11th Geant4 Collaboration Workshop

12 October 2006

Universidate Tecnica de Lisboa

Hajime Yoshida

Naruto University of Education, JST

2000 statement on the interactivity refrained

● There is an obvious requirement for Geant4 to fit into existing Interactive Frameworks. Examples of Interactive Frameworks are: Momo, Explorer, AVS, the BaBar Framework, Gaudi, OPACS, JAS, WIRED, ROOT. They each come with their own way of interacting with the applications inside.

● **Geant4 already offers interactivity through intercoms** (G4UImanager, several concrete G4UIsession classes - G4Uterminal, etc.); Momo uses this approach.

● **An alternative approach commonly used is to wrap application classes directly with wrappers/adaptors.**Tools for automatic creation of wrappers exist or are in development, e.g., SWIG (for Tcl, Python, etc.), JACO (Java Access to C++ objects).

● Conclusion: **it appears that Geant4 is now sufficiently open that both techniques can be used but this remains to be tested in a real application**.

This would be sufficient for most frameworks. Reservations which arise from the current dominance of the intercoms way of interacting:

There might be, or might come to be, functionality that is only accessible through intercoms, **which would restrict direct wrapping techniques and other direct object oriented interfacing techniques**.

There is a dual use of intercoms ((a) for inter-category communication and (b) for a command interpreter) and they might need distinguishing in future.

• **Now in 2006, we have Geant4Py. WEb page in KEK.**

•**We found no restrictions in wrapping G4 kernel classes.**

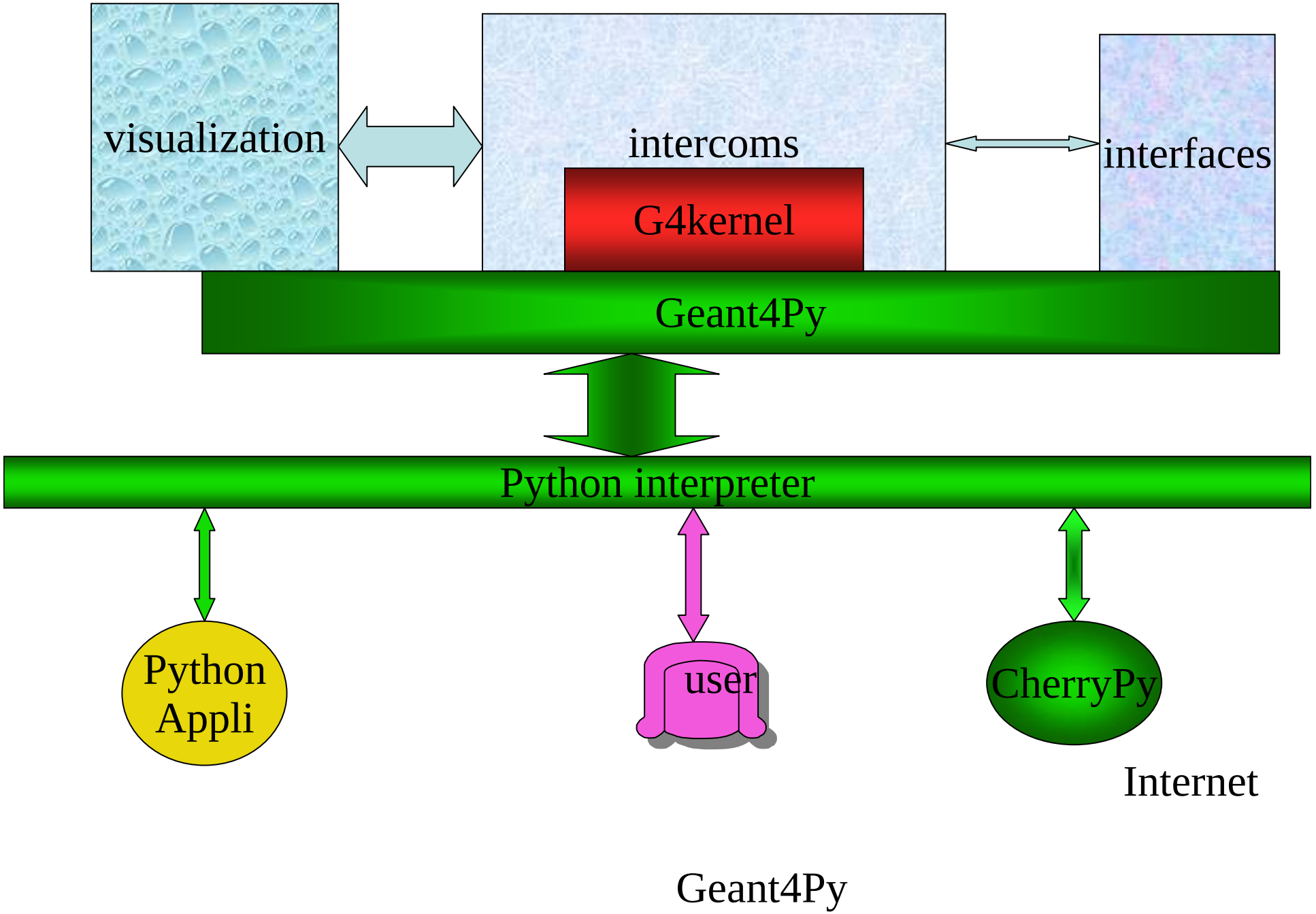
Investigation of Python interface

- Report of 2004 Workshop -

- In 2003, we planned to investigate the possibility of new interactivity which might be introduced by Python wrappers
  - User requirements and user's implementation
    - Space science etc. (with the Gaudi framework)
  - choice of Python wrapper ; uncertainty of the de fact standards
    - Which wrapper, **Boost**, or Swig?
  - Scope and depth of wrapper ; G4 interactivity
    - Which Geant4 **classes are to be wrapped**
  - Talk by Koichi

