

Applications of Geant4 for Education

*The Japan Taiwan Symposium on
Simulation in Medicine
Tsukuba, Japan*

14 December 2006

*Hajime Yoshida
Naruto University of Education, JST*

Contents

- Motivation
 - Geant4's aspects for education;
 - Not to learn Geant4 but to use Geant4 to learn
 - Workshop in 2005
- Educational examples in other fields
- Geant4's tool kits for educational applications
 - Geant4Py; Python wrapper of Geant4
 - WxPython; Python wrapper of wxWidgets
 - Visualization and analysis tools on the Python s/w bus
 - Web service
- Examples
- TODO : Courseware

Questions

- This topic isn't the main stream of simulation applications but will contribute for better understanding of the microscopic phenomena and improvement of interdisciplinary understanding of macroscopic measurements
- Which level?
 - University, and if possible Professionals
- Which Learning style?
 - Lab works, Hands-on, projects in one CD
 - Distant learning
- How can we create materials?
- Who can participate for creation? What knowledge is required?

Educational uses of Partial Differential Equation-based simulations

- Create application oriented models to fill the gap between the PDEs and phenomena and to understand them more deeply
- Reports from Comsol MultiPhysics Conference 2005
 - To investigate the basic laws of physics: Tuebingen: Theoretische Physik
 - To teach Biomechanics and Biomedical Physics: Uppsala, Molecular Biotechnology
 - To Teach and Discover Transport Phenomena: Rensselaer Polytechnic, Chemical and Biological engineering
 - To Teach Chemical Engineering: Worcester Polytechnic
 - For Bioengineering Education: Penn State Univ. Bioengineering

Teach Biomechanics and Biomedical physics through models

•J. Gantelius, Uppsala Univ.

- Blood flow and Gas exchange in an Alveolus
- Model the treatment of a liver cancer tumour through resistive heating of an inserted electric probe
- Structural-Fluid interaction in a Network of Blood Vessels
 - Convey a more practical feeling of the meaning of the equations
 - Real examples, solve real problems
 - Apply theoretical knowledge to numerical methods

•P. J. Butler and M. C. Ferko of Penn SU

- 3-D vascular graft simulation: structural fluid interaction
 - Poiseuille flow of blood and its interaction with the elastic wall
 - Students gained new insight that was otherwise unobtainable either by experiments or by conventional analytical models.

Geant4's educational aspects

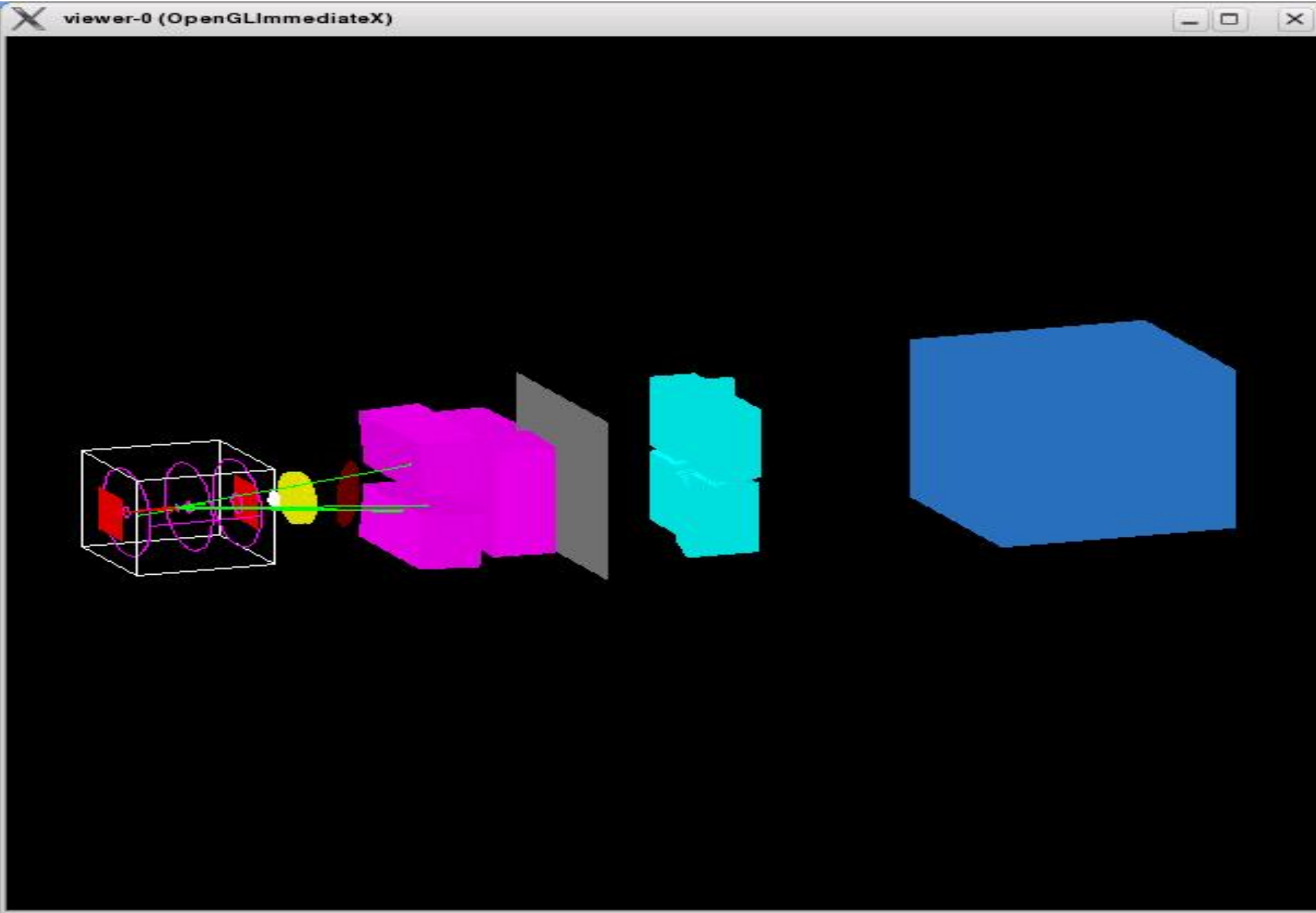
- Geant4 is based on a set of phenomenological knowledge, each of them having its own limitation of applicability, and can be used to learn the real life of various fields; physics, space, medical etc..
- Geant4 is capable of handling complex geometries encountered in the real life of various fields
- We expect that the educational materials will fill the gap between microscopic phenomena and the macroscopic quantities or observables.
- Geant4 public distribution contains many realistic examples and test suites
 - Course materials for standard electromagnetic physics by M. Maire
 - Advanced examples coordinated by M. g. Pia
 - And others
- Geant4 provides full set of toolkits i.e. GUI, visualization, interfaces to analysis tools etc. which are the key elements for creation of good educational contents.

