

Python & Gtk: Where are we?

Python-gtk, gobject-introspection and stuff

Neil Muller

May 28, 2011

Talk Overview

- ▶ Brief overview of gtk
- ▶ python-gtk
- ▶ gobject-introspection
- ▶ Moving forward: Python3 and Gtk3

Gtk from many thousand feet

- ▶ Multiplatform Widget Library
 - ▶ Written in C, but OO design (gobject hierachy from glib)
 - ▶ LGPL v2.1 license
- ▶ Widget hierachy of containers
 - ▶ Button contains a label for text, or a image, or both, etc.
 - ▶ Layers of orientated boxes (HBox and VBox) as basic layout mechanism
- ▶ Tools for visual UI design and such exist
 - ▶ Glade -> produces XML files for gtk_builder and friends

Python-Gtk

- ▶ Static binding for the gtk2 library
- ▶ Python2 only
- ▶ Fairly idiomatic wrapper around gtk - numerous overrides of the default gtk API to make things easier / clearer in python
- ▶ Possible to write custom widgets in pure python
- ▶ Mostly complete coverage of gtk+ API up to 2.24
 - ▶ spottier for some of the newer stuff

Gobject Introspection

- ▶ The problem
 - ▶ Python-gtk is a large and complex code base
 - ▶ Needs to be kept manually in sync with gtk changes
 - ▶ hard to port to gtk3 and/or python3
 - ▶ Same issues apply to other gtk language bindings
- ▶ Gobject introspection
 - ▶ Annotate gtk+ API docs with indicators that can be used to generate bindings
 - ▶ Generates XML description of the API - generated from headers and compiled library (usually generated at install time)
 - ▶ Typelib (compiled version of XML) exists for speed
- ▶ Significantly reduces the effort to maintain language bindings
- ▶ Language bindings pick up extensions to gtk+ automatically
- ▶ “from gi.repository import Gtk”

Porting Python-gtk to GObject-introspection

- ▶ Efforts to keep things similar, but are some significant differences between the two API's
- ▶ numerous things renamed - helper scripts exist to help deal with this

Porting Python-gtk to gobject-introspection

- ▶ Efforts to keep things similar, but are some significant differences between the two API's
- ▶ numerous things renamed - helper scripts exist to help deal with this
- ▶ s/gtk/Gtk/ covers a lot of cases
- ▶ Constants moved into their own namespaces
 - ▶ `gtk.MESSAGE_INFO` becomes `gtk.MessageType.INFO` and so on
- ▶ Various default arguments differ between python-gtk and gobject introspection
- ▶ Some arguments now need to be specified as keyword arguments

Porting Python-gtk to GObject-introspection

- ▶ Efforts to keep things similar, but are some significant differences between the two API's
- ▶ numerous things renamed - helper scripts exist to help deal with this
- ▶ s/gtk/Gtk/ covers a lot of cases
- ▶ Constants moved into their own namespaces
 - ▶ `gtk.MESSAGE_INFO` becomes `gtk.MessageType.INFO` and so on
- ▶ Various default arguments differ between python-gtk and GObject introspection
- ▶ Some arguments now need to be specified as keyword arguments
- ▶ Gtk annotations still being tweaked as people make more use of it, so expect to find some occasional warts

Moving Forward: Python3 and Gtk3

- ▶ Python-object supports python2 and python3
- ▶ Can read typelib for both gtk2 and gtk3
 - ▶ gtk3 is recommended as has fewer unintrospectable methods & better support for introspection
- ▶ It is possible to write python programs that will work against both gtk2 & gtk3, but tricky
- ▶ Common subset of functionality excludes:
 - ▶ managing window size requests
 - ▶ Lots of the low level drawing stuff (due to move to cairo everywhere)
 - ▶ Various bits of image manipulation
 - ▶ Theming stuff
 - ▶ And some other things

- ▶ Changes from gtk2 to gtk3 are usually quite small though:
 - ▶ replace `size_request` with `preferred_width/height`
 - ▶ replace `gdk.Image` / `gdk.Pixmap` with cairo equivalents
 - ▶ redo drawing code to use cairo (which isn't too big a change from the gdk drawing stuff)
- ▶ gtk2 and gtk3 can be used with either python2 or python3, so that porting order is unimportant

- ▶ Changes from gtk2 to gtk3 are usually quite small though:
 - ▶ replace `size_request` with `preferred_width/height`
 - ▶ replace `gdk.Image` / `gdk.Pixmap` with cairo equivalents
 - ▶ redo drawing code to use cairo (which isn't too big a change from the gdk drawing stuff)
- ▶ gtk2 and gtk3 can be used with either python2 or python3, so that porting order is unimportant
- ▶ Recommended approach is:
 - ▶ port from `python-gtk` to `python-gobject` + `gobject-introspection`
 - ▶ move to `gtk3`
 - ▶ then port to `python3`