Report of 2005 Workshop

1. Migration to <sstream>
2. Geant4Py; Pythonization
   1. Using Boost-Python
   2. Important amount of classes are wrapped, taking care of the security issues
3. Consequences of Pythonization;
   1. Geant4CherryPy, a Geant4 Web Service
   2. GUI using Tkinter, the default Python GUI based on Tk
4. Report of the “Geant4 for Education Workshop” at Naruto UE
   1. Geant4Py as a toolkit for courseware makers


visualization
intercoms
G4kernel
interfaces
Geant4Py
Python interpreter
Python
Appli
user
CherryPy
Internet

Geant4Py

Status of 2005-2006

- Geant4Py included in the release 8.1
- Various applications are implemented and pythonic toolkits are tested
  - Plotting cross sections
  - Histogramming toolkits
  - Web server
  - New GUI, using wxPython, the wrapped wxwidgets by Python
  - Examples

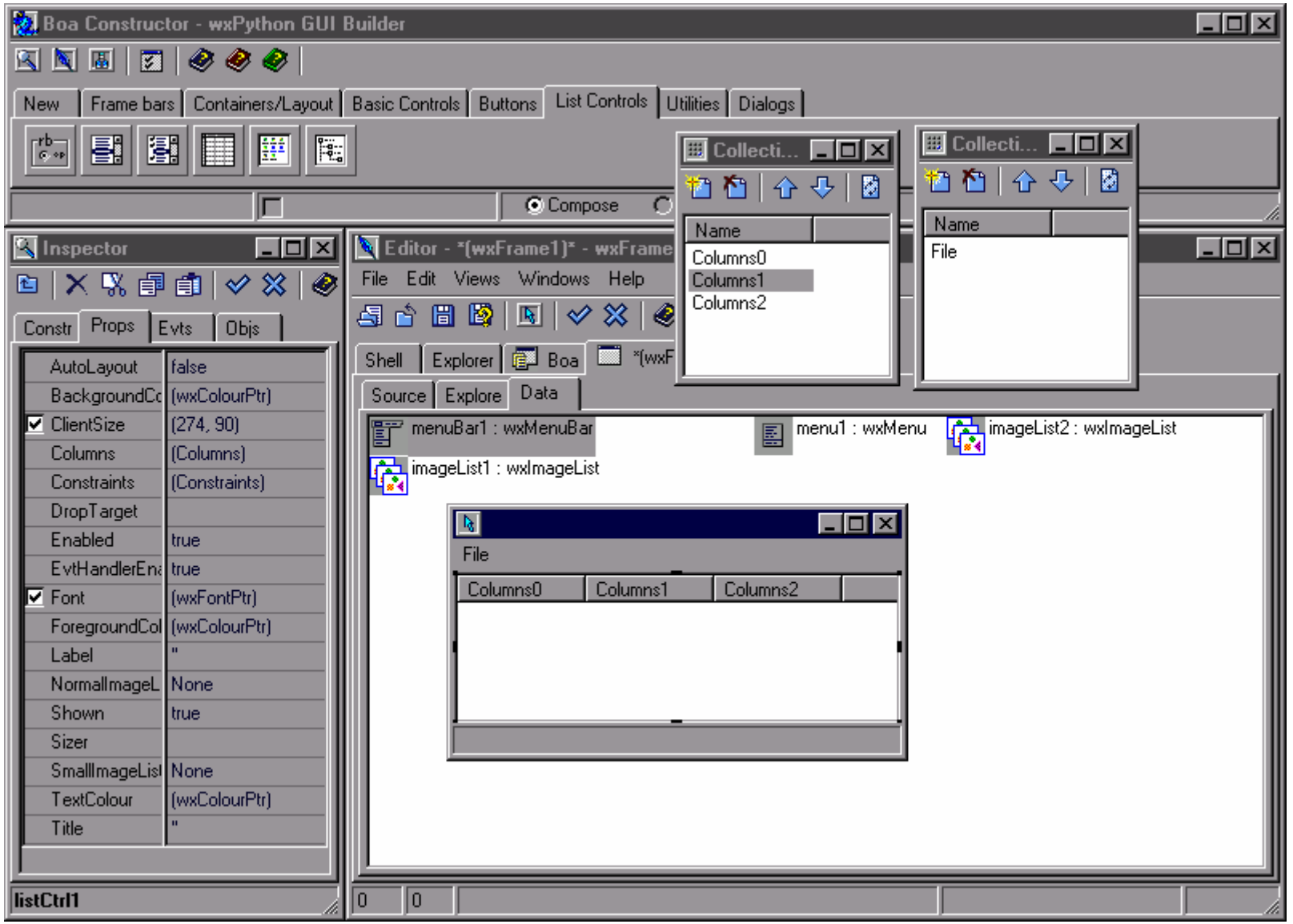
Example of wrapping C++ classes

- Tests/gtest02
  - ExN03DetectorConstruction is wrapped
    - All of its methods are exposed
  - ExN03PhysicsList is wrapped
    - All of its methods are exposed
    - SetProcessActivation method of the G4ProcessTable is exposed and used
- examples/demo/water_phantom
  - Voxelized geometry implemented with C++
  - Sensitive volume implemented with Python
  - Connection with ROOT
  - Visualization with OpenGL

GUI on the top of Geant4Py

- GUIs may depend on specific applications
- Presenting some examples will be useful for users to develop their own GUI
- wxPython is the GUI toolkit of our choice
  - Wxwidgwts(C++) classes are wrapped by Python using SWIG
  - Cross platform GUI library
  - Robust and highly functional
  - http://wxpython.org
  - Rich online documentations
- **What they say on wxPython**
  - **wxPython is the best and most mature cross-platform GUI toolkit, given a number of constraints. The only reason wxPython isn't the standard Python GUI toolkit is that Tkinter was there first. -- Guido van Rossum**
- Active user communities: http://wiki.wxpython.org/
- Many tools like boa-constructor(IDE) etc.