Geant4 medical examples

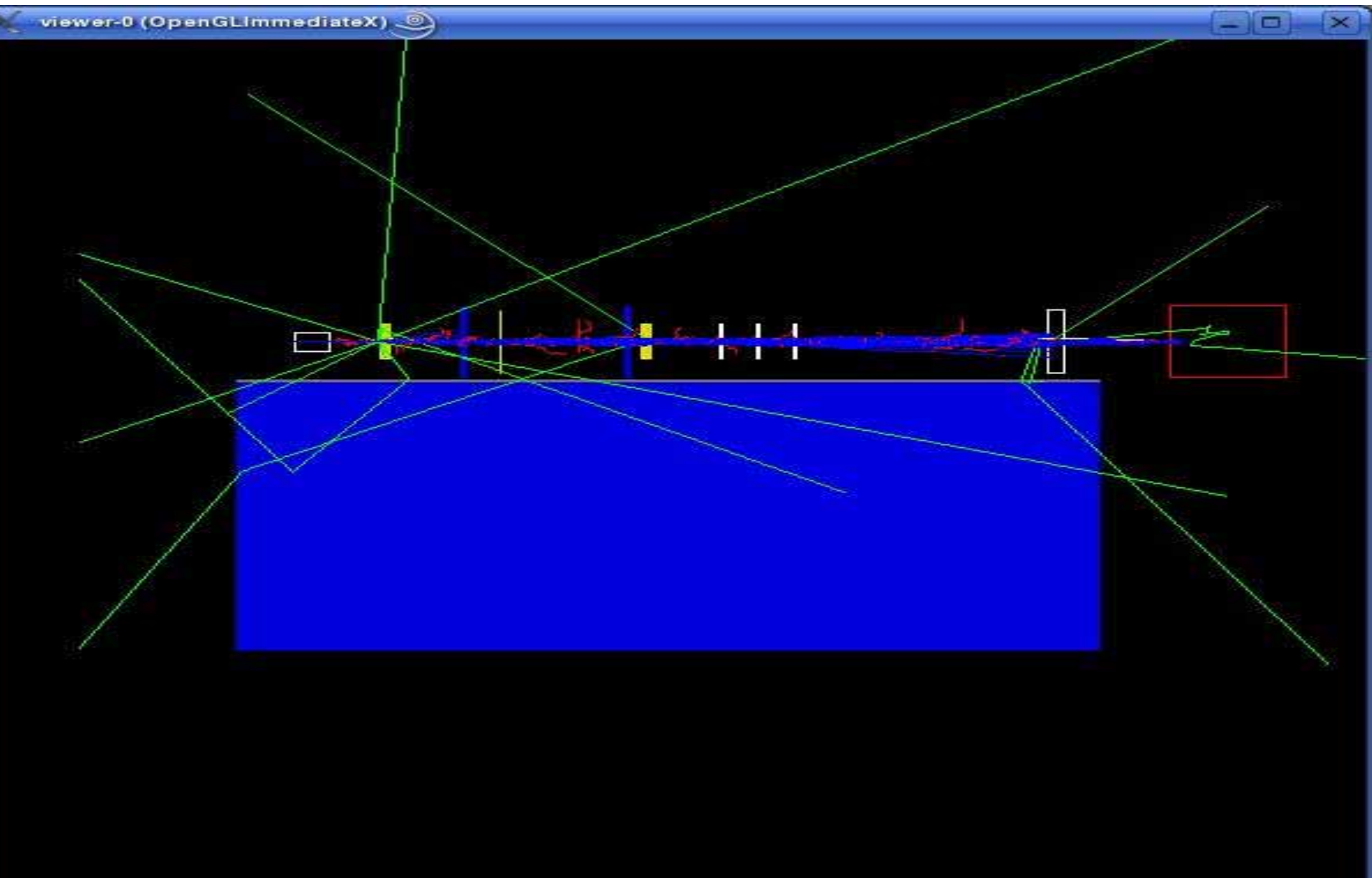
- Medical_linac by M. Piergentile
 - IMRT
 - voxelized water phantom
- Brachytherapy by S. Guatelli

- Hadrontherapy by G.A.P. Cirrone et al.
 - Beam line geometry of LNS-INFN Catania
- DICOM by L. Archambault et al.
 - Voxelized DICOM geometry
- Gammatherapy by
 - 50 MeV electron beam line
- And others
- These are the raw stuffs for educational purposes.

