LINUX LABS :
FILE SYSTEMS - PT1

NEW SERIES!

LibreOffice 3
The Document Foundation

Text Document   Drawing

Spreadsheet   Database

Presentation   Formula

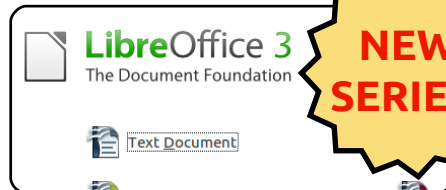Open...   Templates...

NEW SERIES - LIBREOFFICE
PART 1 : AN INTRODUCTION TO THE OPENOFFICE REPLACEMENT

Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

NEW SERIES!
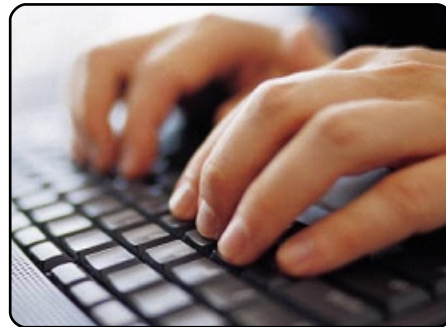
Guidelines for submitting an article to Full Circle. We rely on reader submissions so please write!

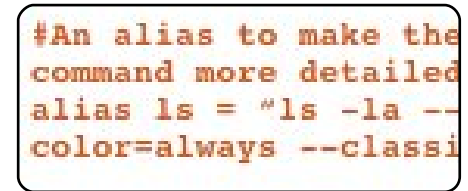What's a file system, and what's an EXT3? Sit down and let your Uncle Robin explain...

```
#An alias to make the
command more detailed
alias ls = "ls -la --
color=always --classi
```

Tonic: C   Mode: <no mode>

Welcome to another issue of Full Circle!

This month, we have a new series starting. A couple of months ago, Elmer Perry began emailing me some articles on using OpenOffice. I held them in reserve (since the Virtualization series was still running), and, by the time I got around to using them, LibreOffice had been released. Thankfully, Elmer volunteered to update the articles from OpenOffice to LibreOffice, so **big** thanks to Elmer for that. In case you hadn't heard, many people now disagree with the folks who own OpenOffice, so the code has been forked and the programmers behind OpenOffice are now working on LibreOffice. Expect LibreOffice to overtake OpenOffice as it's already been announced as being the default office suite in Ubuntu 11.04 and probably the same will happen with many other Linux distributions as OpenOffice will now probably sit stale and stagnant.

We've also started releasing the **long** awaited special editions. The first one, *The Perfect Server*, is simply a collection of the server articles from past issues of FCM. Expect a special edition with my old GIMP series, and <u>several</u> special editions for the Python series.

If you don't know your EXT3 from your Reiser, Robin Catling (of podcast fame) has written two informative articles on file systems, the first of them is used this month with part two coming next month.

You'll notice that the Letters page is only one page this month. Maybe I smell funny, but you folks seem to have stopped emailing me. Drop me an email telling me what you think of the magazine, what we should change (or improve upon), or just let us know what you've been up to with your favorite distro. Either way, it'll make you feel fuzzy inside and stop me feeling paranoid.

All the best, and keep in touch.
Ronnie
ronnie@fullcirclemagazine.org

This magazine was created using :

SCRIBUS
Desktop Publishing for Linux
Initialization...

gimp

LibreOffice
The Document Foundation

creative commons

Full Circle Podcast

Released every two weeks, each episode covers all the latest Ubuntu news, opinions, reviews, interviews and listener feedback. The Side-Pod is a new addition, it's an extra (irregular) short-form podcast which is intended to be a branch of the main podcast. It's somewhere to put all the general technology and non-Ubuntu stuff that doesn't fit in the main podcast.

Hosts:
Robin Catling
Ed Hewitt
Dave Wilkins

http://fullcirclemagazine.org

AUDIO MP3   AUDIO OGG
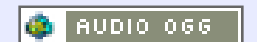
Hello Linux Open-Xchange

Where can we find refuge from Microsoft Exchange? One option is Open-Xchange, a groupware suite that serves as a replacement for Microsoft Exchange. From a licensing and cost perspective, OX looks like a great deal--but that means nothing if your users are going to revolt. So let's look at how OX stands up from the user's perspective, and whether you can cut the ties or not and still keep users happy.

Now, it doesn't take much to convince Linux-friendly admins that Exchange is expendable. But many admins are just as happy using Mutt and IMAP rather than Exchange, and you're probably not going to convice the account department that Mutt is a suitable replacement for Outlook.

Source: linuxplanet.com

Wearable, Atom-powered Computer

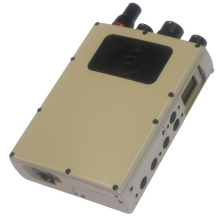Eurotech subsidiary Parvus announced two Linux-compatible, Intel Atom-based devices aimed primarily at the military market. The Zypad BR2000 is a rugged, 1.8-pound, wearable computer and vehicle server with wireless options and flash-based storage, while the DuraCOR 830 is a rugged tactical mission processor subsystem with dual avionics interfaces and PC/104 expansion, says the company.

