GUI Parallel Session

Hajime Yoshida

- Practical Usage of Geant4Py presented by Koichi Murakami
- Users comments by Michel Maire
- Hot discussions
  - Ana, John, John, Gabriele, Michel, Joseph, Fang, Witold, Takashi, Vladimir, Koichi, Hajime,

Contents

- Installation notes
- Exposed classes/methods in use cases
- Wrapping out your applications
- Connection to analysis tools
- Examples

How to Build Geant4Py

- There is a configuration script for building the package.
  - ✓ configure --help shows more detailed options.

```
# ./configure linux
  --with-g4-incdir=/opt/heplib/Geant4/geant4.8.1/include
  --with-g4-libdir=/opt/heplib/Geant4/geant4.8.1/slib/Linux-g++
  --with-clhep-incdir=/opt/heplib/CLHEP/2.0.2.3/include
  --with-clhep-libdir=/opt/heplib/CLHEP/2.0.2.3/lib
  --with-clhep-lib=CLHEP-2.0.2.3
```

-
  - ✓ In case of both libXXX.a and libXXX.so existing, linker will link with the shared library.
    - » libCLHEP.a : link to libCLHEP-2.0.2.3.a
    - » libCLHEP.so -> remove it
    - » libCLHEP-2.0.2.3.so // use it in case of using shared library

- After executing configure script, you can go ahead to building procedures.

```
# make
# make install
```

What is Exposed to Python

- Currently, over 100 classes over different categories are exposed to Python.
  - Classes for Geant4 managers
    - G4RunManager, G4EventManager, …
  - UI classes
    - G4UImanager, G4UIterminal, G4UIcommand, …
  - Utility classes
    - G4String, G4ThreeVector, G4RotationMatrix, ...
  - Classes of base classes of user actions
    - G4UserDetetorConstruction, G4UserPhysicsList,
    - G4UserXXXAction
      - PrimaryGenerator, Run, Event, Stepping,...
    - » can be inherited in Python side
  - Classes having information to be analyzed
    - G4Step, G4Track, G4StepPoint, G4ParticleDefinition, ...
  - Classes for construction user inputs
    - G4ParticleGun, G4Box, G4PVPlacement, ...
- List of all exposed classes List of all exposed classes of respective Geant4 releases

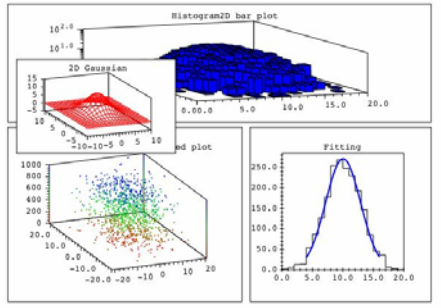
- NOT all methods are exposed.
  - Only safe methods are exposed.
    - Getting internal information are exposed.
    - Some setter methods can easily break simulation results.

How to expose your applications

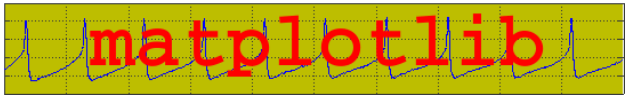
- ExN03 setup as an example
  - Each user component can be build as a Python module.
- Detector Construction
  - site-modules/geometries/ExN03geom/
- Physics List
  - site-modules/physics_lists/ExN03pl/
- Primary Generator Action as particle gun
  - site-modules/primaries/ParticleGun/
  - reusable in most cases

Connection to Analysis Tools

- Analysis tools
  - ROOT-Python interface
    - plot example : examples/emplot/
    - (online) histogram example:
      - examples/demos/water_phantom/
    - tree example:
      - site-modules/utils/MCScore/
  - PAIDA
    - AIDA Python implementation