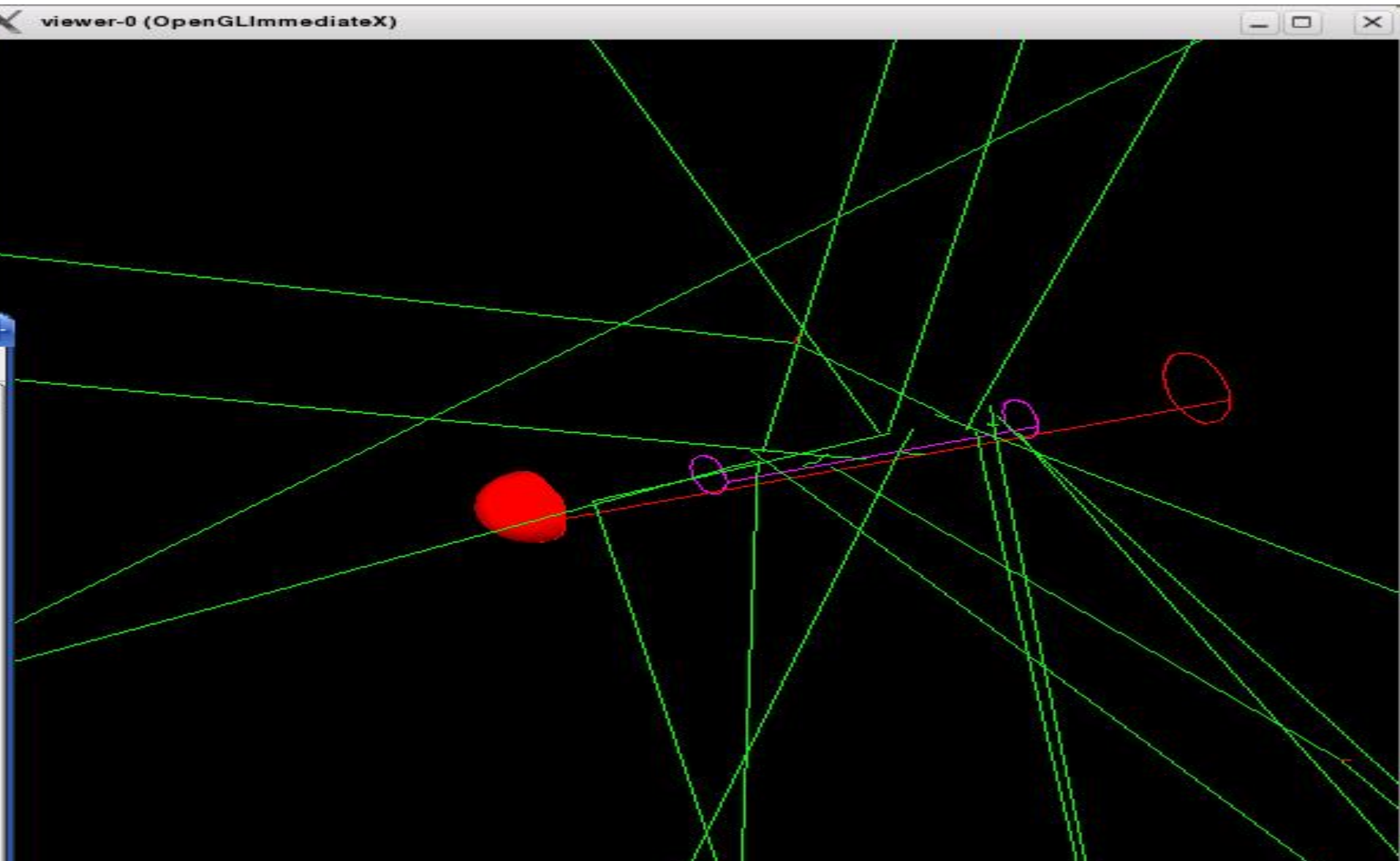
Medical Linac



Hadrontherapy



Brachytherapy



Geant4Py Tool kits for Educational Applications

- We anticipate **two** user categories;
 - Contents Creators (*teachers, or professionals*)
 - End Users (*students and eventually professionals under distant learning*)
- **Geant4Py Tool kits For Contents Creators**
 - Developed by K. Murakami, now available in geant4-8.1/environments/g4py
 - Python's powerful scripting capabilities are exploitable
 - Python interface can work as component bus.
 - Modules are available
 - Material / Geometry (predefined geometry / easy geometry set-up)
 - Physics list (EM, Hadrons, Ion)
 - Detector response (Calorimeter / Tracker)
 - Analysis packages (ROOT, HBOOK, AIDA, ...)
 - Visualization
 - GUI (Qt, Tkinter, wxPython...) / Web applications (mod-python, CherryPy)
- **Course ware For End Users**
 - Scripting with Python is **NOT** required!
 - They are not necessarily required to learn Python language.
 - Of course, they can modify the course materials with the knowledge of Python and can contribute for their improvements.
 - **GUI / Web applications should be provided for e-learning**
 - They can be built on the Python interface.

Steering Geant4 Applications with Python

- Characteristics of Geant4 Simulation
 - Big software company structure, complex
 - Long life evolving
 - Unexpected use cases
- Paper of Boost.python
 - Existing C++ libraries are wrapable by end users who only have access to header files and binaries.
- Steering with Python
 - Use of the interpreting language to interface the compiled libraries
 - No "main" program. Control of simulation by Python

Use of Geant4Py

case 1 : wrapping C++ codes

- Create an application in C++ and wrap its classes as necessary
 - Examples are found in `g4py/site-modules/`
 - `examples/education/lesson2`
 - Performance isn't deteriorated,
 - Much more interactive than the terminal interfaces
 - Python based GUI tool kits can be employed for better user friendliness
 - Connection with analysis tools is easy
 - Integration into the Web server

Predefined Packages

- Site-module package contains pre-defined components.
 - Material
 - sets of pre-defined materials
 - NIST materials via *G4NistManager*
 - Geometry
 - “exN03” geometry as an example of pre-defined geometries
 - “EZgeometry”
 - provides functionalities for easy geometry set-up (applicable to target experiments)
 - Physics List
 - pre-defined physics lists, exN03 etc.
 - easy access to cross sections, stopping powers, ... via *G4EmCalculator*
 - Primary Generator Action
 - particle gun
 - Sensitive Detector
 - calorimeter type
 - tracker type
- They can be used just by importing modules.
- They can be combined and connected to higher application layers (Analysis / GUI components).

