Tutorial Description

The excellent dvdauthor application includes a tool named 'spumux' which allows DVD menu creation, with custom buttons. This package, '*DvdAuthor Buttons*', is a Python-Fu plugin to the GIMP, and provides some shortcuts for making menu buttons.

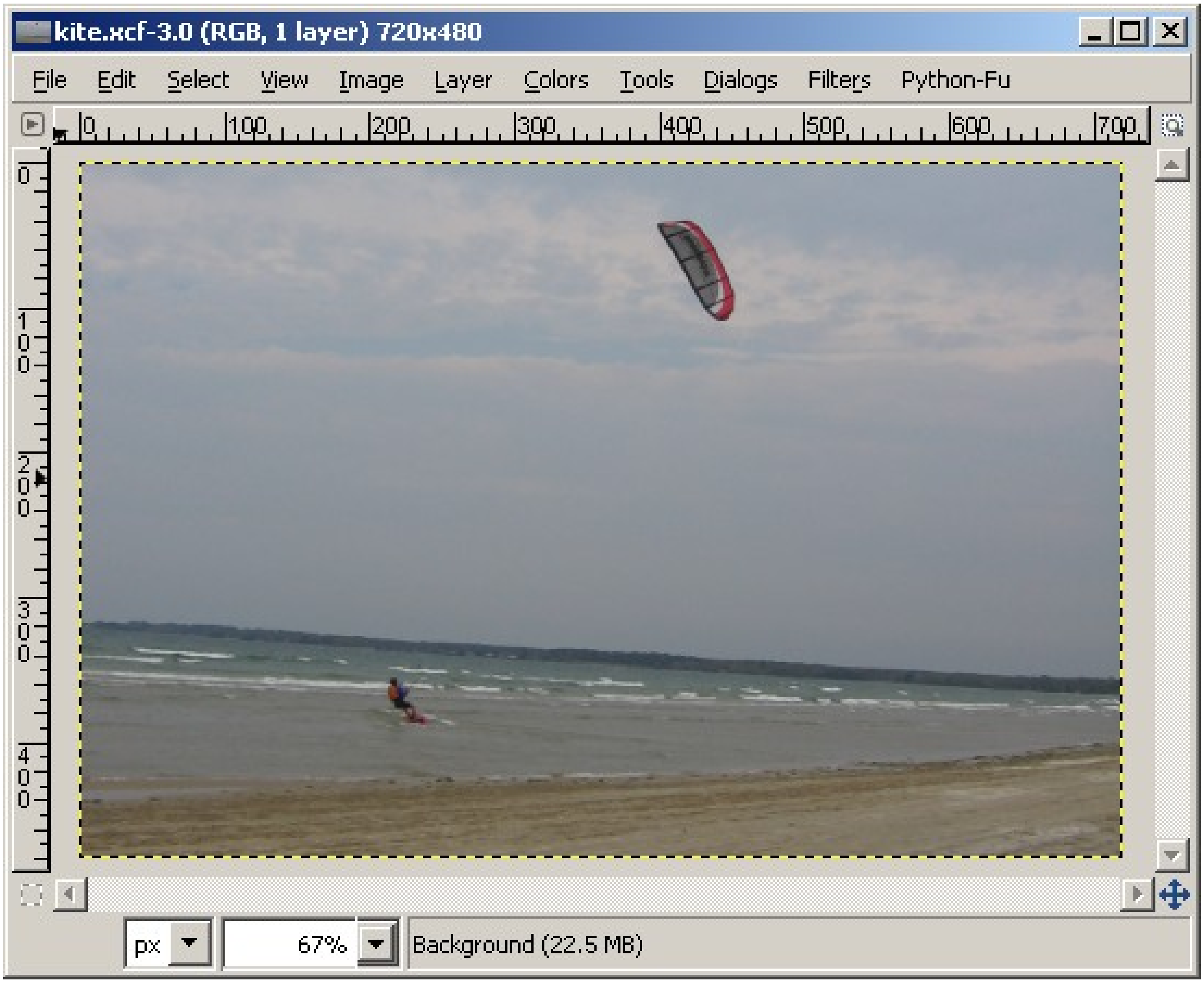
This tutorial's goal is to show how to:

- Create buttons within the GIMP
- Export these buttons, including images and positional information, for use with *dvdauthor's* spumux caption multiplexor

Without further ado, here we go!