Both [...] are touted for supporting so-called Size, Weight and Power (SWaP) constrained vehicle and aircraft platforms, as well as Command and Control (C2) applications. Both devices run Linux or Windows Embedded operating systems [...]. In addition, the company says it is considering adding Android to the Zypad BR2000.

Source: linuxfordevices.com

Full Circle Notifier - Beta Release!

Our very own **Full Circle Notifier** is now at 1.0.2. FCN is a small application that sits in your system tray and will not only announce issue/podcast releases, and can be set to automatically download them for you too! Several people have created packages of FCN and translations are starting. **For more info, see the FCN Google Group:**
http://goo.gl/4Ob4
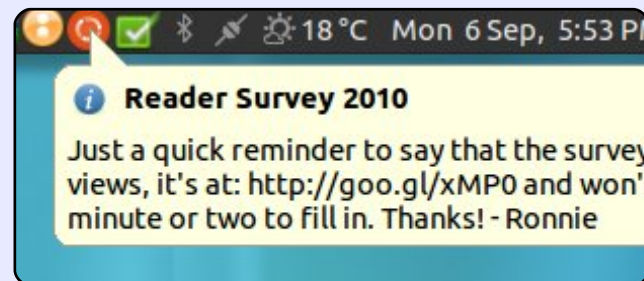
Canonical To Donate 25% Ubuntu One Revenues To Gnome Foundation

Banshee Team, which is primarily pushing Microsoft's .Net trojan Mono into the GNU/Linux world, has bent to Canonical's new proposal to give a 75% cut from Amazon store affiliated sales.

Banshee will also add Canonical's Ubuntu One store as default. Now, Canonical will donate 25% of its revenue to GNOME. They will now do the same for Rhythmbox.

With Novell's death Canonical is taking a big risk by making Mono based Banshee its default music player. Will Canonical soon be facing Microsoft's wrath of being accused of infringing on its Linux patents?

Source: muktware.com

This issue I intend to cover two topics: creating a to-do list using Conky and Bash, and introducing the basics of Zenity.

Both of these were requests. I'll be covering the way I use Conky with my To-Do list, and I'll be suggesting a few other additions one could make. However, I won't go into any of the additions in great detail unless I have an influx of requests for it.

To-Do List

The way I do a To-Do list is simply by creating a symbolic link in my Dropbox folder to a directory I called *Reminders*. Within the directory, I have a bunch of files sorted into topics (university, FCM, work, and personal), and then in Conky I call a Python script (which I wrote) that goes to each file, and prints each line with a "-" before it. This can be done in any language one would like. Shown above is my Python script for anyone who is interested.

Be sure to change "Reminders" into "<path from home directory>" for it to work - the variable home takes care of the "/home/$USERNAME/" part, so specify only the portion after that. I also tell it to ignore any hidden backup files - files that end with a tilde (~). I realize that my method is extremely low-tech, but it works. If you want to add due-dates in, simply type them into the file while you're adding items to it. Once you start using dates though, it becomes difficult to sort the items properly. For that, I've written two scripts. The first one is shown right, the other is shown at the top of the next page.

What these two scripts do is quite simple. The createToDo.sh script takes all the items within a file (I put it into the $file variable), removes the date (in the format: Month Day HH:MM), and replaces it with unix times (seconds since the unix epoch, aka 1st of Jan 1970 00:00), which is then written into a file (a file that is deleted at the beginning of each run of the script, to avoid duplicates). Once it has done this, it takes the file and sorts it from smallest to largest number (nearest date to "latest" date, i.e. first thing due to last thing due).

```python
#!/usr/bin/env python
import os

home=os.path.expanduser("~")

for root, dirs, files in
os.walk(os.path.join(home,"Reminders")):
    for infile in [f for f in files]:
        if(infile.endswith("~")!=True):
            fh=open(os.path.abspath(os.path.join(root,infile)))
            for line in fh:
                print("- "+line, end=' ')
            fh.close()
```

```bash
createToDo.sh:

#!/bin/bash
file=~/ToDo.txt
toDo=~/toDo.txt
if [[ -e $toDo ]]; then
    `rm "$toDo"`
fi
while read line; do
    date=`date -d"$(echo "$line"|sed 's/\(.\) -.*$/\1/g')" +%s`;
    echo "$(echo "$line"|sed -e s/".*-"/"$date -"/g)" >> "$toDo";
done < "$file"

if [[ -e "$toDo" ]]; then
    temp=`sort -n "$toDo"`
    echo "$temp" > "$toDo"
fi
```

```
printToDo.sh:

#!/bin/bash
toDo=~/toDo.txt
while read line; do
    if [[ "$line" != "" ]]; then
        date=`date -d@"$(echo "$line"|sed -e s/"-[^-]*$"//g)" +"%a %b %d %H:%M"`
        echo "$(echo "$line"|sed -e s/".*-"/"$date -"/g)";
    fi
done < "$toDo"
```

The printToDo.sh script simply takes each line from within the newly-created toDo.txt file, replaces the epoch time with a normal date, and prints it out. The first script could become an hourly cronjob (or every few minutes, if you prefer), and the second one could be run from Conky as a normal bash script. I haven't done too much testing for these scripts, but they do certainly work. I'm not sure if there is an easier method, but I think these scripts might teach/introduce you to more.

If you improve the scripts, I'd like to hear about it, and I'd be more than happy to list a few solutions in next month's C&C.