Use of Geant4Py

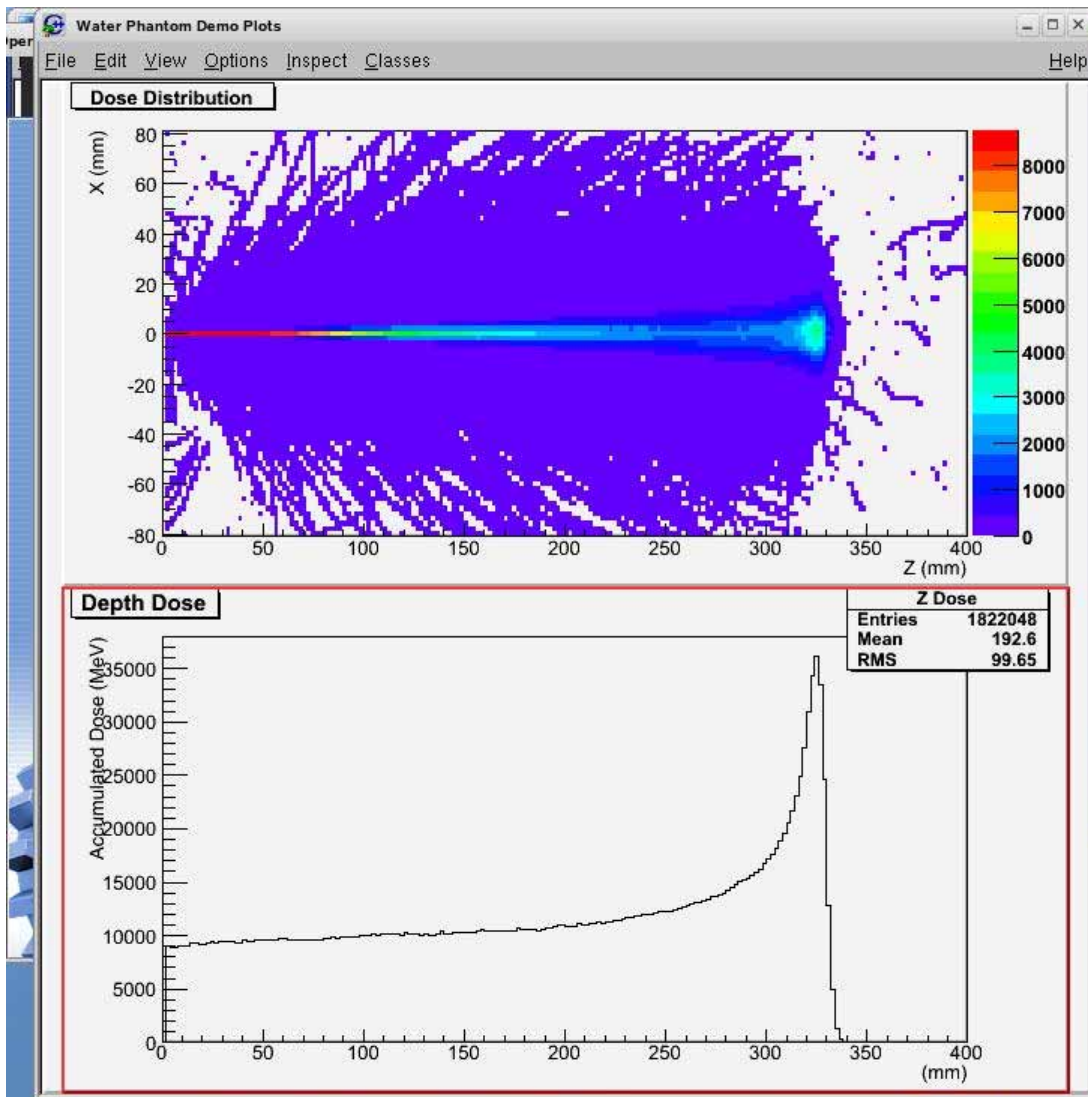
case 2 : purely Pythonic scripting

- Pre-defined “site-modules” provide easy construction of “simple” geometries
 - No C++ coding is necessary to create your own detector, beam line etc.. Python script can do all.
 - Typical e.m. Physics list is provided. Importing them is sufficient to use them in your Python script
 - Performance isn't so bad
- Integration with analysis tools and use of fancy GUI tools are just same as the case 1
- [examples/education/lesson1](#)

Extending its connectivity

- Plot tools : matplotlib, GNU PLOT
- Analysis tools : ROOT, PAIDA
- Web server : CherryPy
 - Purely Pythonic Web server
 - Powerful template language supported
 - Session and cookie management etc.
- GUI tool kit : Tkinter, wxPython
 - Dedicated for each example
 - Replacing the old GUI tools of Geant4
- Geant4 for Education project is the typical case which requires all these functionalities

Connect to ROOT histogramming on the fly



```
yoshidah on linux: /home/yoshidah/g4py/examples/demos/water_phantom - シェル - Konsole
セッション 編集 表示 ブックマーク 設定 ヘルプ

w, Integral mode 1

msc: Model variant of multiple scattering for proton
Lambda tables from 100 eV to 100 TeV in 120 bins.
Boundary algorithm is active with facrange= 0.199

msc: Model variant of multiple scattering for pi-
Lambda tables from 100 eV to 100 TeV in 120 bins.
Boundary algorithm is active with facrange= 0.199

*** End of Run
- Run summary : (id= 0, #events= 100)

In [2]: de
def del delattr demo_wp

In [2]: gRunManager.BeamOn(100)
*** End of Run
- Run summary : (id= 1, #events= 100)

In [3]: gRunManager.BeamOn(10000)
*** End of Run
- Run summary : (id= 2, #events= 10000)

In [4]: █

シェル
```

Geant4 Web Server steering, visualization and analysis

The screenshot displays a web browser window titled "Geant4 Web Services" and a "Water Phantom Demo Plots" window. The browser window shows a sidebar with navigation links: [About](#), [Choose an Exp](#), [Exposed Classes](#), [VRML](#), [DAWN](#), [BeamOn](#), [ExecCom](#), and [ROOT](#). The main content area features the heading "Geant4 Web Services" and "Python Geant4 Command?". Below this is a text input field containing "ApplyU" and a button labeled "提出". At the bottom, it displays "Build time: 0.028s," "Page size: 0.25KB".

