

GUI Examples using wxPython

***11th Geant4 Workshop
LIP, Lisboa***

***GUI parallel session
10 October 2006***

***Hajime Yoshida
Naruto University of Education, JST***

2000 statement on the interactivity refrained

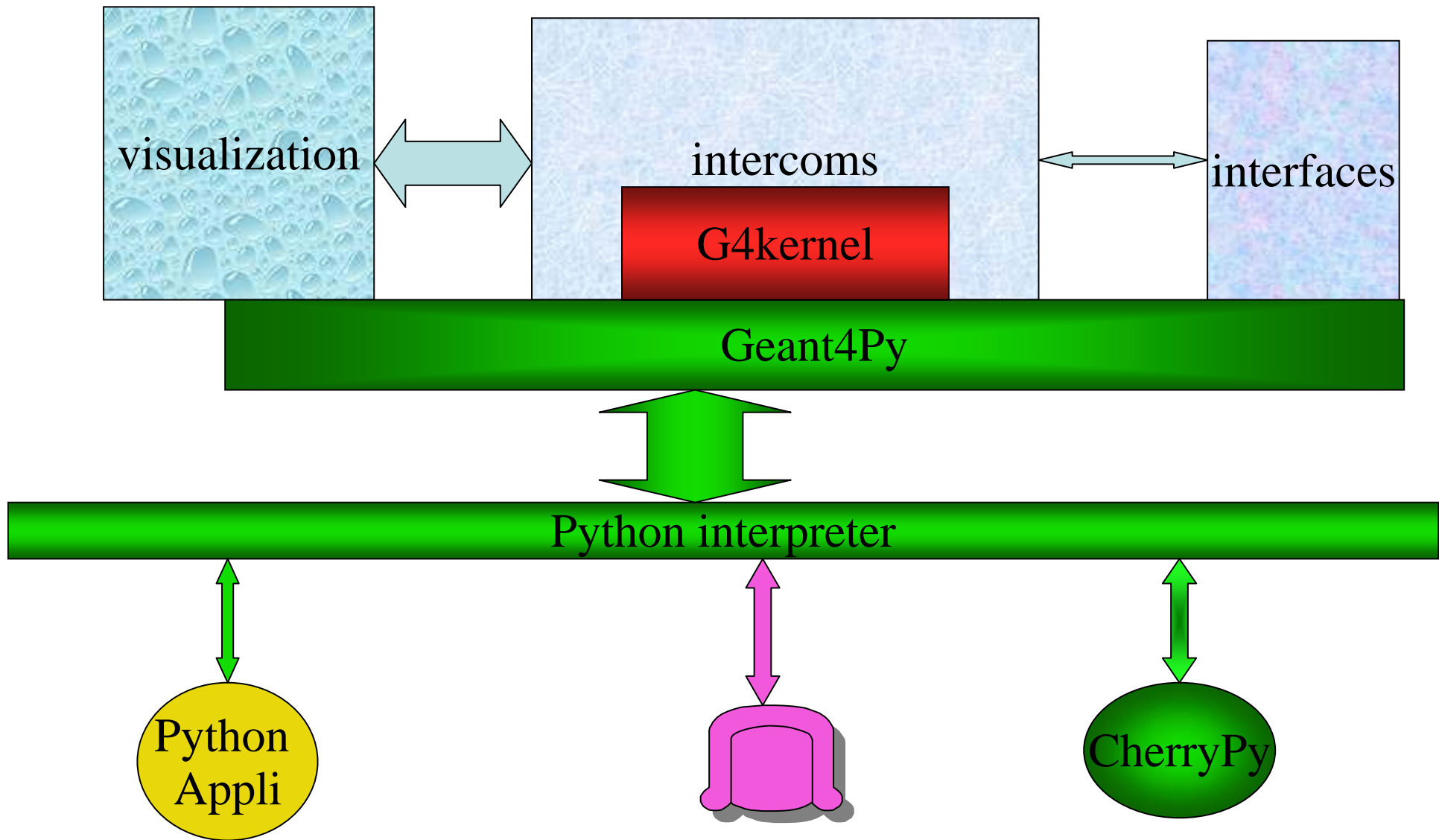
- There is an obvious requirement for Geant4 to fit into existing Interactive Frameworks. Examples of Interactive Frameworks are: Momo, Explorer, AVS, the BaBar Framework, Gaudi, OPACS, JAS, WIRED, ROOT. They each come with their own way of interacting with the applications inside.
- Geant4 already offers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



