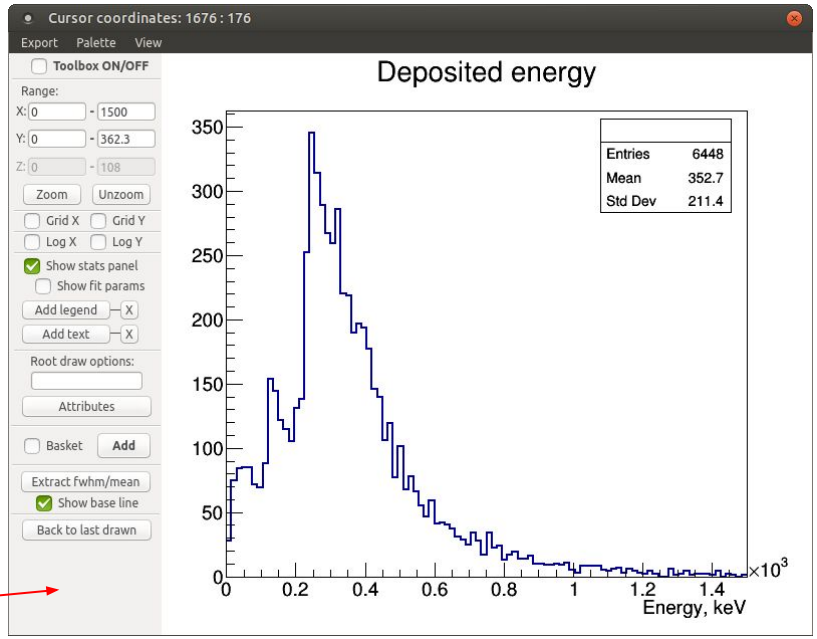
- Particles seen
- Processes seen
- Traveled distance
- Deposited energy

Very flexible cuts!