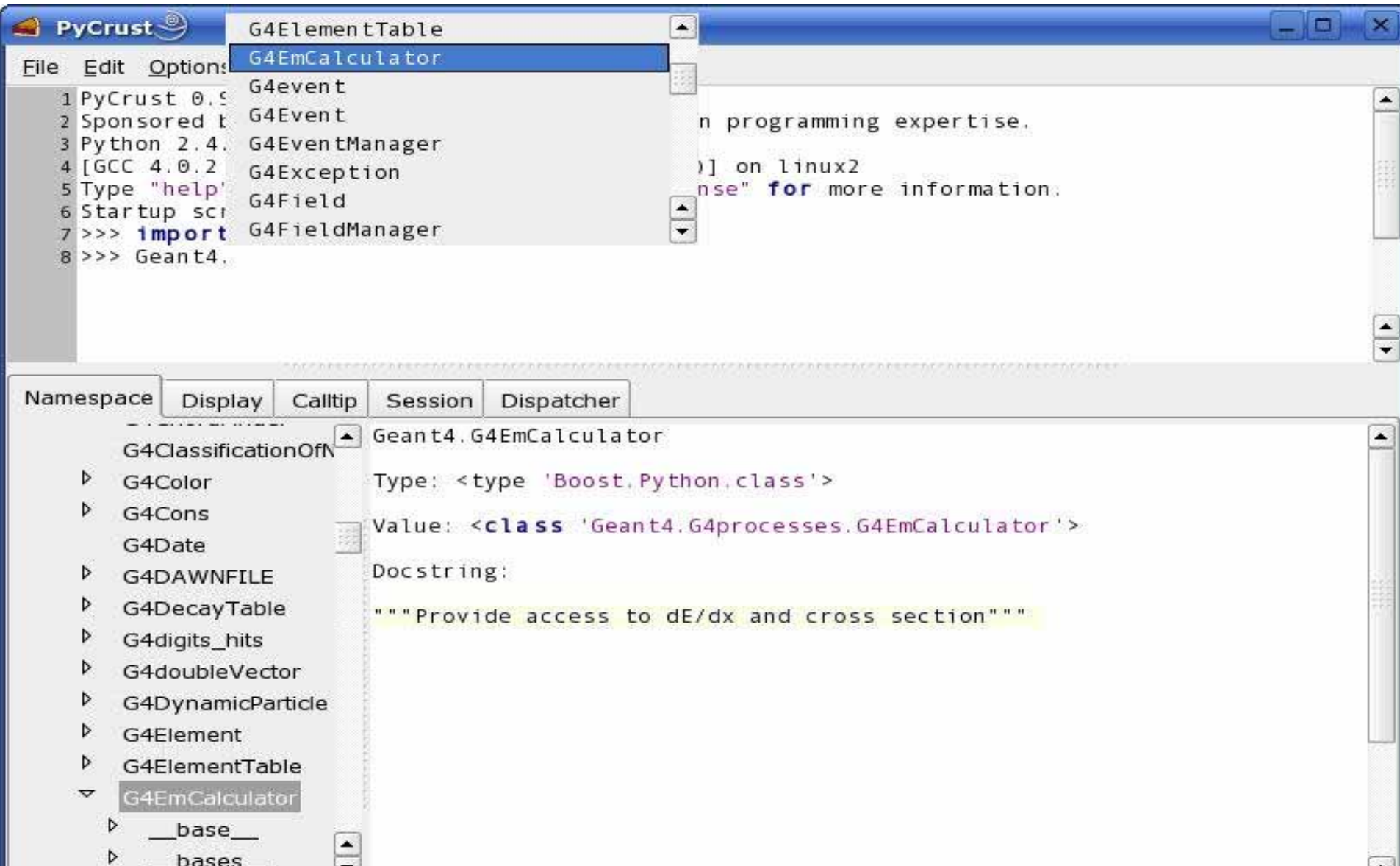
The "Water Phantom Demo Plots" window contains two plots. The top plot, titled "Dose Distribution", is a 2D heatmap showing dose distribution in a water phantom. The vertical axis is labeled "mm" with a value of 80. The horizontal axis is labeled "Z (mm)" with values from 50 to 400. A color scale on the right ranges from 0 (blue) to 70 (red). The bottom plot, titled "Z Dose", is a line graph showing the dose profile along the Z-axis. The vertical axis is labeled "Dose" and the horizontal axis is labeled "Z (mm)" with values from 50 to 400. A statistics table in the top right corner of the Z Dose plot provides the following data:

Z Dose	
Entries	64940
Mean	182.9
RMS	100.9

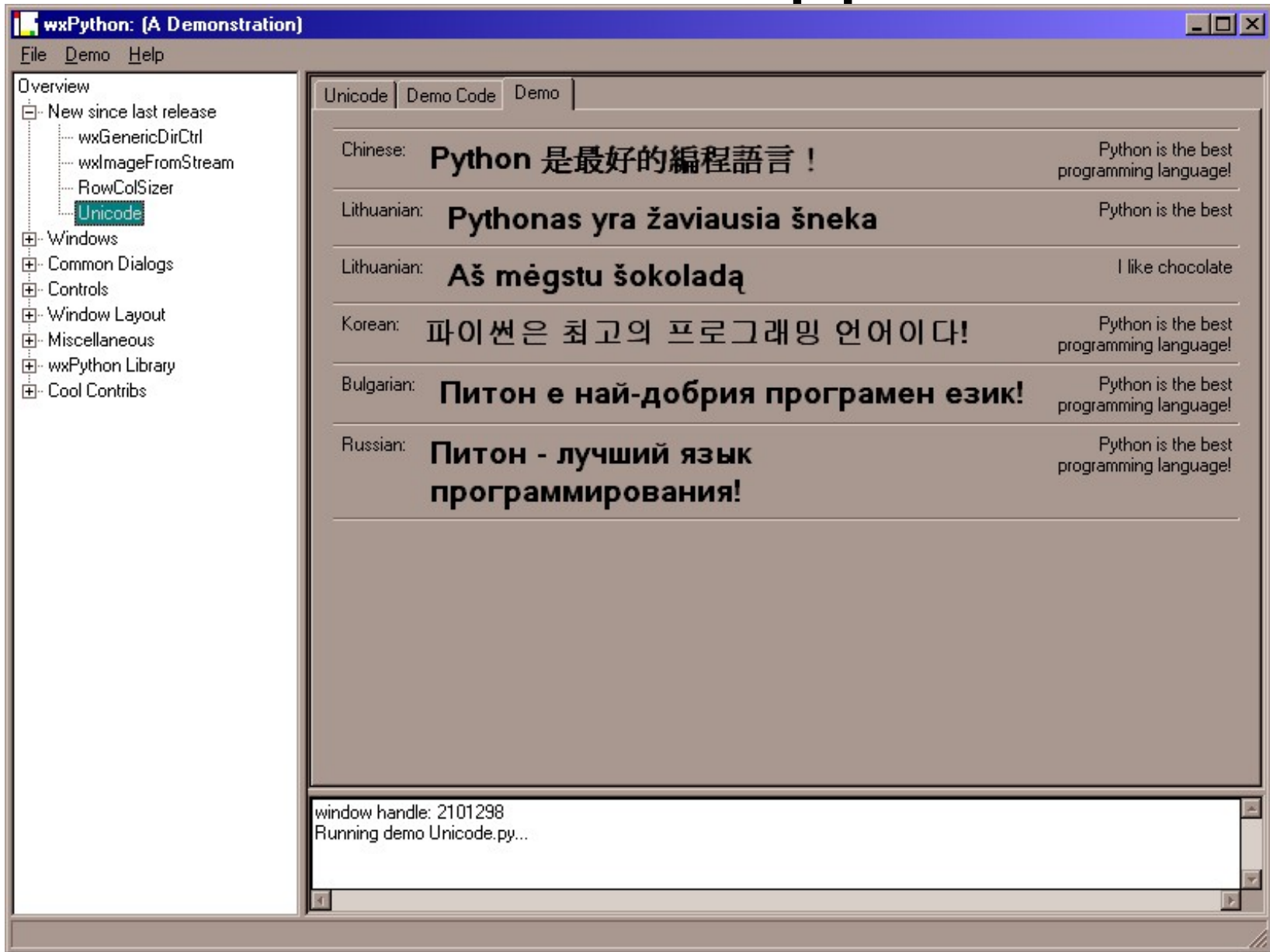
GUI builder is mandatory

- WxPython is our choice
 - Wxwidgets (C++ library) is wrapped with Python
 - Open and free
 - Many advanced features and widgets
 - A little lengthy scripting (than Tkinter) to profit its power
 - Multi platform, keeping platforms look and feel
 - Book has come

IDEs are available



Unicode is supported



Educational examples

- Virtual experiments to understand the microscopic phenomena, included since geant4-version 8.1

- lesson1

- measurement of mass attenuation coefficients in various materials with variable dimensions
 - And other observations

- lesson2

- taken from Michel Maire's exampleN03
 - sandwich calorimeter
 - electromagnetic processes on/off

- Platforms

- Linux and Mac OS X are tested

14keV proton into air

The image shows a screenshot of a Linux desktop environment with a Geant4Py simulation window. The main window displays a visualization of a proton track starting from a point on the left and spreading out as it moves to the right, represented by a dense cloud of blue lines. The track starts as a single line and branches into many lines, indicating particle interactions. The background is black, and the track is bright blue.

The Geant4Py window is titled "Geant4Py" and has two tabs: "ExampleN03" and "Geant4 Commands". Below the tabs is a "Run Start" button. The configuration panel is divided into several sections:

- Absorber Materials:** Aluminium, Lead
- Gap Materials:** liquidArgon, Scintillator, Air, Aerogel, Galactic
- Particles:** proton, mu+, gamma, e-, e+, mu-
- Physics Processes:** phot, compt, conv, msc, eloni, eBrem, annihil, muloni, muBrems, hloni
- Number of Layers:** 1
- Absorber Thickness:** 0.005 micrometer (range 0 to 5)
- Gap Thickness:** 918.000 micrometer (range 918 to 0)
- Section Size:** 665.000 micrometer (range 665 to 0)
- incident beam energy:** 14.010 keV (range 14 to 10)
- Cut Length:** 0.001 micrometer (range 0 to 1)
- Magnetic Field:** 0.000 Tesla (range 0 to 0)
- Number of Events:** 1

The desktop environment includes a taskbar at the bottom with icons for "Lesson1", "シェル", and "oglsviewer". The system tray shows the time "17:45".

37keV positron into 7um lead

The screenshot displays a Geant4Py simulation window titled "ExampleN03" with a "Run Start" button. The simulation parameters are as follows:

- Absorber Materials:** Aluminium, Lead
- Gap Materials:** liquidArgon, Scintillator, Air, Aerogel, Galactic
- Particles:** proton, mu+, gamma, e-, e+, mu-
- Interaction Processes:** 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

The main visualization area shows a 3D simulation of a lead absorber (a rectangular block) with a positron track (blue line) entering from the left and interacting within the lead. Green lines represent the paths of secondary particles or photons produced during the interaction.

33MeV electron into water, Ionization is on

The image shows a screenshot of a Geant4 simulation environment. The main window displays a 3D visualization of a 33 MeV electron beam entering a water volume, with ionization tracks shown in red. Overlaid windows include:

- VMware Tools Properties**: Options tab selected, Miscellaneous Options section with Time synchronization between the virtual machine and the host operating system.
- Konsole**: Terminal output showing `= 0.2, integral: 1` and `ulation needed : No`.
- Geant4Py**: Configuration window for Lesson1, Geant4 Commands, and Vis Commands. It includes a Run Start button and various simulation parameters.

Geant4Py Configuration:

- Materials: lead, water, gold, iron, air, aluminum
- Particles: gamma, e-, proton
- Physics Lists: phot, compt, conv, msc, eloni, eBrem, annihil, hloni
- Incident beam energy: 33.000 MeV
- Absorber thickness: 197.000 mm
- Number of Events: 1

At the bottom, a terminal window shows the command `G4SteppingManager::InvokeAtRestDoIt` and system icons for Printer and シェル (Shell).

Multiple scattering is on

The screenshot displays a Geant4 simulation environment. The main window shows a 3D visualization of a particle beam, represented by a dense cloud of red lines, originating from a point on the right and spreading out to the left against a cyan background. A red line indicates the beam's direction.

Overlaid windows include:

- VMware Tools Properties**: A dialog box with tabs for Options, Devices, Scripts, Shrink, and About. The Miscellaneous Options section is visible, showing a checkbox for "Time synchronization between the virtual machine and the host operating system" which is currently unchecked.
- Konsole**: A terminal window displaying the text: `= 0.2, integral: 1`, followed by a series of equals signs, and then `ulation needed : No`.
- Geant4Py**: A command interface window with tabs for Lesson1, Geant4 Commands, and Vis Commands. The Geant4 Commands tab is active, showing a list of commands: `/run/beamOn 100`, `/tracking/verbose`, `/event/abort`, `/event/verbose`, `/event/stack/status`, `/event/stack/clear`, `/run/initialize`, `/run/beamOn` (highlighted), `/run/verbose`, `/run/dumpRegion`, `/run/dumpCouples`, `/run/optimizeGeometry`, and `/run/breakAtBeginOfEvent`. An "Execute" button is located to the right of the command list. On the right side of the window, there is a "Start a Run" section with parameters: `Parameter: numberOfEvent Type: i`, `Parameter: macroFile Type: s`, and `Parameter: nSelect Type: i`.