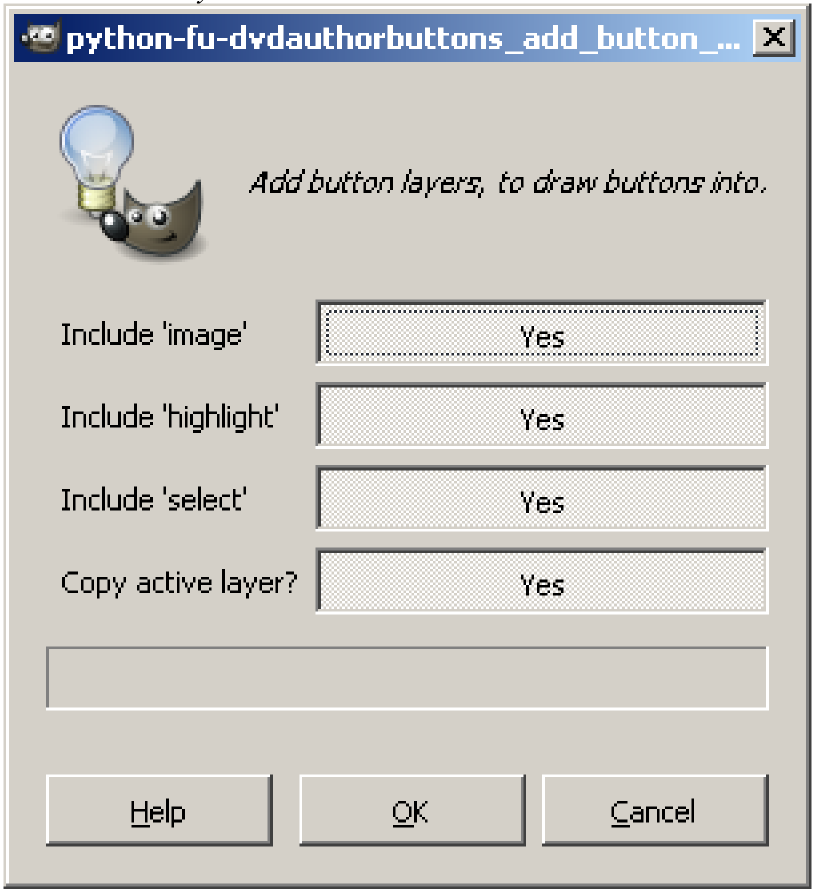
Tutorial Steps

- Create a new 720x480 image (why yes, that is me!)



Note: if you have an actual menu background (eg a frame from an MPEG), load it here instead, so you can align the buttons.

- First, I went ahead and added a new temporary layer, and put my button text into it.
- In this temp layer, I added the text for Play, Options, Bloopers, and Scenes. (Make sure the text is actually in the layer, and not in it's own layer!)
  This temporary layer will be used as a source for all the button layers.
- Now we need to add some layers to draw our button images. Select the menu item *Python Fu/DVDAuthorButtons/Add Button Layers*



python-fu-dvdauthorbuttons_add_button_...

Add button layers, to draw buttons into.