Event viewer and history analyzer

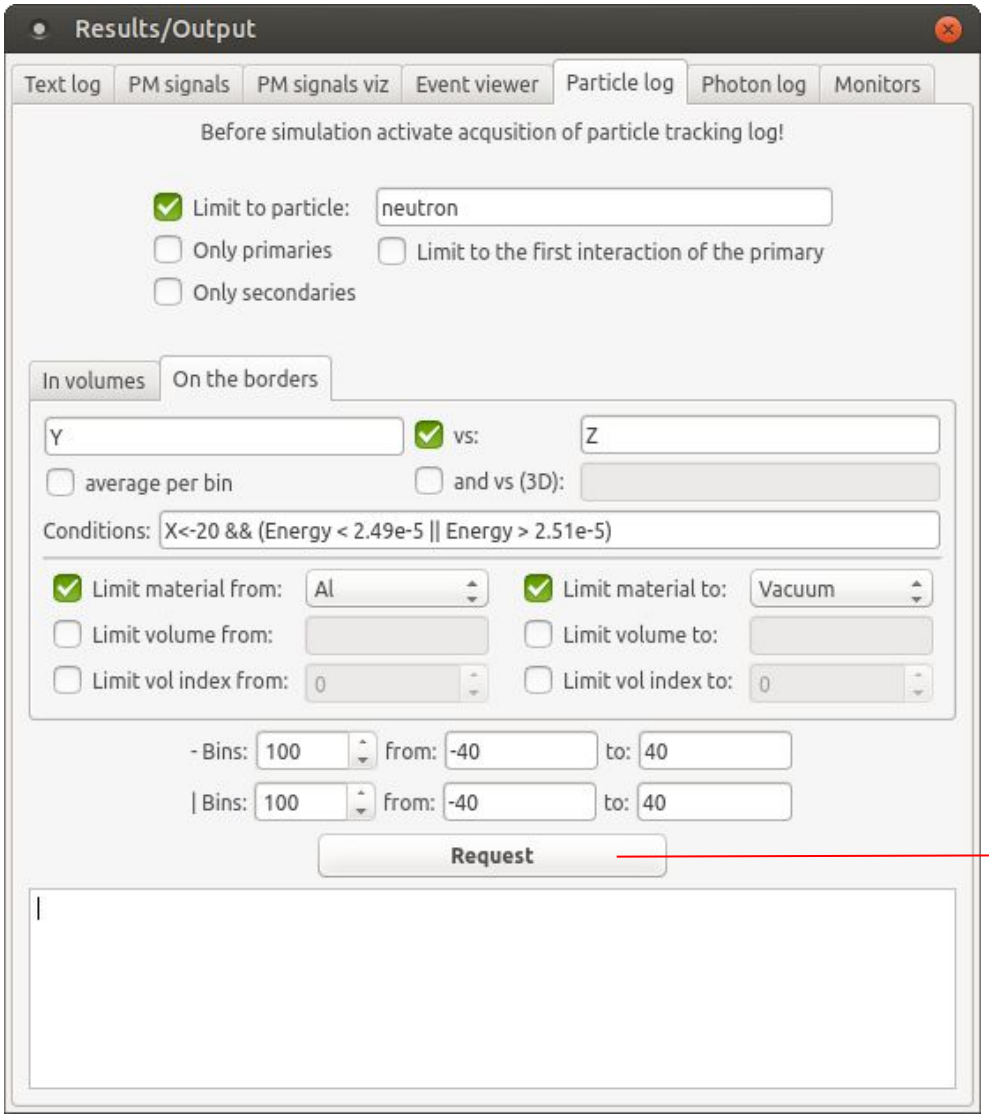
Event analyzer: “In volumes” tab





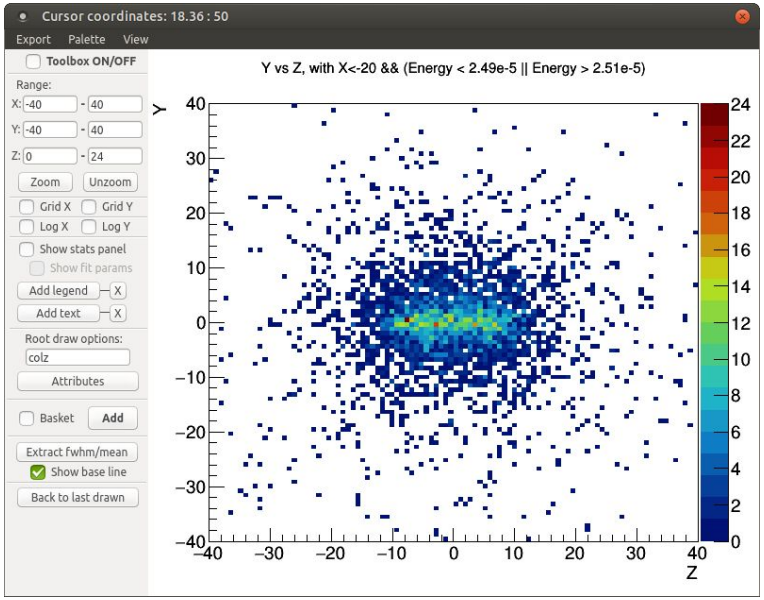
Event viewer and history analyzer

Event analyzer: “On the borders” tab. Statistics for particles crossing volume borders



Requests / cuts:

Arbitrary combination of parameters, including position (***X***, ***Y***, ***Z***), ***Energy***, ***Time***, and direction vector components (***Vx***, ***Vy***, ***Vz***) and **mathematical expressions** and **logical operations** supported by TFormula class of ROOT, e.g., 2 * pi * sqrt(X^2 + Y^2)



Spatial distribution of back-scattered neutrons

Semi-automatic detector optimization

Detector optimization over several “entangled” parameters:

- **Brute force approach**: perform simulations covering a large grid in a multi-dimensional parameter space
- A smarter approach: implement a **minimization algorithm** operating with a custom cost function

  On each call from the minimizer, the function

  - ➢ receives a value for each optimization parameter
  - ➢ modifies the detector model according to the parameter values
  - ➢ runs a simulation
  - ➢ processes the results
  - ➢ calculates the value of a user-defined *goodness parameter* and returns it to the minimizer