Geant4Py Wrapper

Use of Python Wrappers

case 1 = C++ + Python

- Create an application in C++ and wrap its classes as necessary
 - Examples are found in g4py/site-modules/
 - ExN03 geometry and physics list
 - Voxelized water phantom etc.
 - Performance isn't deteriorated,
- Much more interactive than the “old” scheme
 - Use of interactive Python shell : ipython
 - See attributes of the exposed objects
 - Powerful Python's data structure etc. are exploitable
- Python based GUI tool kits can be employed for better user friendliness
- Connection with analysis tools is straightforward

Use of Python Wrappers

case 2 = purely Pythonic scripting

- Ezgeom
 - 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

WxPython: GUI toolkit

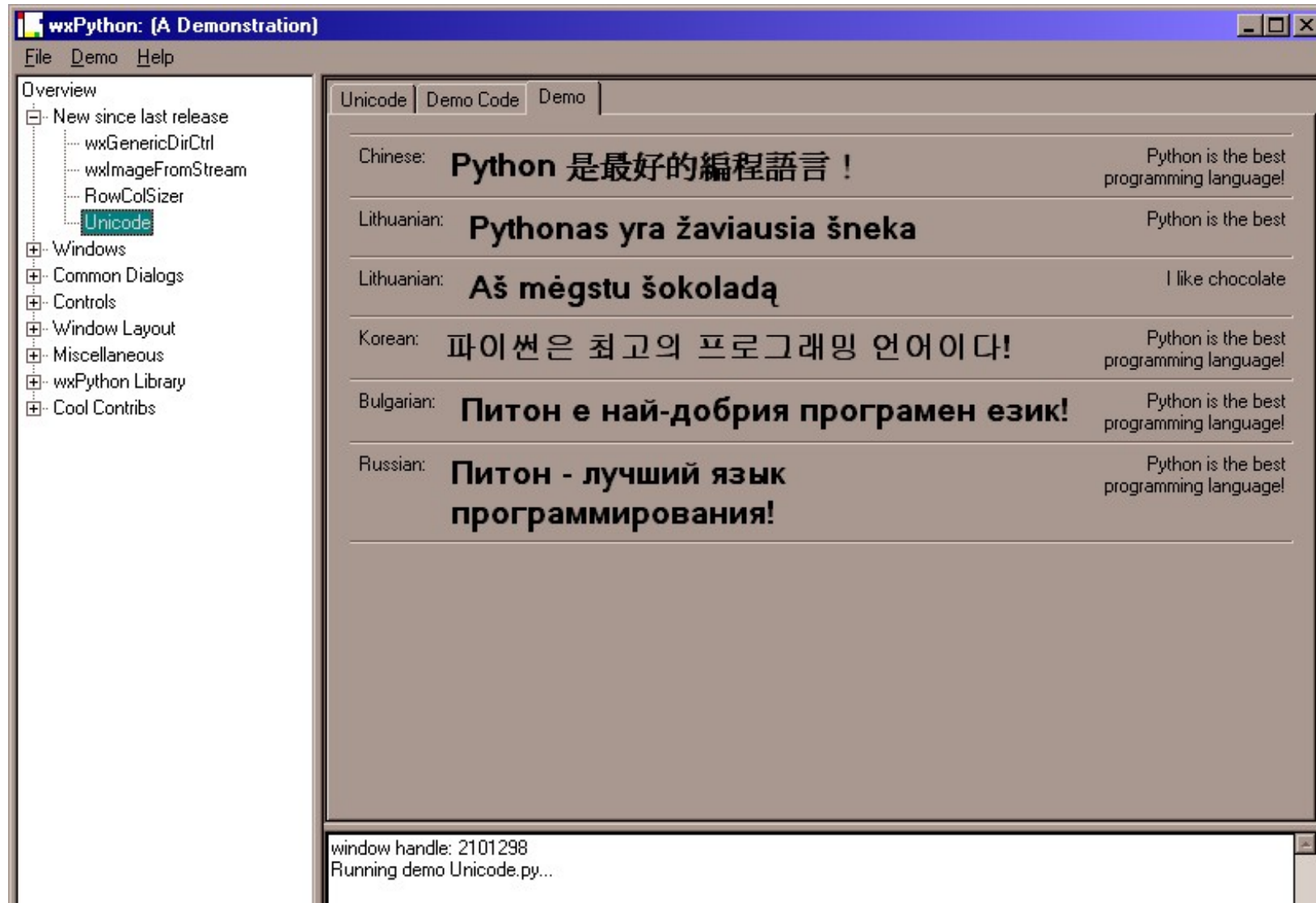
- Tkinter

- Is the default GUI of Python, since it was included from the beginning
- Is simple to use but lacking advanced widgets like tree etc.
- Multi platform but original Tk's look and feel