Include 'image' Yes

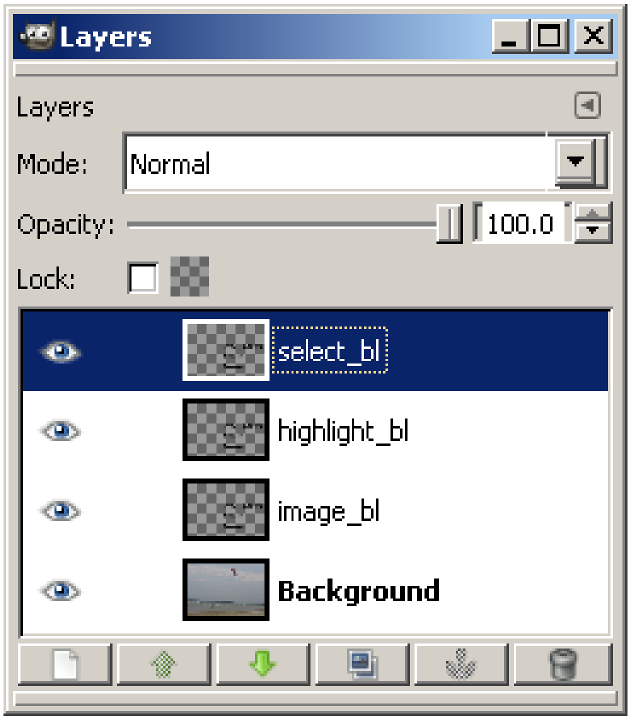
Include 'highlight' Yes

Include 'select' Yes

Copy active layer? Yes

- *For this example, leave all layers checked, and check "Copy active layer".*
- The "image" layer will always be shown, for all buttons (except the currently highlighted button).
- The "highlight" layer will be shown for one button at a time, when that button has the navigation cursor.
- The "select" layer will be shown for one button when the button is selected.
  Note, only ONE layer will be shown at a time! In my first try, I thought that the 'highlight_bl' layer would be superimposed over the 'image_bl' layer, but this is not the case.
- Normally, the newly created layers are completly transparent. However, we will use the "Copy active layer" option to fill the layers with our button text.
- So, make sure the temporary layer with the text in it is the current layer, check "Copy active layer", and click OK.
- After that, go ahead and delete the temporary layer.

- You should now see the following layers in the Layers dialog. ('bl' stands for *button layer*, in case you were wondering....)