boa_constructor; IDE

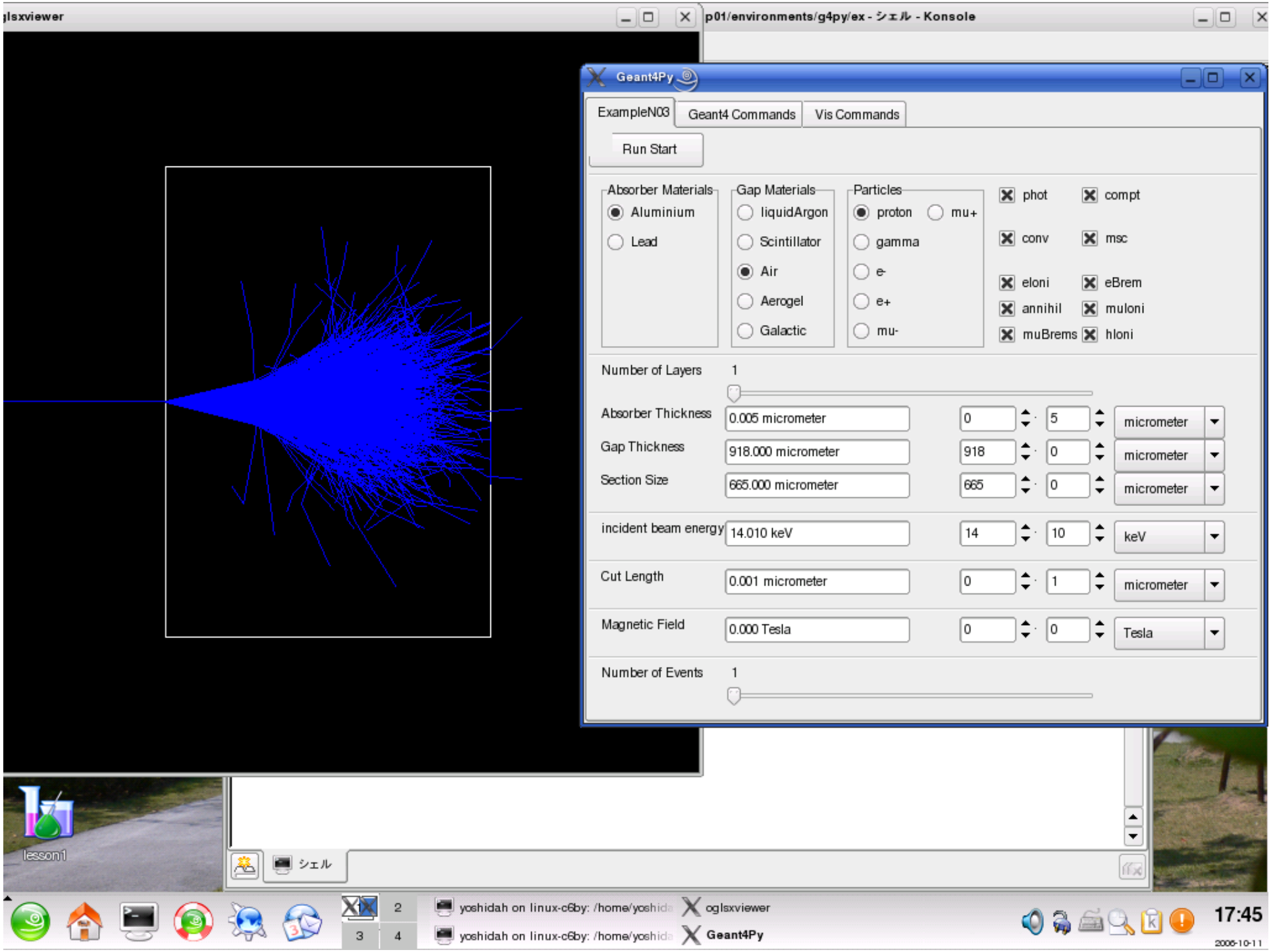


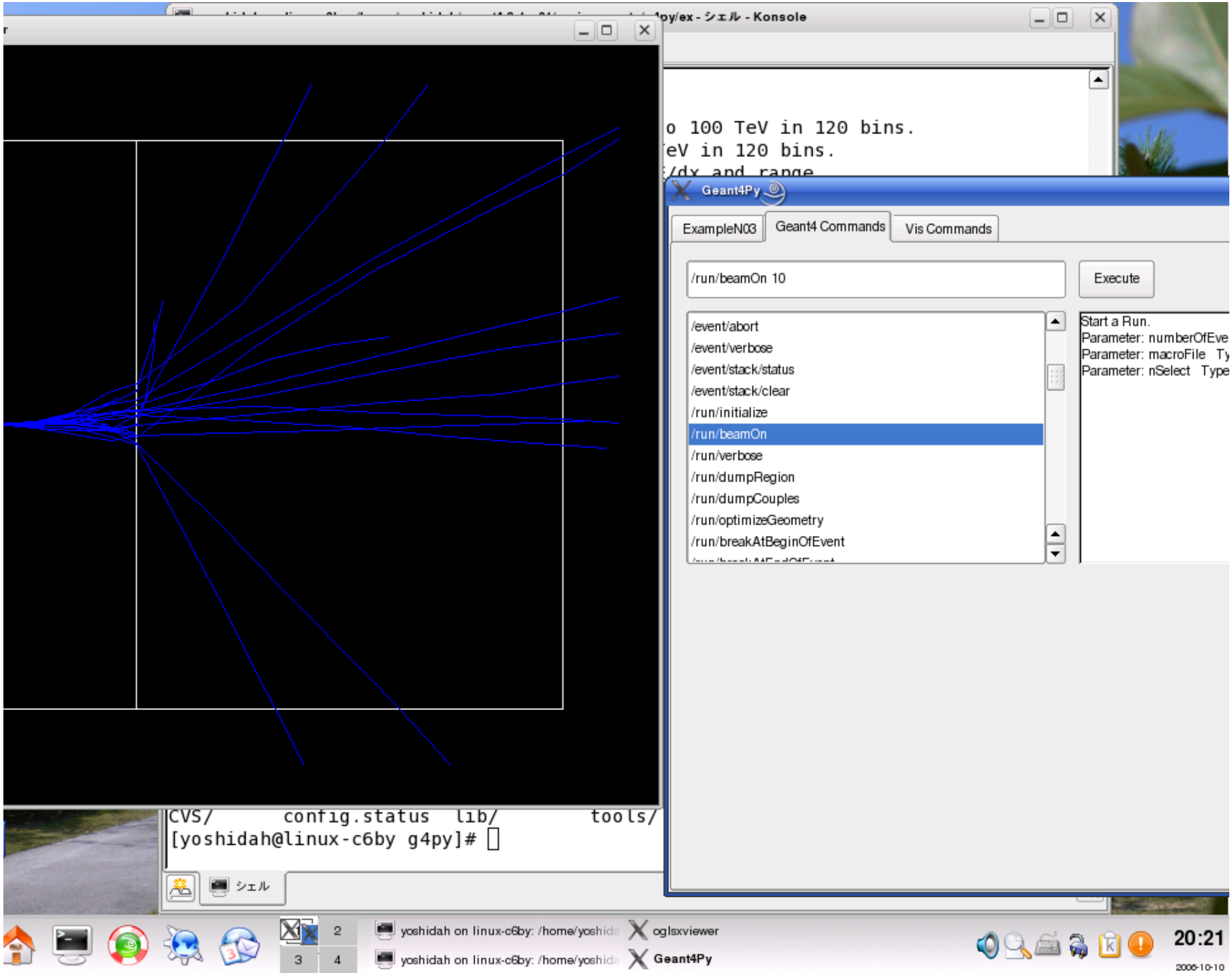
Extending interactivity

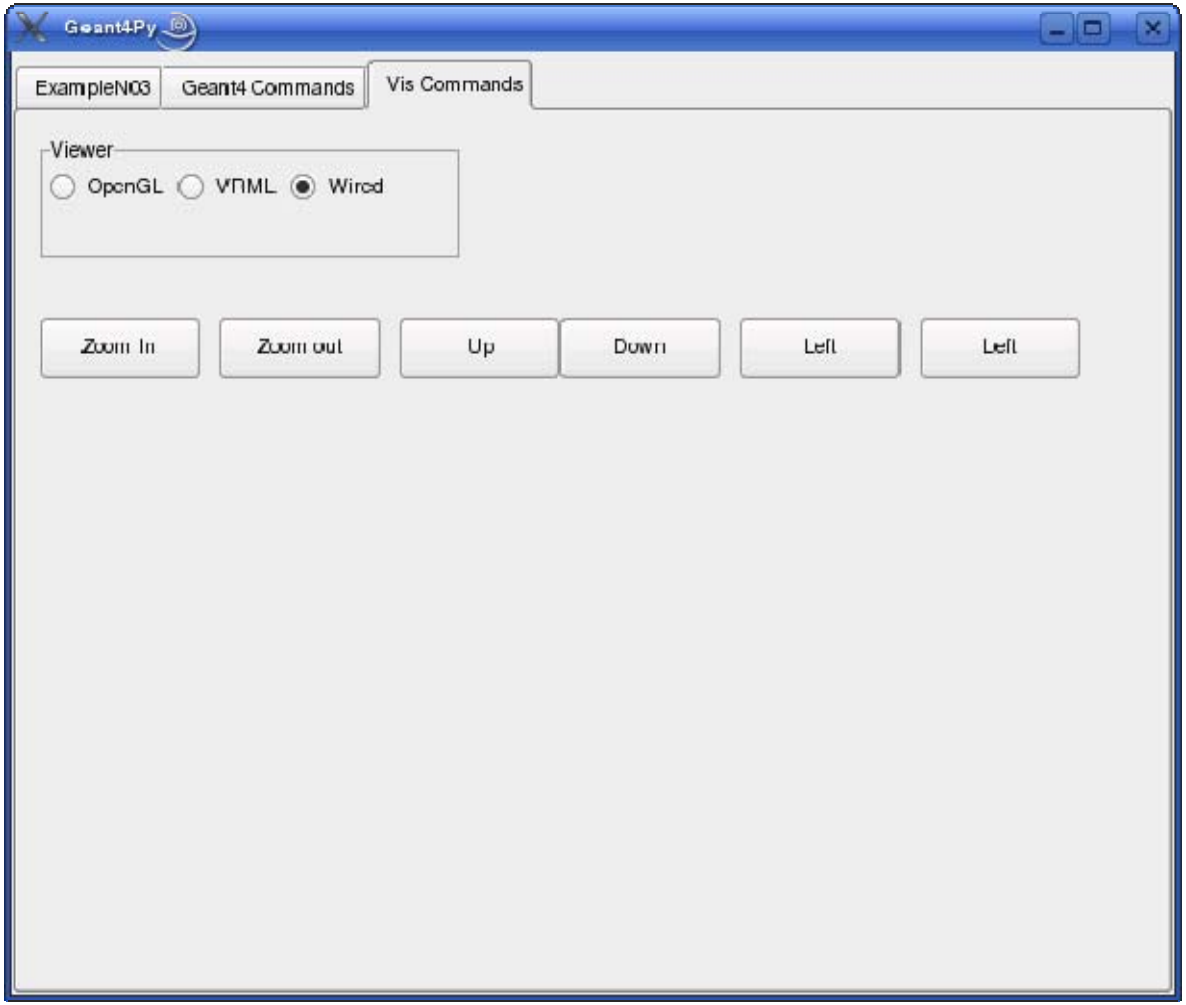
- Plot tools : matplotlib, plot library a la Matlab
  - Embeddable in wxPython
- Web server : CherryPy, purely Pythonic Web server
  - Powerful template language supported
  - Session and cookie management etc.
- Applications
  - **Geant4 for Education project, combining the above user friendly environment**
  - **SIG: Python for Education**