The bottom of the image shows a Linux desktop environment with a taskbar containing icons for a printer, shell, and other applications. The system tray at the bottom right shows the time as 21:40 and includes icons for network, volume, and power.

Bremsstrahlung is on

The image displays a Geant4 simulation environment. The main window shows a 3D visualization of particle tracks, with a dense central region of tracks and a fan-like spread of tracks extending to the left. The tracks are color-coded, with red and orange at the center transitioning to green and blue towards the edges. The background is a light blue gradient.

Overlaid on the simulation are several windows:

- VMware Tools Properties**: A window with tabs for Options, Devices, Scripts, Shrink, and About. The Miscellaneous Options section is visible, showing a checkbox for "Time synchronization between the virtual machine and the host operating system" which is currently unchecked.
- Konsole**: A terminal window displaying the output of a command: `= 0.2, integral: 1` followed by a separator line and the text `ulation needed : No`.
- Geant4Py**: A control window with tabs for Lesson1, Geant4 Commands, and Vis Commands. It features a "Run Start" button and various configuration options:
 - Materials**: Radio buttons for lead, water (selected), gold, iron, air, and aluminum.
 - Particles**: Radio buttons for gamma, e- (selected), and proton.
 - Physics Lists**: Checkboxes for phot, compt, conv, msc (checked), eloni (checked), eBrem (checked), annihil (checked), and hlioni (checked).
 - Incident beam energy**: A text field containing "33.000 MeV" and a spinner control.
 - Absorber thickness**: A text field containing "197.000 mm" and a spinner control.
 - Number of Events**: A text field containing "1" and a slider control.

At the bottom of the screen, a taskbar shows the system tray with icons for a printer, shell, and other applications. The system clock in the bottom right corner displays "21:41".

Compton, photoelectric and pair creation are on

The image displays a Geant4 simulation environment. The main window shows a 3D visualization of particle tracks, with a dense cluster of tracks on the left and a more sparse distribution on the right. The tracks are color-coded, with red and orange indicating higher energy or interaction points. The background is a light blue gradient.

Overlaid on the simulation are several windows:

- VMware Tools Properties**: A window with tabs for Options, Devices, Scripts, Shrink, and About. The Miscellaneous Options section is visible, with a checkbox for "Time synchronization between the virtual machine and the host operating system" which is currently unchecked.
- Konsole**: A terminal window showing the output of a command: `= 0.2, integral: 1` followed by a separator line and the text `ulation needed : No`.
- Geant4Py**: A control window with tabs for Lesson1, Geant4 Commands, and Vis Commands. It features a "Run Start" button and various configuration options:
 - Materials**: Radio buttons for lead, water (selected), gold, iron, air, and aluminum.
 - Particles**: Radio buttons for gamma, e- (selected), and proton.
 - Interaction Processes**: Checkboxes for phot, compt, conv, msc, eloni, eBrem, annihil, and hloni, all of which are checked.
 - Incident beam energy**: A text input field containing "33.000 MeV" and a numeric spinner set to 33.
 - Absorber thickness**: A text input field containing "197.000 mm" and a numeric spinner set to 197.
 - Number of Events**: A text input field containing "1" and a slider.

The bottom of the screen shows a Windows taskbar with the system clock at 21:42 and the user name "yoshidah on linux-c6by".

Measurement with Wired

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: 10

Absorber Thickness: 1.000 cm 1 0 cm

Gap Thickness: 9.000 cm 9 0 cm

Section Size: 100.000 cm 100 0 cm

incident beam energy: 408.000 MeV 408 0 MeV

Cut Length: 0.287 mm 0 287 mm

Magnetic Field: 1.000 Tesla 1 0 Tesla

Number of Events: 1

WIRED3 HepRep Browser (version 3.14.1)

File Options Window Help

ile / home/yoshidah/tmp/Lesson2_00.heprep -- Parallel

Selected Item(s)

Trajectories[42]		Trajectories[45]		Trajectories[36]	
Attribute	Value	Attribute	Value	Attribute	Value
Ch	-1 e+	Ch	-1 e+	Ch	-1 e+
IMag	6.51706 ...	IMag	0.52866 ...	IMag	73.5972 ...
PDG	11	PDG	11	PDG	11
ID	43	ID	42	ID	2
PN	e-	PN	e-	PN	e-
PID	2	PID	2	PID	1
IMom-Z	2.12292 ...	IMom-Z	-0.08251...	IMom-Z	0.039552...
IMom-Y	4.8094 M...	IMom-Y	-0.18324...	IMom-Y	0.819382...
IMom-X	3.85161 ...	IMom-X	0.488971...	IMom-X	73.5927 ...
NTP	11	NTP	2	NTP	28

Cut Control... Label Control... Cut Control... Label Control... Cut Control... Label Control...