Mode: Normal

Opacity: 100.0

Lock: □

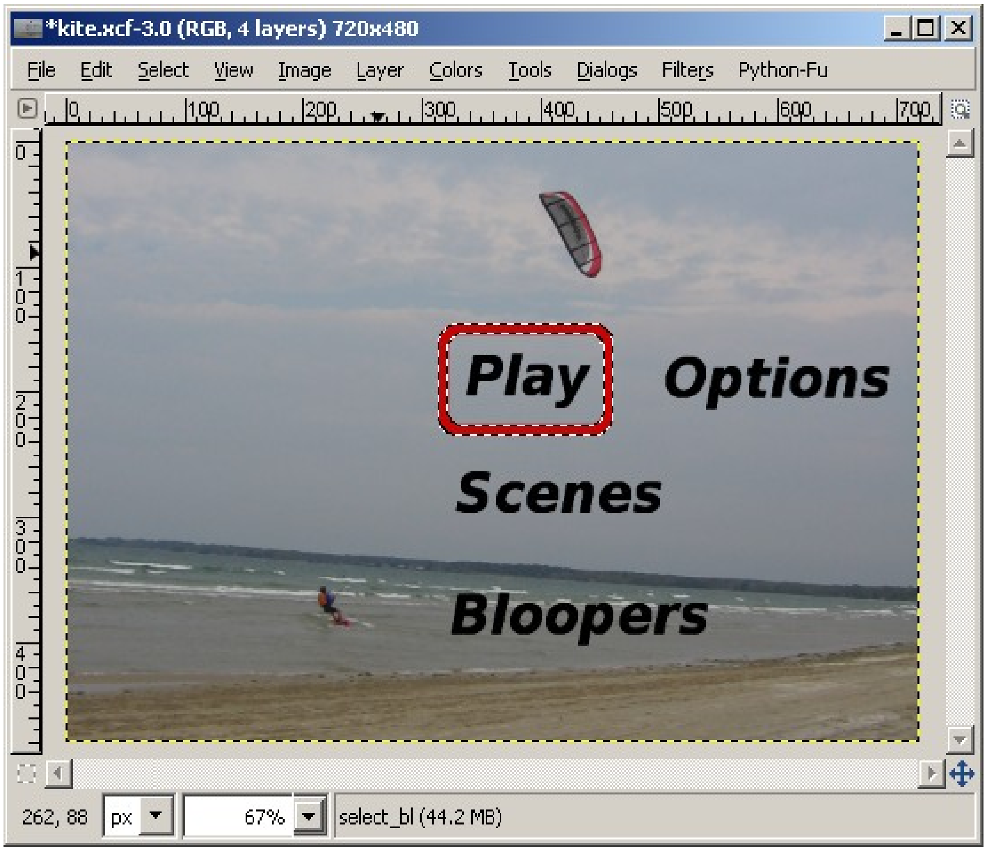
select_bl

highlight_bl

image_bl

Background

- Then, I created a rectangle selection around the “Play” text, rounded the corners (*Select/Rounded Rectangle*), and took a border of it (*Select/Border*).
- Next, I set the 'highlight_bl' as the active layer, and filled the selection with yellow.
- Then I set the 'select_bl' as the active layer, and filled the selection with red.
- Here is the result with all layers visible. You can't see the yellow, because the red 'select' image is obscuring it.

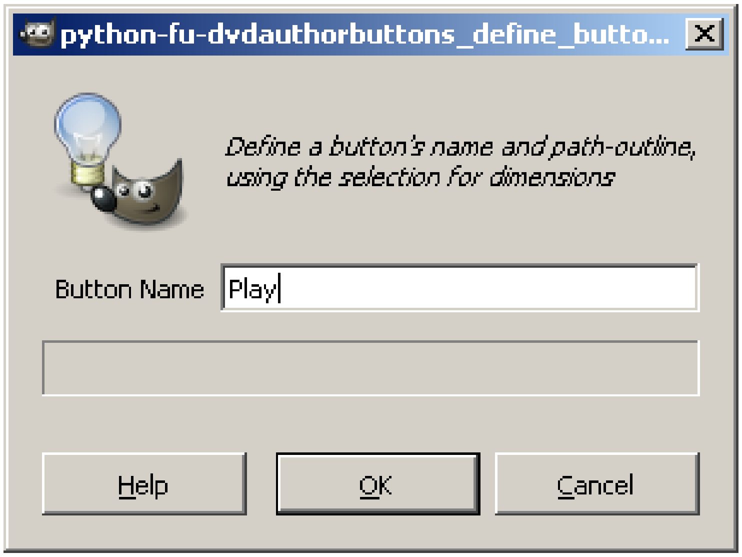


*kite.xcf-3.0 (RGB, 4 layers) 720x480

262, 88 px 67% select_bl (44.2 MB)

Next, we will indicate that this is actually a button.

- With the selection still active, select *Python-Fu/DvdAuthor-Buttons/Define a button using selection*