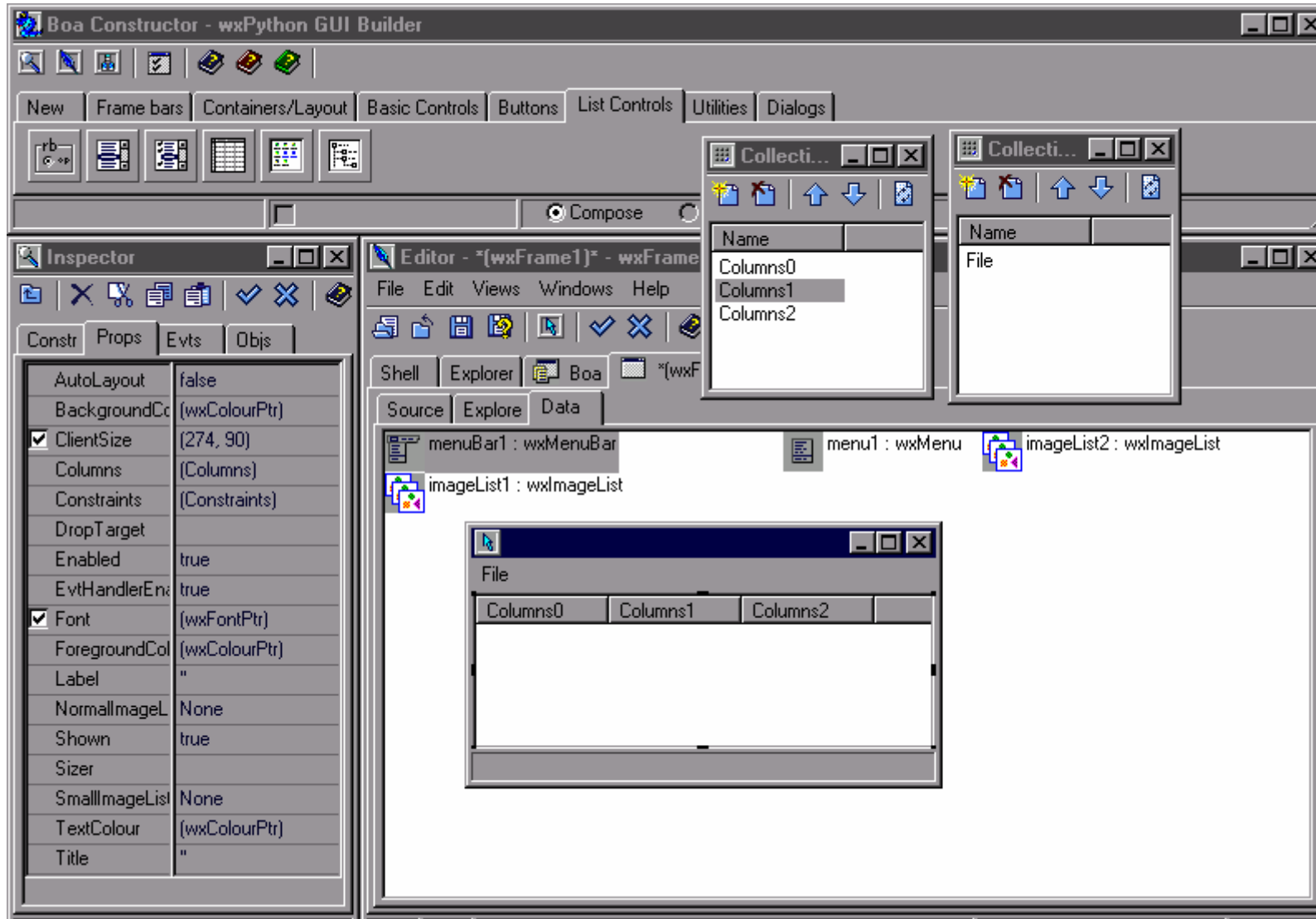
- WxPython

- Has been the second popular GUI toolkit after Tkinter
- Wxwidgets (C++ library) is wrapped with Python
- Many advanced features and widgets
- A little lengthy scripting (than Tkinter) to profit its power
- Multi platform, keeping each platform's look and feel
- A good book has come

Unicode is supported



boa_constructor; IDE



Extending user-friendliness

- Plot tools : [matplotlib, plot library a la Matlab](#)
- Web server : [CherryPy, purely Pythonic Web server](#)
 - Powerful template language supported
 - Session and cookie management etc.
- GUI toolkit : Tkinter, [They say about wxPython](#)
 - May replace the old GUI tools of Geant4
- **Geant4 for Education** project, combining the above user friendly environment
- [SIG: Python for Education](#)

Examples and Demonstration

- 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
 - The classes of Michel Maire's exampleN03 are wrapped
 - 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
- Demonstration
 - Suse Linux + Python + wxPython