Zenity

For those of you who don't know Zenity, it's a command-line tool intended to create dialog boxes (graphical elements). Since I know that many users are shy of the command-line when first starting off, this might be useful for anyone who is trying to help a beginner along. I plan to cover the extreme basics as an introduction to Zenity, and I will write an in-depth tutorial for next month.

Zenity is capable of creating text-entry windows, calendar dialogs, info windows, progress dialogs, notification icons, list dialogs, save dialogs, checklists, error windows, and so forth. A few examples are as follows:

```
<command>|  tee  > (zenity --progress --pulsate) >file
```

This command runs a progress bar as long as tee is reading in input from STDIN, and then saves it in a file. In this case, you have to pipe the output of any command to tee.

```
zenity --question --test "Question ?"; echo $?
```

This command creates a dialog box with a question and an OK/Cancel button. Echo $? returns 0 if the user hits OK, and 1 if the user hits cancel (useful for user-intervention). This is the exact syntax for warnings as well - simply replace "--question" with "--warning".

```
<command>|zenity --text-info --width <size in pixels>
```

This command takes the output of whatever command is piped to it, and prints it within a textbox within the dialog box.

```
input=$(zenity --entry --text "How are you?" --entry-text "enter text here");
echo $input
```

This Zenity command creates a text-input window, and returns the entry within the variable $input (hence the echo statement).

```
zenity --error --text "An error occurred!"
```

This command creates an error window and puts the text in the window.

There are quite a few more commands that Zenity offers, but this should get any enthusiastic coder prepared for most scripting needs. Next month, I plan to implement some of these commands into a useful little script. If anyone has a request as to what the script should do, feel free to send me an email at lswest34@gmail.com. If you do email me, please put FCM or C&C in the subject line, so that I don't overlook it! Any comments on this article, or requests in general, are always welcome.

Lucas has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. You can email Lucas at: lswest34@gmail.com.

Welcome back. This time we will re-address GUI programming, but this time we will be using the pyGTK library. We won't be working with a GUI designer right now, we'll just be working with the library.

Use Synaptic to install python-gtk2, python-gtk2-tutorial, and python-gtk2-doc.

Let's jump right in and make our first program using pyGTK, it's shown above right.

For awhile, we will be building on this simple code set. On line #3 is a new command. The line "pygtk.require('2.0')" means that the application will not run unless the pygtk module is at least version 2.0. In the __init__ routine, we assign a window to the self.window variable (line 8), and then show it (line 9). Remember that the __init__ routine is run as soon as we instantiate the class (line 13). Save this code as "simple1.py".

Run it in a terminal. You'll see a simple window show up somewhere on your desktop. On mine, it shows up in the upper left corner of my desktop. In order to end the program, you have to hit Ctrl-C in the terminal. Why? We haven't added any code to destroy and actually end the app. That's what we'll do next. Add the following line before the self.window.show() line...

```
self.window.connect("delete_e
vent", self.delete_event)
```

Then after the gtk.main() call, add the following routine...

```
def delete_event(self,
widget, event, data=None):
    gtk.main_quit()
    return False
```

Now save your app as "simple2.py", and, once again, run it from a terminal. Now, when you click the "X" on the title bar, the application will exit. What is actually happening here? The first line we added (self.window.connect...) connects the delete event to a callback

```
# simple.py
import pygtk
pygtk.require('2.0')
import gtk

class Simple:
    def __init__(self):
        self.window = gtk.Window(gtk.WINDOW_TOPLEVEL)
        self.window.show()
    def main(self):
        gtk.main()
if __name__ == "__main__":
    simple = Simple()
    simple.main()
```

routine, in this case self.delete_event. By returning "False" to the system, it also destroys the actual window from system memory.

Now, I don't know about you, but I prefer my applications to open in the center of the screen, not someplace random, or in a corner - where it might be obscured by something else. Let's modify the code to do this. All we need to do is add the following line before the self.window.connect line in the __init__ function:

```
self.window.set_position(gtk.
WIN_POS_CENTER)
```

As you might guess, this sets the position of the window in the center of the screen. Save the app as "simple3.py" and run it.

That's much nicer, but there's not much there. So, let's try to add a widget. If you remember WAY back when we worked with Boa Constructor, widgets are simply predefined controls that we can add to our window to do things. One of the simplest controls to add is a button. We will add the following code right after the self.window.connect line in our previous code in the __init__ routine:

```
self.button =
gtk.Button("Close Me")

self.button.connect("clicked"
,self.btn1Clicked,None)

self.window.add(self.button)

self.button.show()
```

The first line defines the button, and the text on the button surface. The next line is the connector to the click event. The third line adds the button to the window, and the fourth line shows the button on the window surface. Looking at the self.button.connect line, you'll see that there are three arguments. The first is the event we want to connect to, the second is the routine that will be called when the event is triggered, in this case "self.btn1Clicked", and the third is the argument (if any) that will be passed to the routine we just defined.