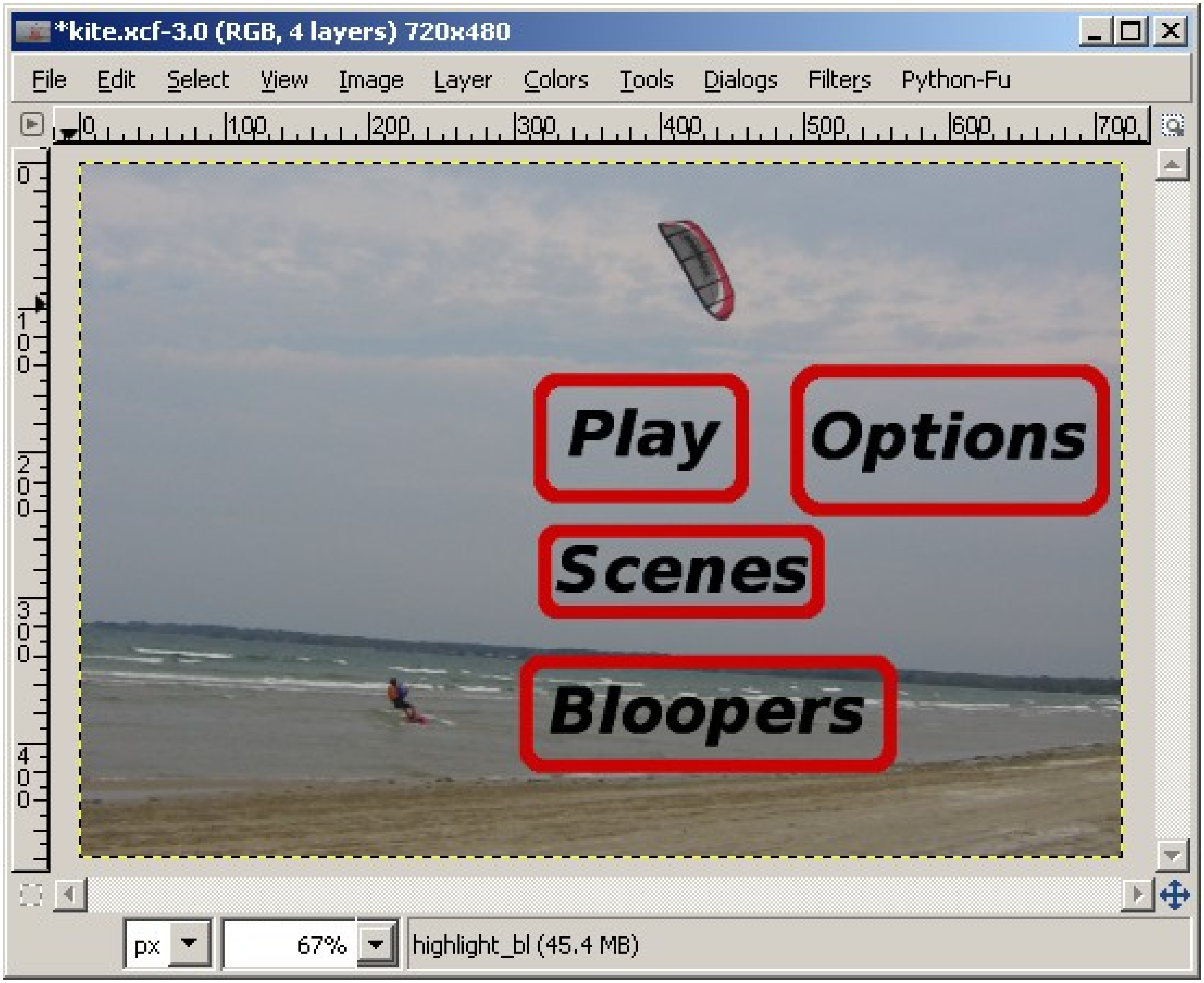


python-fu-dvdauthorbuttons_define_butto...

Define a button's name and path-outline, using the selection for dimensions

Button Name: Play

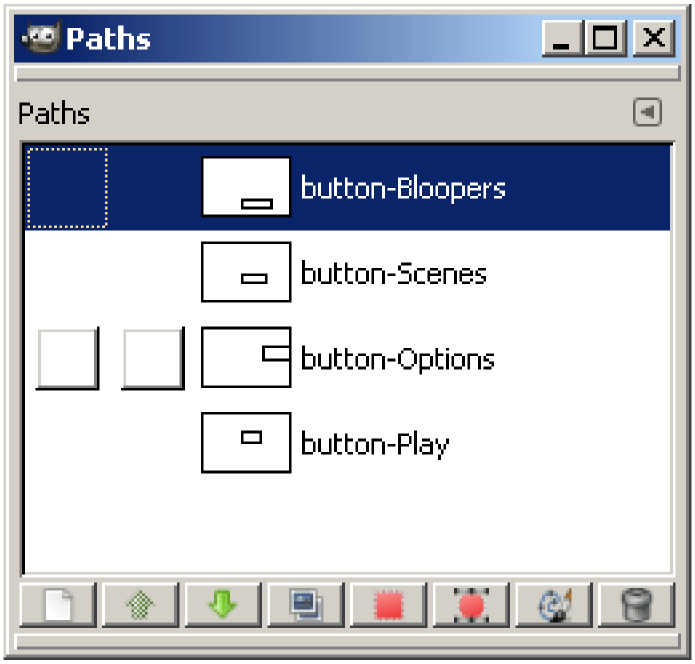
- I changed the button name to *Play*
- Then I repeated this procedure for the three other buttons. I did a bit of a sloppy job, but hey, I am in a rush here!
- NOTE: BE SURE YOU DO NOT OVERLAP ANYTHING!!!
- Here it is with all layers visible.