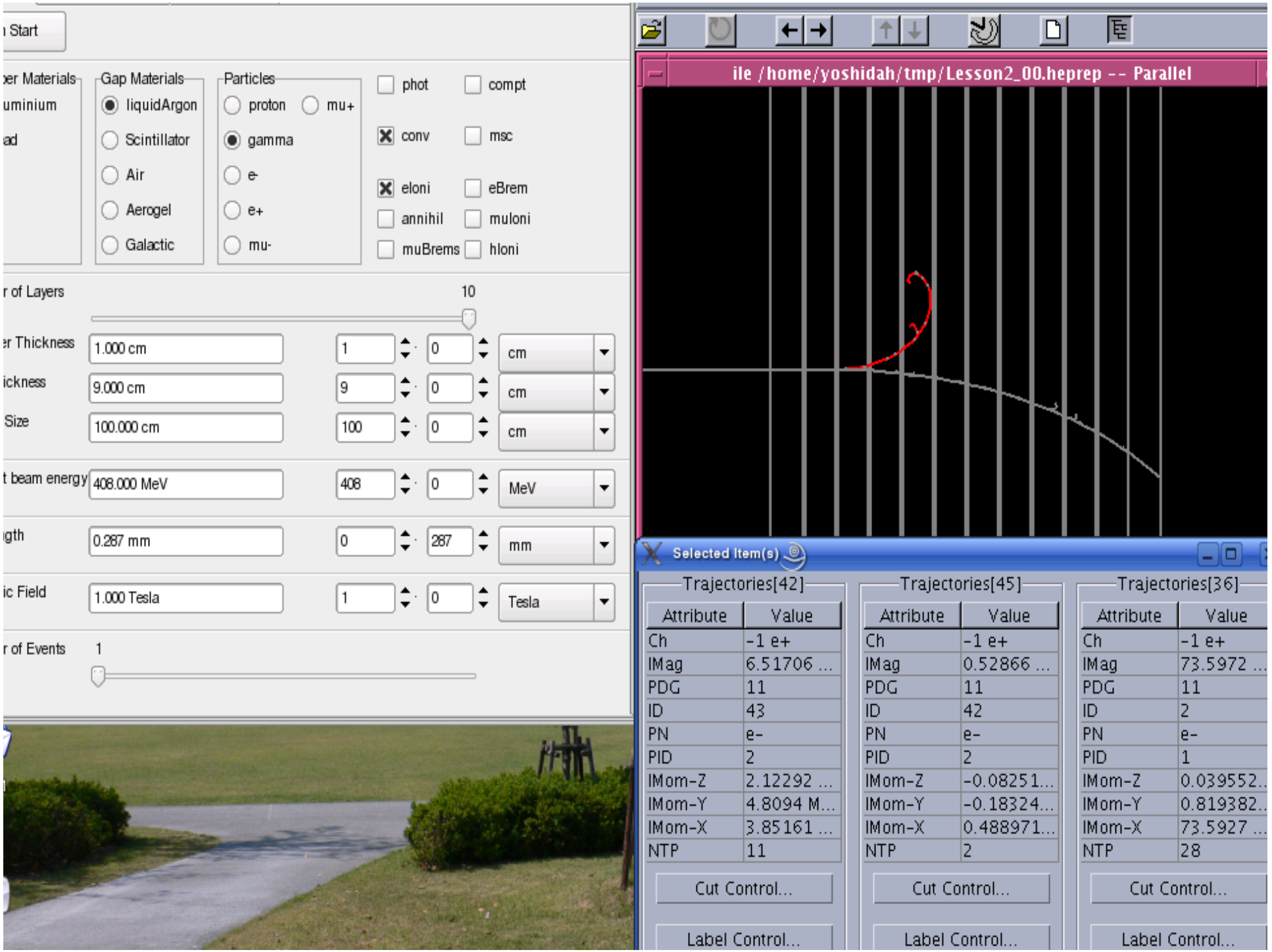
Examples

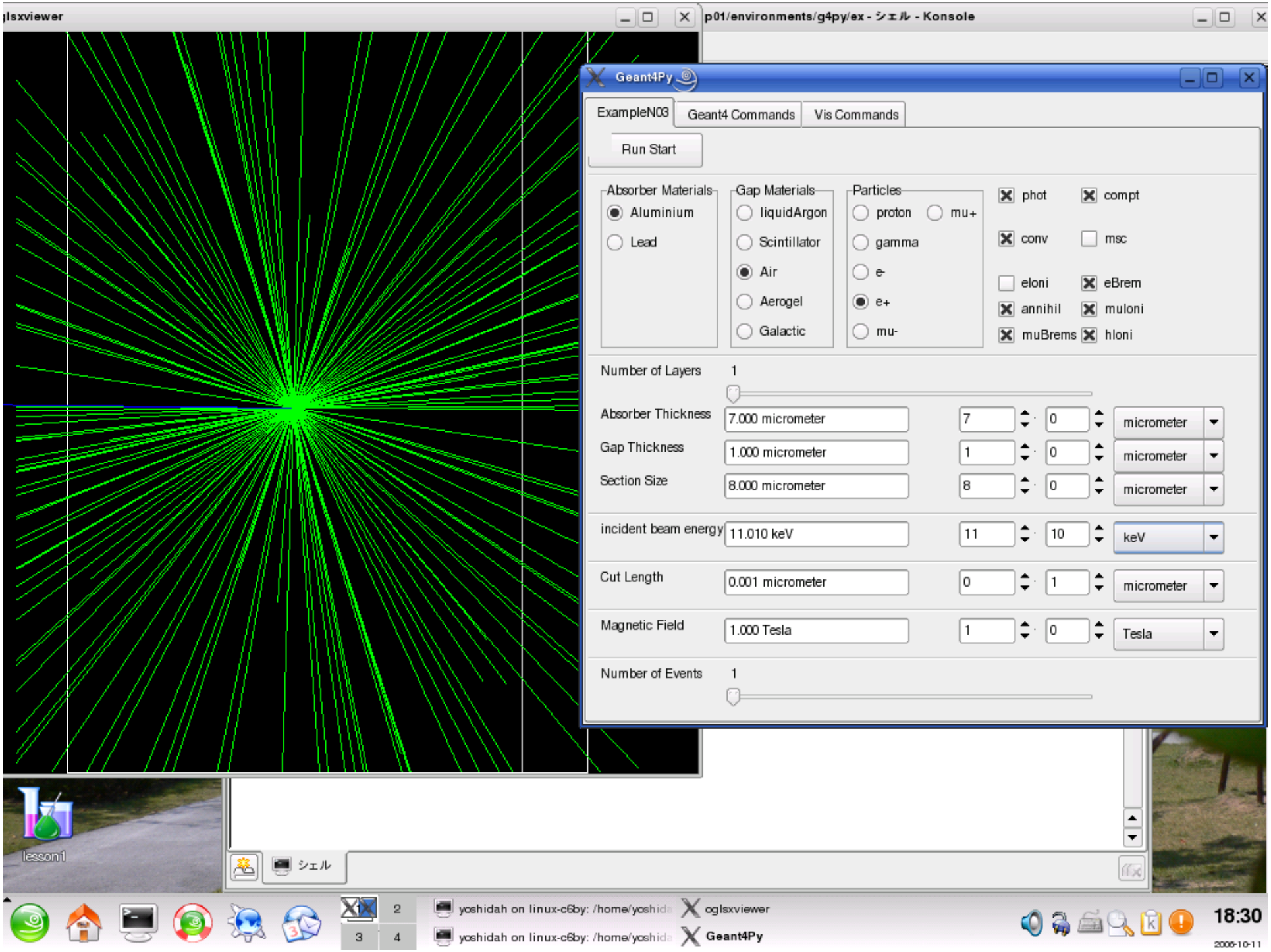
- Educational examples with wxPython GUI
  - Lesson1 : purely Pythonic script
    - measurement of mass attenuation coefficients in various materials with variable dimensions
    - And other observations
  - Lesson2 : wrapped C++ classes of exampleN03
    - sandwich calorimeter geometry is modifiable with GUI
    - electromagnetic processes can be switched on/off with GUI
- Visualization tools can be switched easily from one to another: OpenGL, VRML or Wired
- Preliminary implementation of GAG