Next, we need to create the self.btn1Clicked routine. Put this after the self.delete_event routine:

```
def
btn1Clicked(self,widget,data=
None):

    print "Button 1 clicked"

    gtk.main_quit()
```

As you can see, the routine doesn't do much. It prints in the terminal "Button 1 clicked", and then calls the gtk.main_quit() routine. This will close the window and terminate the application - just as if you had clicked the "X" on the title bar. Again, save this as "simple4.py", and run it in a terminal. You'll see our centered window with a button that says "Close me". Click on it, and the application closes, as designed. Notice, however, that the window is much smaller than it was in the simple3.py application. You can resize the application, but the button resizes with it. Why is this? Well, we simply shoved a button into the window and the window resized to fit the control.

We sort of broke the rules of GUI programming by putting the button directly on the form, without using a container. Remember back when we did our first series on GUI programming using Boa Constructor - we used sizer boxes (containers) to hold our controls. We should do this, even if we only have just one control. For our next example, we'll add a HBox (Horizontal box) to hold our button, and add another button. If we wanted a vertical container, we would use a VBox.

To start, use "simple4.py" as our base code. Delete everything between the lines self.window.connect(...) and self.window.show(). This is where we will add our new lines. The code for the HBox and our first button are...

```
self.box1 = gtk.HBox(False,0)

self.window.add(self.box1)

self.button =
gtk.Button("Button 1")

self.button.connect("clicked"
,self.btn1Clicked,None)

self.box1.pack_start(self.but
ton,True,True,0)

self.button.show()
```

Breaking down this code, we add a HBox, naming it self.box1. The parameters we pass to the HBox are homogeneous (True or False), and a spacing value:

```
HBox =
gtk.HBox(homogeneous=False,
spacing=0)
```

PROGRAM IN PYTHON - PART 20

The homogeneous parameter controls whether each widget in the box has the same size (width in the case of an HBox and height in the case of a VBox.) In this case, we pass it false, and a spacing value of 0. Next, we add the box to the window. Now, we create the button as before, and connect the clicked event to our routine.

Now, we come to a new command. The self.box1.pack_start command is used to add the button to the container (HBox). We use this command instead of the self.window.add command for the widgets we want to be in the container. The command (as above) is...

```
box.pack_start(widget,expand=
True, fill=True, padding=0)
```

The pack_start command has the following parameters. First is the widget, next is expand (True or False), then fill (True or False), and a padding value. Spacing for the containers is the amount of space in between the widgets, and padding is for the right/left side of the widgets. The expand argument allows you to choose whether the widgets in the box will fill all the extra space in the box (True), or if the box shrinks to fit the widgets (False). The fill argument has an effect only if the expand argument is True. Finally we show the button. Next is the code for the second button:

```
self.button2 =
gtk.Button("Button 2")

self.button2.connect("clicked
",self.btn2Clicked,None)

self.box1.pack_start(self.but
ton2,True,True,0)

self.button2.show()

self.box1.show()
```

Notice that this code is pretty much the same thing as the first button widget. The last line of this new code shows the box.

Now, we have to add the self.btn2Clicked routine. After the self.btn1Clicked routine, add the following code...

```
def
btn2Clicked(self,widget,data=
None):

    print "Button 2 clicked"
```

and in the btn1Clicked routine, comment out the line:

```
gtk.main_quit()
```

We want both buttons to print their "Button X clicked" response without closing the window.

Save this as "simple4a.py". Run it in a terminal. What you will see is a centered window with two buttons (right up to the edges of the window) marked "Button 1" and "Button 2". Click on them and notice that they properly respond to the click event as we have discussed. Now, before closing the window, resize it (drag at the bottom right of the window), and notice that the buttons grow and shrink equally as you resize the window. To understand the expand parameter, change the code for the self.box1.pack_start from True to False in both lines. Re-run your program and see what happens. This time, the window starts out looking the same, but when you resize the window, the buttons stay the same width, and there is empty space to the right as you expand the window. Next, change the expand parameter back to True and set the fill parameter to False. Re-run and notice that the buttons stay the same width, but there is empty space to the left and right of the buttons as you resize the window. Remember the fill parameter doesn't do anything if the expand parameter is set to False.

Another way to pack widgets is by using a table. Many times, if everything you have can fit into a grid-like structure easily, then a table is your best (and easiest) bet. You can think of a table like a spreadsheet grid with rows and columns holding widgets. Each widget can take up one or more cells - as your application requires. Maybe the following diagram will help visualize the possibility. Here is a 2x2 grid:

```
  0           1          2
0+----------+----------+
 |          |          |
1+----------+----------+
 |          |          |
2+----------+----------+
```
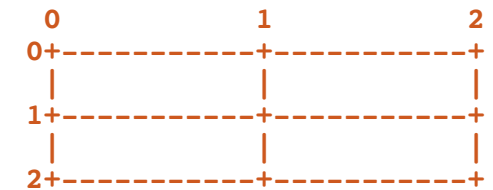
Into the first row, we will place two buttons. One in column 1 and one in column 2. Into the second row, we will place one button spanning both columns. Like this...

```
 0            1           2
0+-----------+-----------+
 | Button 1  | Button 2  |
1+-----------+-----------+
 |         Button 3      |
2+-----------+-----------+
```

To set up a table, we create a table object and add it into the window. The call to create the table is...

```
Table =
gtk.Table(rows=1,columns=1,ho
mogeneous=True)
```