*kite.xcf-3.0 (RGB, 4 layers) 720x480

highlight_bl (45.4 MB)

- The buttons are actually stored as paths, so if you select *Dialogs/Paths*, you can see your button borders:



Paths

Paths

button-Bloopers

button-Scenes

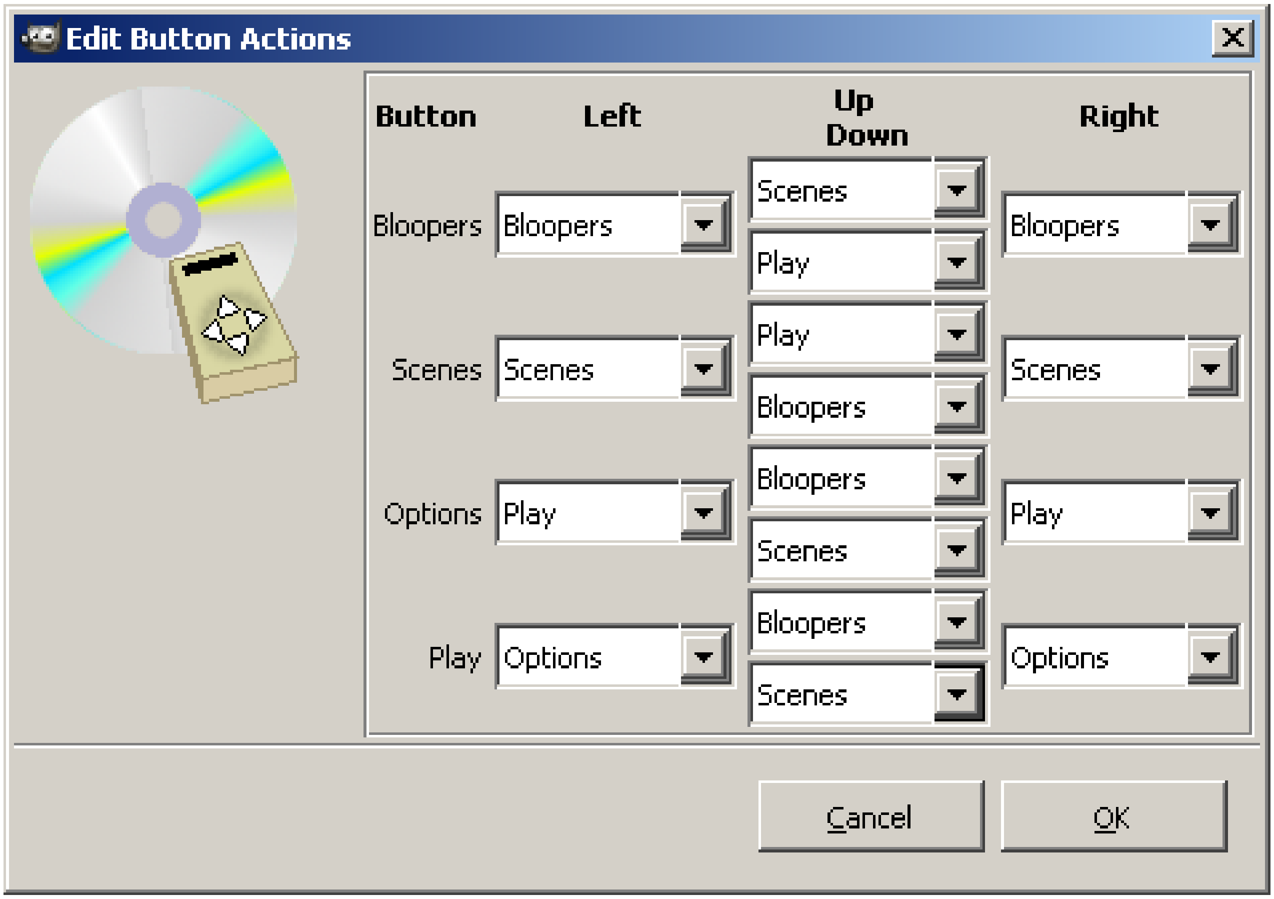
button-Options

button-Play

Next, we will define the button *actions*, which lets you specify what button to go to when you press an arrow key.

For example, when 'Play' is highlighted, and you press the Down key on your remote when watching the DVD, I want the selection to go to 'Scenes'

- I selected *Python-Fu/DvdAuthor-Buttons/ Edit Button Actions...*
- Then, I filled in the table, specifying what button to go to for each direction.
- Geez I hope this is self explanatory...