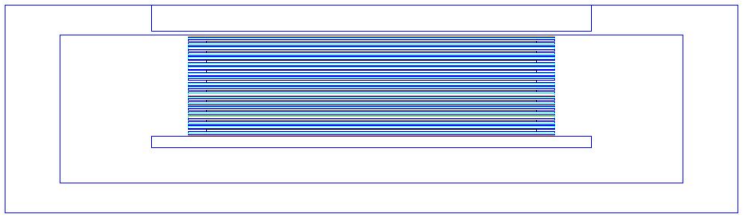
Semi-automatic detector optimization

In ANTS2 this minimization-based approach is realized using the scripting system (JavaScript or Python):

- Access is provided to a CERN ROOT minimizer
- An arbitrary number of parameters can be configured
  - These parameters can have a defined upper and/or lower bounds
- Custom cost function is defined directly in the script (no compilation is needed)
  - It is straightforward to change any detector property: configuration is stored in a JSON object (collection of key/value pairs)
  - Scripting gives access to the simulation and reconstruction managers, the data hub and the history analyzer

Semi-automatic detector optimization

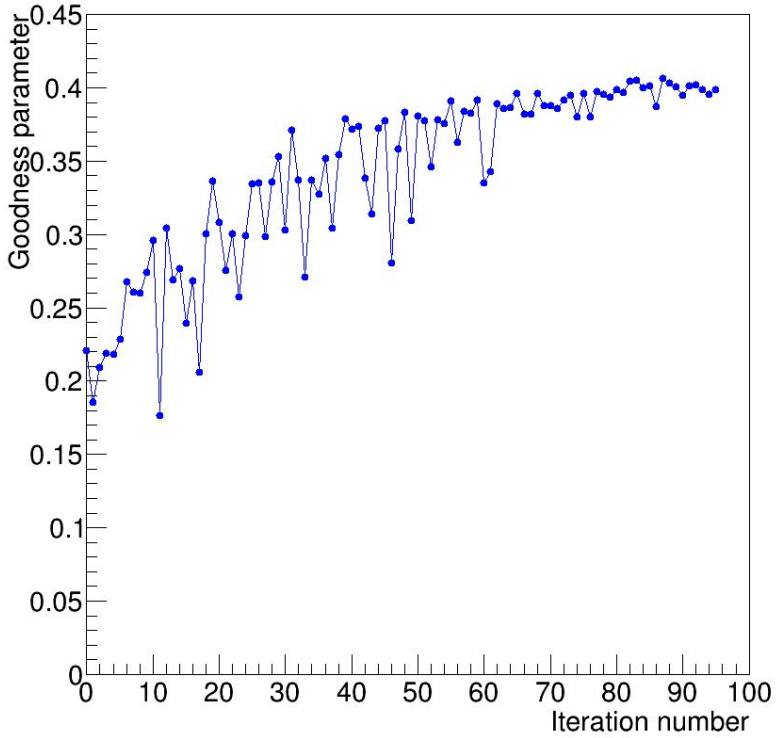
An example: Optimization of an RPC neutron detector



Parameters: 5 thicknesses of neutron converter layers

Cost function: A product of an *equality parameter* and the total detection efficiency:

we target to equalize the count rate of all individual RPCs as much as possible maintaining high detection efficiency



- Optimization started from converter layer thickness of 1 μm for all RPCs
- After 87 iteration (20 minutes) an optimum was found with layer thicknesses of 0.36, 0.46, 0.72, 1.23 and 2.54 μm
- Compared to the detector prototype, we were able to improved the equality parameter by a factor of 2.5

Future work

- Develop web interface for ANTS2 targeting didactic / outreach activities
- Implement VecGeom geometry manager and navigator for direct compatibility with Geant4
- Implement tracing of optical photons on GPU (ambitious!)
- ANTS2 as a Python package?

Open source at Github:

https://github.com/andrmor/ANTS2 (use Dev branch for the latest features)
https://github.com/andrmor/G4ants

If you have questions, send them to andrei@coimbra.lip.pt