If the homogeneous flag is set to True, the size of the table boxes are resized to the largest widget in the table. If set to False, the size of the table boxes will be dictated by the tallest widget in the same row and the widest widget in its column. We then create a widget (like a button above), then attach that widget into the table in the proper row and column. The attach call is as follows...

```
table.attach(widget,left
point,right point,top
point,bottom
point,xoptions=EXPAND|FILL,yo
ptions=EXPAND|FILL,
xpadding=0,ypadding=0)
```

The only required parameters are the first 5. So, to attach a button to the table in row 0 column 0, we would use the following command...

```
table.attach(buttonx,0,1,0,1)
```

If it were to be placed into row 0 column 1 (remember this is zero based) as button 2 is above, the call would be...

```
table.attach(buttonx,1,2,0,1)
```

Hopefully, this is as clear as mud for you now. Let's get started with our code, and you'll understand better. First the common part...

```
# table1.py
import pygtk
pygtk.require('2.0')
import gtk
class Table:
    def __init__(self):
        self.window =
gtk.Window(gtk.WINDOW_TOPLEVE
L)

        self.window.set_posit
ion(gtk.WIN_POS_CENTER)

        self.window.set_title
("Table Test 1")

        self.window.set_borde
r_width(20)
```

```
        self.window.set_size_
request(250, 100)

        self.window.connect("
delete_event",
self.delete_event)
```

There are a couple of new things here that we need to discuss before we move on. Line 9 sets the title of the window to "Table Test 1". We use the "set_border_width" call to give a border of 20 pixels around the entire window before any widgets are placed. Finally, we are forcing the window to 250 x 100 pixels using the "set_size_request" funciton. Makes sense so far? Now, we create the table and add it to the window...

```
table = gtk.Table(2, 2,
True) # Create a 2x2 grid

self.window.add(table)
```

Next, we create our first button, set up the event connection, attach it to the table grid point, and show it...

```
button1 = gtk.Button("Button
1")
```

```
button1.connect("clicked",sel
f.callback,"button 1")
```

```
table.attach(button1,0,1,0,1)
```

```
button1.show()
```

Now button number 2...

```
button2 = gtk.Button("Button
2")
```

```
button2.connect("clicked",sel
f.callback,"button 2")
```

```
table.attach(button2,1,2,0,1)
```

```
button2.show()
```

Almost exactly the same as button number 1, but notice the change in the table.attach call. Also notice that the routine we will be using for the event handling is called "self.callback", and is the same for both buttons. That's good for now. You'll understand what we're doing in a moment.

Now for the third button. This will be our "Quit" button:

```
button3 = gtk.Button("Quit")
```

```
button3.connect("clicked",sel
f.ExitApp,"button 3")
```

```
table.attach(button3,0,2,1,2)
```

```
button3.show()
```

Finally, show the table and the

window. Also here is the main routine and the delete routine we have used before:

```
    table.show()

    self.window.show()

def main(self):

    gtk.main()

def delete_event(self,
widget, event, data=None):

    gtk.main_quit()

    return False
```

Now for the fun part. For both button 1 and button 2, we set the event handler routine to "self.callback". Here's the code for that.

```
def
callback(self,widget,data=Non
e):

    print "%s was pressed"
% data
```

What happens is that when the user clicks on the button, the click event is triggered, and the data that was provided when we set the event connection is sent in. For button 1, the data that will be sent is "button 1", and for button 2 it is

"button 2". All we are doing here is printing "button x was pressed" into the terminal. I'm sure you can see that this could be a very useful tool when combined with a nicely structured IF | ELIF | ELSE routine.

Now to finish up, we have to define the "ExitApp" routine for when the "Quit" button is clicked...

```
def ExitApp(self, widget,
event, data=None):

    print "Quit button was
pressed"

    gtk.main_quit()
```

And now the final main code...

```
if __name__ == "__main__":

    table = Table()

    table.main()
```

Combine all this code into a single app called "table1.py". Run it in a terminal.

So to recap, when we want to use pyGTK to create a GUI program, the steps are...

• Create the window.
• Create HBox(s), VBox(s) or Table(s) to hold your widgets.

• Pack or attach the widgets (depending on box or table).
• Show the widgets.
• Show the box or table.
• Show the window.

Now we have many of the tools and knowledge to go forward. All code is up on Pastebin at http://fullcirclemagazine.pastebin.com/wnzRsXn9. See you next time.

Greg Walters is owner of RainyDay Solutions, LLC, a consulting company in Aurora, Colorado, and has been programming since 1972. He enjoys cooking, hiking, music, and spending time with his family.

Full Circle Podcast

LibreOffice — The Document Foundation

In this how-to, I will introduce you to the LibreOffice suite, and give you a description of each of the modules in the suite. As we go through the series, I will go into greater detail on each module of the suite, as well as ways to share information between the modules.