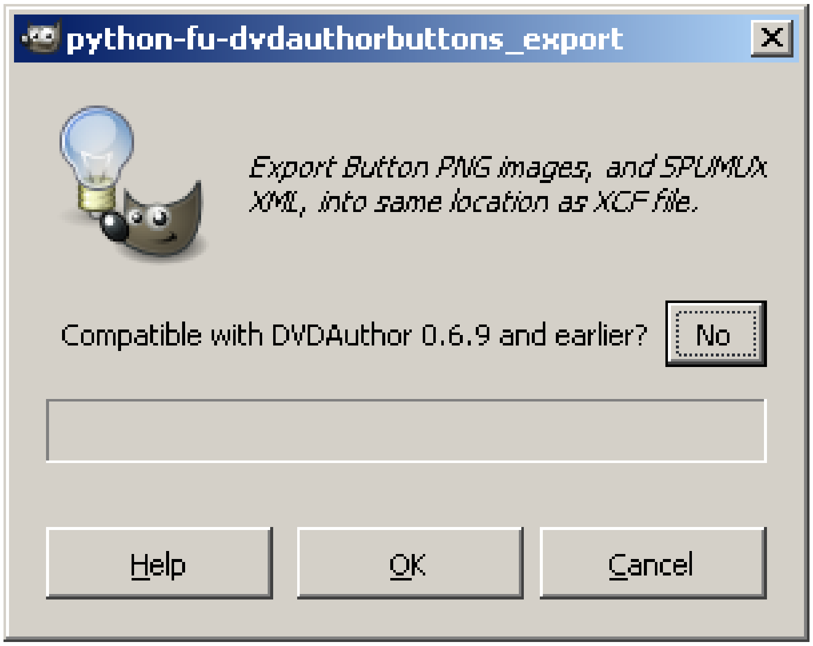


Edit Button Actions

Button	Left	Up Down	Right
Bloopers	Bloopers	Scenes Play	Bloopers
Scenes	Scenes	Play Bloopers	Scenes
Options	Play	Bloopers Scenes	Play
Play	Options	Bloopers Scenes	Options

- This says that for button “Bloopers”, left and right will go to itself, up will go to scenes, and down will go to play. For button “Play”, left will go to options, right will go to options, up will go to bloopers, and down will go to scenes (As you can see, I am a fan of wrapping around.)

- Save the file as an XCF, into the very same directory you want to save the spumux xml, and button images.
- Click the 'Export...' in the dvdauthor buttons menu.



Export Button PNG images, and SPUMUX XML, into same location as XCF file.