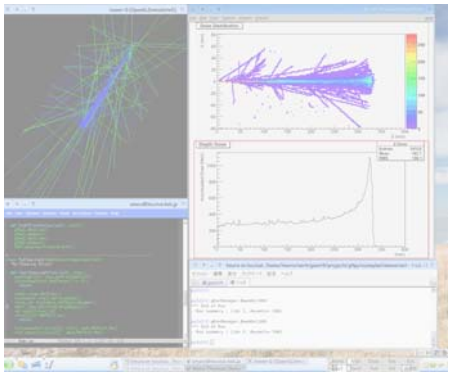


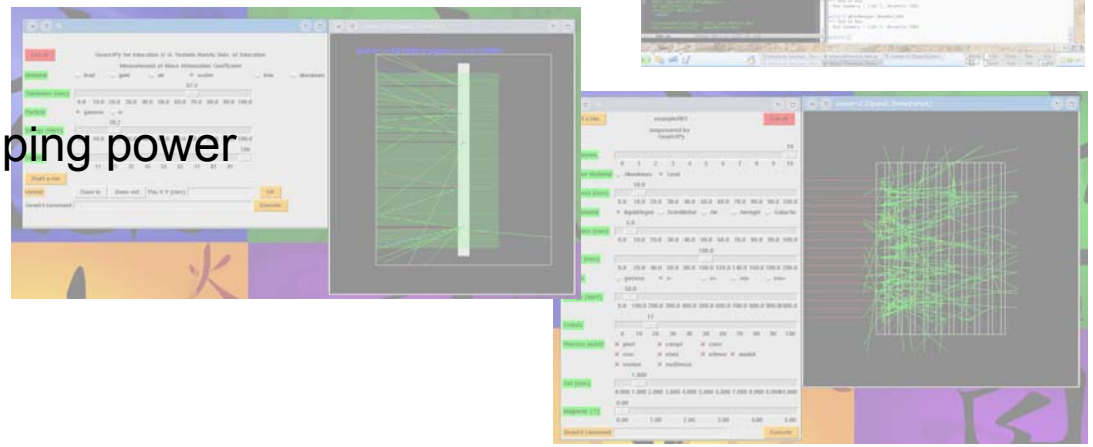
- Plotting tools
  - matplotlib
    - histogramming interface (mathist) is in development.
      - site-modules/utils/mathist

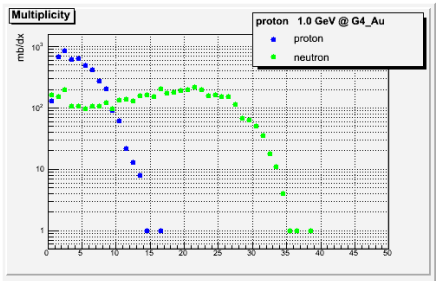


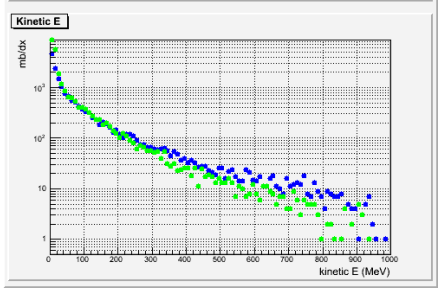
more Example – in examples/

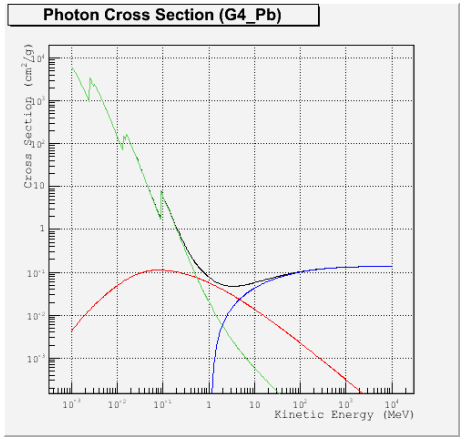
- demos/water_phantom/
  - ✓ This demo program shows that a Geant4 application coworks with ROOT on the Python front end.
- education/lesson1/2
  - GUI example for educational uses
- emplot/
  - photon cross sections/electron stopping power
  - using EZgeom
- hadrontest/
  - proton/neutron/pion production
  - using EZgeom
  - store MC information (histogram/ROOT tree/TXT)

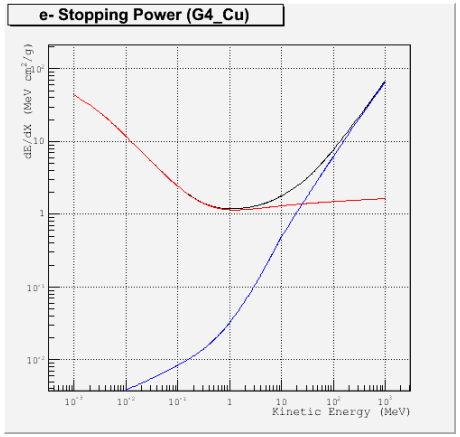












What a user expects from a graphical interface ?

- a toolkit, easy to use, which allows him to built his own interactive Geant4 application
  - easy = by non-expert
- remark 1 : an interactive application include necessarily visualization and analysis tool
- remark 2 : the graphical interactive mode must be compatible with more ‘classical’ approach : commands line or batch
- Compatibility of libraries
- remark 3 : the toolkit itself must be easy to install

discussions

- Integration of GUI, vis and analysis
  - Python provides these tools or interfaces to them
  - OpenInventor for example integrates UI and Vis
- Ease of use of GUI toolkits
  - GUI may depend on specific applications
  - GUI programming requires some expertise
- For easy installation
  - It must be well documented
  - We must use hypernews for better communication
- Another approach of Python interface by PyROOT was commented
  - In cases where C++ classes exist, their dictionaries are created automatically and exposed to Python
  - Personal experience of trouble in installing PyROOT was reported