LibreOffice is slated as the default office suite for Ubuntu 11.04, and the suite contains several modules that allow you to create text documents, spreadsheets, presentations, and drawings. The suite is multi-platform, and available for Linux, Windows, and Mac OS. The suite allows you to save and read documents in the default Open Document Format (ODF), as well as several versions of Microsoft Office, rich text format (RTF), and DocBook. This feature alone makes it a great suite for both the home and office.
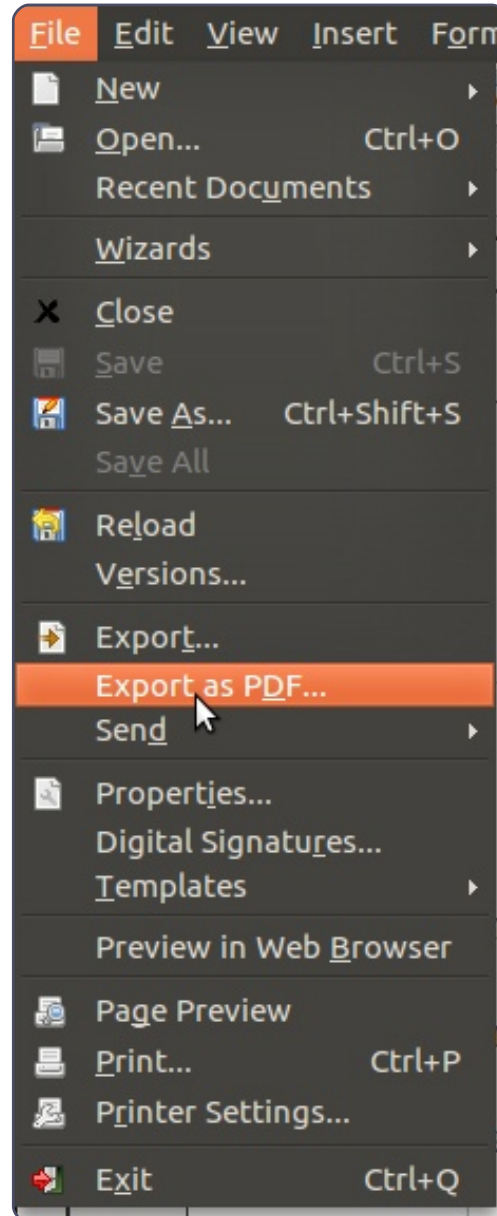
If you can't wait until April to try LibreOffice in Ubuntu, you can install it through the PPA. Type the following three commands in the terminal (Applications > Accessories > Terminal):

```
sudo add-apt-repository ppa:libreoffice/ppa

sudo apt-get update

sudo apt-get install libreoffice
```

The last command may take a while to complete. On my system, I had to completely remove OpenOffice before LibreOffice would install from the PPA. Not sure if this glitch was specific to my machine or if it is impossible to have both installed using PPAs. If you want to use both OpenOffice and LibreOffice, your best choice is to install them manually by downloading them from libreoffice.org and openoffice.org.

[Editors note: *When I installed LibreOffice from the above PPA it automatically uninstalled my OpenOffice.*]

One of the most powerful and useful features of the suite is the ability to create a PDF file from the current document. Exporting to PDF gives you greater control over a finished document. The recipient can easily change a file in the ODF or Microsoft Office format. However, the PDF format makes tampering with the document more difficult. You'll find the Export or Export to PDF in most of the modules under File > Export or File > Export to PDF.

Start LibreOffice from the Applications menu: Applications > Office > LibreOffice. With no documents open, LibreOffice displays the Welcome screen. The

Welcome screen gives you quick access to the different modules in the suite.
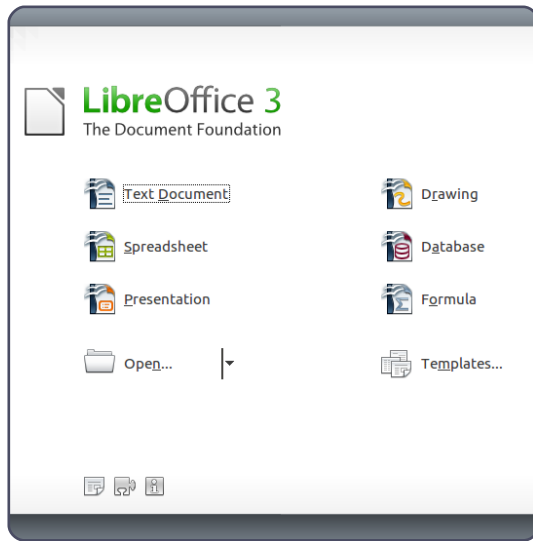
Text Document opens the Writer module. Writer is a full-featured word processor. You can write everything from a simple letter to a novel in Writer. With the Writer word processor, you can create layouts for term papers, technical documents, and product or software documentation. Writer sometimes has problems translating complex layouts from Microsoft Word's format, but, as a general rule, you will not have many problems.

Spreadsheet runs Calc. Calc is compatible with Microsoft Excel. Calc allows you to create a spreadsheet with automatic calculations and formulas. You can also use Calc for data collections to import into other documents. With Calc, you can add graphs and charts to display your data and calculations visually. Tables, graphs, and charts from Calc are easily imported into other LibreOffice documents. Calc gives you the ability to have more than one spreadsheet in one document, allowing you to have monthly, yearly, or other related spreadsheets in one document.

Presentation opens Impress. Impress is LibreOffice's version of Microsoft's PowerPoint, and holds up nicely as a replacement for PowerPoint. Impress has all the features you would expect from a presentation program: slide transitions, object embedding, sound, text effects, graphics, etc. I've used Impress for teaching and seminars, and Impress always gives me what I need.

Drawing starts the Draw module. In Draw, you create simple vector drawings for use in other documents. You could easily make a organizational chart or logo in Draw.