Compatible with DVDAuthor 0.6.9 and earlier? No

- This will generate one PNG image per layer, and will generate the spumux xml.
  The script takes the image's path (~/example/kite.xcf), removes the extension (~/example/kite) and then uses that as the basename for all the other files. Therefore, in this case it will generate *~/example/kite_image_bl.png*, *~/example/kite_select_bl.png, ~/example/kite_highlight_bl.png*, and finally, *~/example/kite_spu.xml*
  Here is the SPU XML file:

```
<?xml version="1.0" ?>
<subpictures>
 <stream>
  <spu force="yes" highlight="kite_highlight_bl.png" image="kite_image_bl.png"
select="kite_select_bl.png" start="00:00:00.00">
   <button down="Play" left="Bloopers" name="Bloopers"
           right="Bloopers" up="Scenes" x0="304" x1="564" y0="340" y1="421"/>
   <button down="Bloopers" left="Scenes" name="Scenes" right="Scenes"
           up="Play" x0="317" x1="514" y0="250" y1="315"/>
   <button down="Scenes" left="Play" name="Options" right="Play"
           up="Bloopers" x0="491" x1="711" y0="140" y1="244"/>
   <button down="Scenes" left="Options" name="Play" right="Options"
           up="Bloopers" x0="314" x1="462" y0="146" y1="235"/>
  </spu>
 </stream>
</subpictures>
```

You can now use this xml configuration file directly with spumux, for example:
%> cd ~/example
%> *spumux* **kite_spu.xml** < Menu.mpg > MenuWithCaption.mpg

Tips:

- Do *NOT* modify the names of the layers, in the Layers dialog. The export script is looking for layers named 'image_bl', 'highlight_bl', and 'select_bl'. Order is irrelevant.
- Paths may be moved by holding the 'alt' key, and dragging, when the path is active.
- I created a DV-NTSC template, so I don't have to type in 720x480, 80x72 dpi when creating a new image. Simply click '*New image',* and then select DV NTSC from the combo-box!
- The button's name will be going straight into an XML attribute, so don't use characters like # or **"** or **'** or **&** etc...
- Because the image as saved as an XCF, complete with layers and paths, you can simply reload the file (eg, menu.xcf), modify if desired (eg., add more buttons, rename buttons, change colours), save it, and then export again!
- You can rename the buttons in the Paths dialog, however make sure the "button-" prefix remains.
- When adding text, it may add it as a new layer. For example, when I added the 'Play' text, it went into it's own layer. You can right-click the layer (eg, the Play layer) and select "Merge Down", thus overlaying the 'Play' layer into the 'image_bl' layer.
- Version 0.6.10 of dvdauthor included changes for spumux, changing the button attribute from 'label' to 'name', and so if you still have 0.6.9 or earlier, you can select “Yes” for compatible with 0.6.9 or earlier, when exporting.

Part 2: Creating the DVD

The remainder of this tutorial is what to do with the exported files.

The first thing I did, was fire up a video editor (Adobe Premier Pro) and turn my background image (kite.png) into a short movie (kite.mpg). I kept it very short for this demo.

Then, I ran the command:

```
%> rm -fr kiteWithCap.mpg
%> spumux kite_spu.xml < kite.mpg > kiteWithCap.mpg
```

This complained that the buttons were not even, so I just edited kite_spu.xml and changed the y0/y1 values to all be even numbers. Future improvement for the script, I suppose!

This file is multiplexed! However, you cannot see it if you simply play the mpeg.
So, in order to test it out, I created a very simple dvdauthor config file.
Then, all that was left to do was run dvdauthor!

```
%> rm -fr DVD
%> dvdauthor -x dvd.xml
```

That's it! The rendered DVD has a looping MPG menu, with buttons that you can click on!
It can be played on your DVD player, and it works in WinDVD4 and most soft DVD players.

Hope you enjoy it, and please send me a note!

Jamie Strachan

frostfreek@yahoo.com