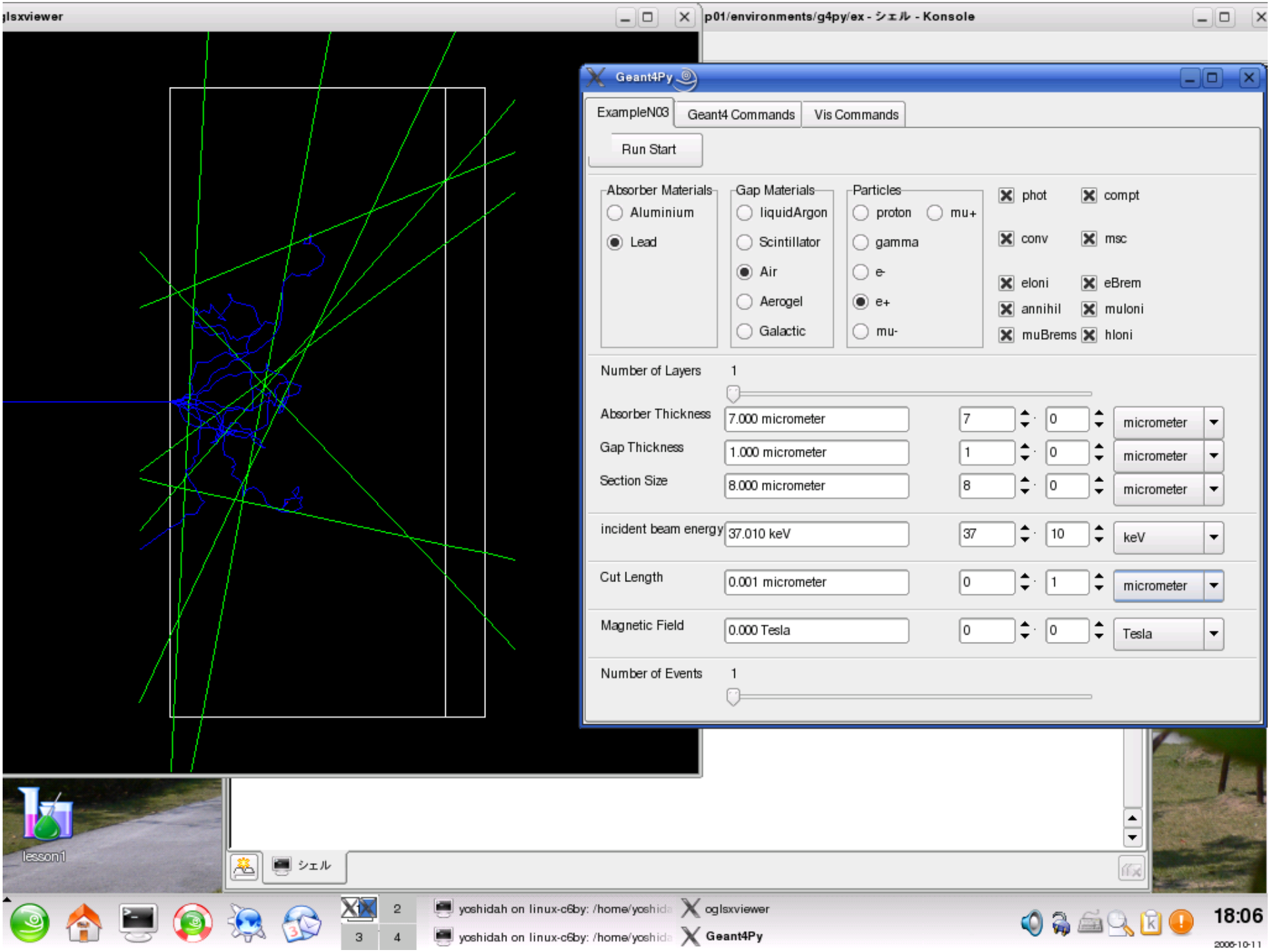

glsxviewer
p01/environments/g4py/ex - シェル - Konsole
Geant4Py
ExampleN03
Geant4 Commands
Vis Commands
Run Start
Absorber Materials
Aluminium
Lead
Gap Materials
liquidArgon
Scintillator
Air
Aerogel
Galactic
Particles
proton
mu+
gamma
e-
e+
mu-
phot
compt
conv
msc
eloni
eBrem
annihil
muloni
muBrems
hloni
Number of Layers
1
Absorber Thickness
0.005 micrometer
0
5
micrometer
Gap Thickness
918.000 micrometer
918
0
micrometer
Section Size
665.000 micrometer
665
0
micrometer
incident beam energy
14.010 keV
14
10
keV
Cut Length
0.001 micrometer
0
1
micrometer
Magnetic Field
0.000 Tesla
0
0
Tesla
Number of Events
1
lesson1
シェル
yoshidah on linux-c6by: /home/yoshida
oglsxviewer
yoshidah on linux-c6by: /home/yoshida
Geant4Py
17:45
2006-10-11


py/ex - シェル - Konsole
o 100 TeV in 120 bins.
eV in 120 bins.
/dx and range
Geant4Py
ExampleN03
Geant4 Commands
Vis Commands
/run/beamOn 10
Execute
/event/abort
/event/verbose
/event/stack/status
/event/stack/clear
/run/initialize
/run/beamOn
/run/verbose
/run/dumpRegion
/run/dumpCouples
/run/optimizeGeometry
/run/breakAtBeginOfEvent
Start a Run.
Parameter: numberOfEve
Parameter: macroFile Ty
Parameter: nSelect Type
CVS/ config.status lib/ tools/
[yoshidah@linux-c6by g4py]#
シェル
yoshidah on linux-c6by: /home/yoshid
oglsxviewer
yoshidah on linux-c6by: /home/yoshid
Geant4Py
20:21
2006-10-10


Geant4Py
ExampleN03
Geant4 Commands
Vis Commands
Viewer
OpenGL
VRML
Wired
Zoom In
Zoom out
Up
Down
Left
Left


Start
Gap Materials
liquidArgon
Scintillator
Air
Aerogel
Galactic
Particles
proton
mu+
gamma
e-
e+
mu-
phot
compt
conv
msc
eloni
eBrem
annihil
muloni
muBrems
hloni
of Layers
10
Thickness
1.000 cm
Size
100.000 cm
beam energy
408.000 MeV
0.287 mm
Field
1.000 Tesla
of Events
1
cm
MeV
mm
Tesla
ile /home/yoshidah/tmp/Lesson2_00.heprep -- Parallel
Selected Item(s)
Trajectories[42]
Trajectories[45]
Trajectories[36]
Attribute