Database opens the Base module. Use Base to create a database using several different engines, depending on what is installed on your system. Base is handy for importing data into other documents in the suite. If you have a need for an inventory or sales database, you might want to give Base a try.

Formula runs the Math module. Have you ever needed to embed a complex math formula into a document? Formula is your answer. Formula gives you the power to create well-formatted formulas for use in other documents. Formula is ideal when you are creating scientific or technical documents, and need to insert a formula or algorithm along with your text.
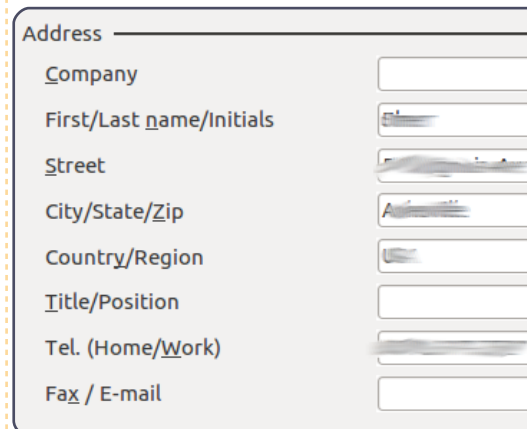
Before we get started with LibreOffice, let's look at some of the options that relate to all the modules. Open Tools > Options > LibreOffice > User Data. Here you enter what personal data you want to make available to the program. You can transfer some of the information entered here into documents. Some of the document properties come from the data in these fields. For example, the document author is taken from the name fields. The program also uses the name fields for revision authors.
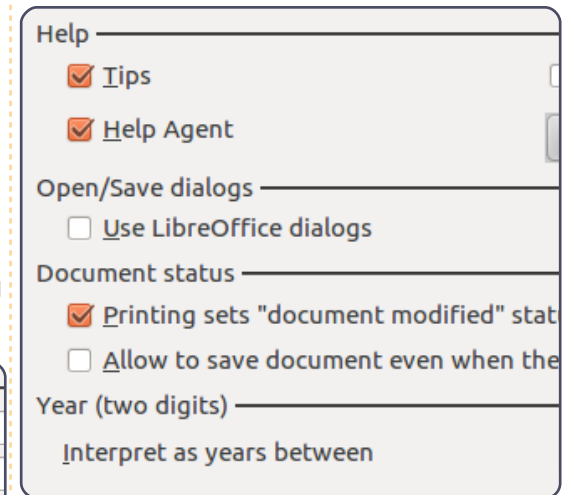
Next, if you are new to LibreOffice, access Tools > Options > LibreOffice > General, and check the Tips and Help Agent. While you learn LibreOffice, you might want to turn on extended tips. Extended tips will show a detailed pop-up balloon for every item as you move the mouse over it.

LIBRE OFFICE - PART 1

Without Extended tips, you get a two or three word description for toolbar buttons only. With the Help Agent on, a help box occasionally displays in the bottom right corner. Clicking on the box will take you to the documentation page for the current task. This greatly speeds up the learning process with LibreOffice. The Help Agent is similar to Microsoft's Office Assistant.

Now, Tools > Options >
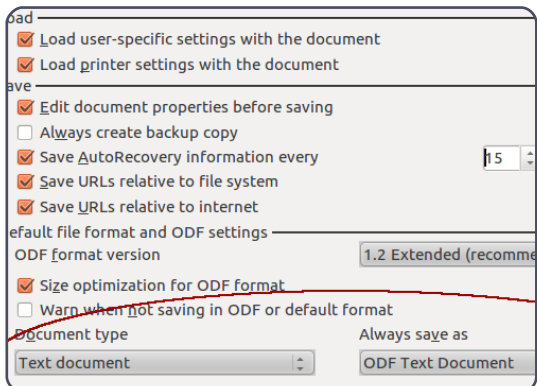
Save/Load > General. Here you can set the default format for different document types. If you need to always save in one of the Microsoft formats, you can set LibreOffice to do this automatically rather than having to always select the needed format. Select the document type from the drop-down box under Document Type. Select the default format for the document type under Always Save As drop-down box.

Under Tools > Options >

Language Settings > Languages, you set up your language options. There is a check box under the language settings that allow you to change the options only for the current document. Language Settings > Writing Aids sets options for dictionaries, spell check, and hyphenation.

Feel free to browse the rest of the options. We will touch on the options for each module as we get to it. Next time, we will begin with the basics of using the Writer module.

> **If you need to always save in one of the Microsofts formats, you can set LibreOffice to do this automatically...**

Anything is possible if you have a tablet.

Except for typing, maybe.

by Richard Redei

I had tried several different approaches to Mythtv, but the only one I had any real success with was Mythbuntu9.04. Sadly, this is no longer supported, so I have had to update to a newer version (when I say update I mean a new from-scratch install to a new 1TB 64MB cached 5400 rpm hard drive). As a long-term-support version, 10.04 would have been the obvious choice, but I went for the 10.10 (64bit). Although this description is very geographically specific, as an example it has application globally.

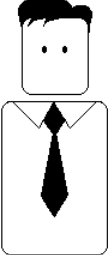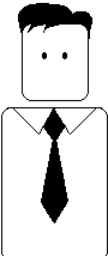My main computer is seldom turned off, so this is where I installed Mythbuntu. However, I do not use this computer for internet banking, credit card transactions, email, or anything where security is a priority (being online twenty-four-seven makes me nervous, and I know little about making it secure beyond using a long random password and a NAT router).