OO-2.0.1

Printer

3 4

oglsxviewer

Geant4Py

yoshidah on linux-c6by: /home/yoshid

Wired3-application-Application

06:52

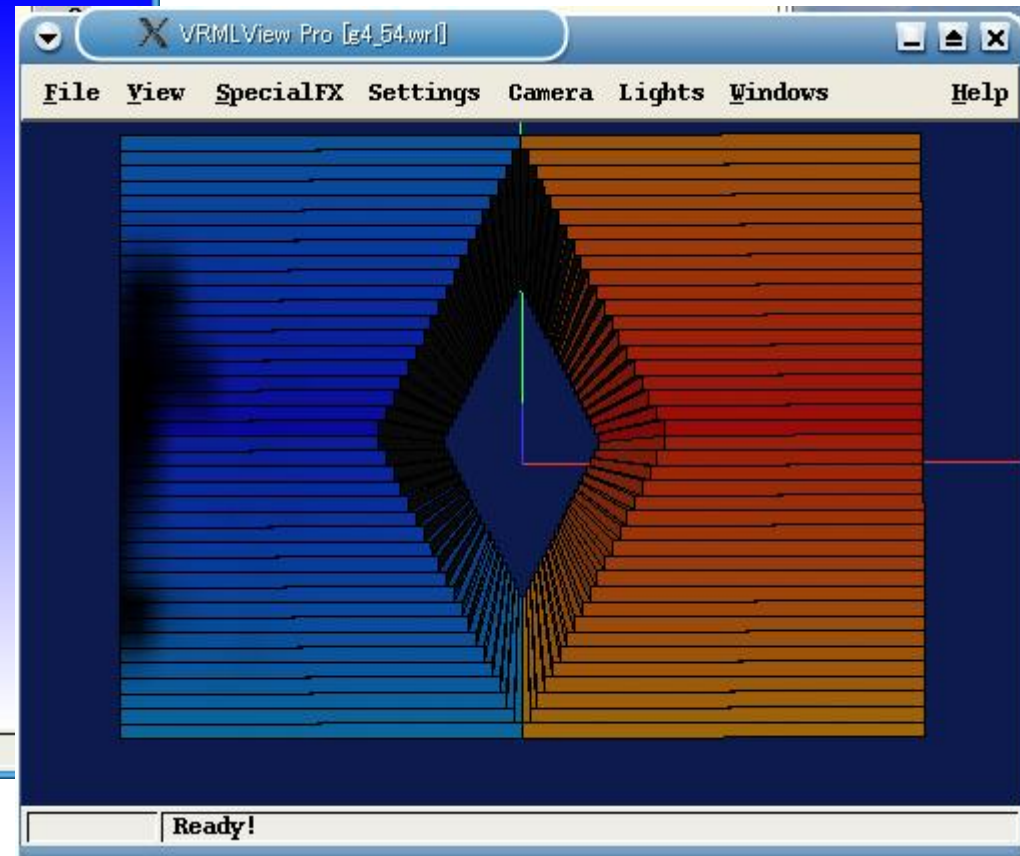
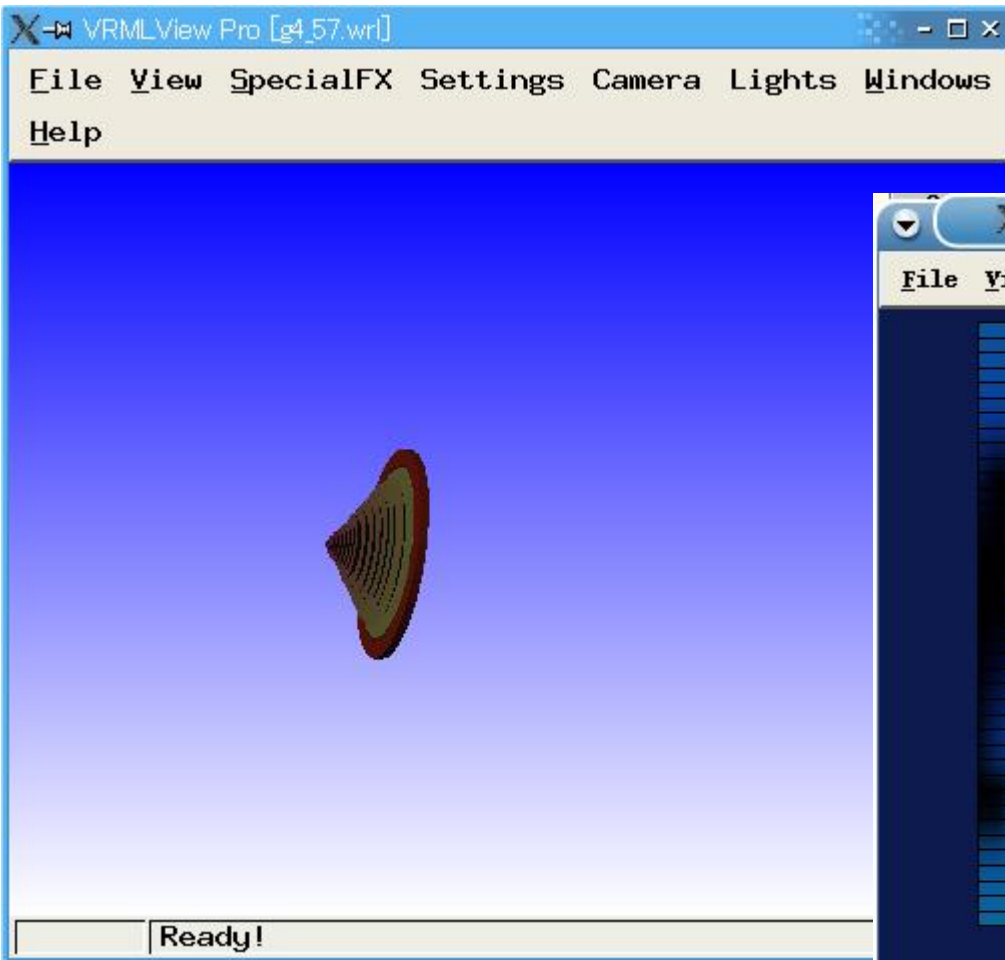
Courseware creation TODO

- Realistic and Standard geometries
 - Concrete and realistic “standard” geometries must be provided whose geometrical data must be available publicly.
- Generic and Customizable geometries
 - Some generic geometries which can be customized by teachers will be useful to create their own course ware
- Interactive customization
 - We need much more interactivities for creators of course ware to customize for their own applications

To-do List of Realistic and Standard Geometries for

- Standard ionization chambers
 - Track visualization in and around
 - Build up cap
 - Total number of created ions
- Curie well chamber
- Gamma camera
 - Number of photons
 - Energy spectrum
- PET
- GM counter
 - Track visualization
- etc.

MOMO: Tool prototype for Geometry creation a la
BEAMnrc:
giving the BEAMnrc parameters, you get the geometry



“Standard” Physics Lists are available

- A common physics list must be provided -> done by Denis Wright
- Medical max < 1 GeV
- Start with N03 by Michel
 - Switching on/off any processes
 - Hadronic processes
 - P elastic, inelastic
 - N elastic, inelastic
 -
 - Ion
 - Radioactive decays, generic decays
 - Choice of models
 - LEP, Bertine, Binary cascade
 - Process can be turned on one by one. Range cut and step size must be easily modifiable
 - Only the hadronic processes can be visualized – Michel’s cut magic

To conclude

- Geant4Py provides highly functional and in depth steering of Geant4 kernel
- It serves as the software bus to external GUI toolkits, analysis tools and others
- Preliminary educational examples are successfully created
- Existing Geant4 examples can be developed for educational applications
- Collaboration between medical specialists are mandatory for successful creation of educational materials