Value
Ch
-1 e+
PDG
11
ID
43
42
2
1
PN
e-
PID
IMom-Z
IMom-Y
IMom-X
NTP
28
Cut Control...
Label Control...


glsxviewer
p01/environments/g4py/ex - シェル - Konsole
Geant4Py
ExampleN03
Geant4 Commands
Vis Commands
Run Start
Absorber Materials
Aluminium
Lead
Gap Materials
liquidArgon
Scintillator
Air
Aerogel
Galactic
Particles
proton
mu+
gamma
e-
e+
mu-
phot
compt
conv
msc
eloni
eBrem
annihil
muloni
muBrems
hloni
Number of Layers
1
Absorber Thickness
7.000 micrometer
Gap Thickness
1.000 micrometer
Section Size
8.000 micrometer
incident beam energy
11.010 keV
Cut Length
0.001 micrometer
Magnetic Field
1.000 Tesla
Number of Events
1
micrometer
keV
Tesla
lesson1
シェル
yoshidah on linux-c6by: /home/yoshida
oglsxviewer
Geant4Py
18:30


glsxviewer
p01/environments/g4py/ex - シェル - Konsole
Geant4Py
ExampleN03
Geant4 Commands
Vis Commands
Run Start
Absorber Materials
Aluminium
Lead
Gap Materials
liquidArgon
Scintillator
Air
Aerogel
Galactic
Particles
proton
mu+
gamma
e-
e+
mu-
phot
compt
conv
msc
eloni
eBrem
annihil
muloni
muBrems
hloni
Number of Layers
1
Absorber Thickness
7.000 micrometer
Gap Thickness
1.000 micrometer
Section Size
8.000 micrometer
incident beam energy
37.010 keV
Cut Length
0.001 micrometer
Magnetic Field
0.000 Tesla
Number of Events
1
micrometer
keV
Tesla
lesson1
シェル
yoshidah on linux-c6by: /home/yoshida
oglsxviewer
Geant4Py
18:06

Plans

- Geant4Py
  - Windows
- GUI with wxPython
  - Prepare Python classes for Geant4 oriented widgets kits
    - Choice of particles, materials, processes etc.
    - Adjustment of energy, length, directions etc.
    - Visualization commands (frequently used ones)?
    - And others
- Integration of plotting toolkit
  - Matplotlib is wxPython embeddable and is the first target http://matplotlib.sourceforge.net/
  - Gnuplot wrapped by Python http://gnuplot-py.sourceforge.net/
- Connection with Geant4 Users Manuals
- More examples of different use cases