This computer has a 24" monitor driven by an Nvidia EN8400GS, and gets used for audio podcasts, web surfing, word processing, bash script building, Skype, etc., as well as TV watching and recording. I have a normal USA style Logitech illuminated keyboard, plus a marble trackball mouse - both of which are mounted in a sliding under-desk tray. I also have another computer hooked up to the 40" TV in my living room - which also runs Mythbuntu10.10. Thus, I can view digital channels, and record programs, even though the TV has only an analog tuner. This computer utilizes an Nvidia GeForce 7600GS fan-less video card, and a JayCar wireless trackball/keyboard. The only video-out that gave good (actually very good) results on the TV was the DVI (fed to the TV via a $12.00 DVI-to-HDMI cable). Originally, I hooked up a 1280x1024 computer monitor in addition to the TV, but the computer wanted to go with the monitor default resolution rather than the 720p TV (simple fix: don't use a computer monitor at all). This TV mutes all other sound inputs when HDMI is selected, so I used a spare pair of home-built amplified PC speakers fed directly from the built-in TVComputer motherboard sound card (another option would have been a box from JayCar that allows audio to be added to the HDMI, but that costs $99.00). Even though this works well, this TV and TVComputer gets very little use.

The installs were made from the normal CD. I did not enable SSH, Apache, virtual keyboard, or remote controls, nor did I use the proprietary Nvidia driver, but I did use the proprietary "Firmware for DVB cards".

TV Tuner Cards

I used two "Fusion dual digital 4" cards on my main computer. These cards are cheap and work really well with Mythbuntu10.10 if they use the DiBcom 7000PC chip (not so good if they have the Zarlink ZL10353). The computer that feeds the TV has a "Fusion dual digital 4" that uses the ZL10353, and for this and the two other capture cards within the TVComputer (which use Conexant CX22702 and Zarlink MT352 chips), I had to run a channels.conf file (found here: http://www.itee.uq.edu.au/~chrisp/Linux-DVB/channels.conf) before SBS and channel 9 or 10 would

Capture Card Setup

Card type: DVB DTV capture card (v3.x)

DVB Device Number: /dev/dvb/adapter0/frontend0

Frontend ID: Zarlink ZL10353 DVB-T   Subtype: DVB-T

Signal Timeout (msec): 1000

Tuning Timeout (msec): 3000

DiSEqC   Recording Options

tune. I also did a manual scan at 599500000Hz, and 7Mhz bandwidth - with all other qualifiers left at auto (this to get channel 44 i.e. Briz31 which does not appear in channels.conf) - although a "Full scan" would have done as a slower alternative to the manual scan. Dual cards use only the one precious PCI slot (two single-tuner cards need two), tend to be cheaper than two single cards, and generally load the TV signal half as much as two single cards - while eliminating one TV signal splitter with its associated insertion loss.

One of life's minor frustrations is to record a TV program only to find on playback that the recording ends before the program did. Using two dual cards (or one dual and 2 single as in the TVComputer case) means that

Scan Configuration

Video Source:	aerial
Input:	[DVB : /dev/dvb/adapter0/frontend0] (DVBInput)
Desired Services:	TV ▾ ☑ Only
Scan Type:	Import channels.conf

generous overlap of recording times is nearly always possible (I may extend recording time by forty minutes past the advertised finish time on some channels to ensure I don't miss the end of a particular program - even while simultaneously recording two or even three other channels from the total of 23 available here).

Frontends And Backends

In both cases, these computers are self contained (i.e. both frontend and backend within each computer). The backend has the setup functions for the capture cards, commercial skipping, etc., while the frontend allows setup of audio mixer, appearance (including mouse unhide), etc. During normal use, it is the frontend that is run by the user, i.e. /usr/bin/mythfrontend starts mythTV for playback or watching TV. The backend concerns itself with scheduling and capture cards.

Navigating The Setup Pages

Tab and shift-tab go forward or backwards through the page options (the down and up arrows do the same thing).

Esc backs out through menu pages.

Enter on Next, Finish, Scan, or Back selects another page.

In the menu paths below, I give the full path from the Desktop, but mostly you will not need to back out all that way. I do not cover everything here, only items which seem less than obvious. You just have to work through the backend and frontend setup menus (a helpful text appears at the foot of each page as you go through each option with the tab key). Mostly the defaults are what you want.

Scan Configuration

Video Source:	aerial
Input:	[DVB : /dev/dvb/adapter0/frontend0] (DVBInput)
Desired Services:	TV ▾ ☑ Only Free ☐ Test
Scan Type:	Full Scan (Tuned)

Frequency:	599500000	LP Coderate:	Auto
		HP Coderate:	Auto
Bandwidth:	7 MHz	Trans. Mode:	Auto

GUI width (px):	0	☑ Use GUI size for TV playback
GUI height (px):	0	☐ Hide Mouse Cursor in MythTV
GUI X offset:	0	☐ Use window border
GUI Y offset:	0	☑ Use fixed window size

Idiosyncrasies Of Frontend Setup

I like to use the mouse with MythTV, even though it is not fully supported, and the official website discourages it just a bit. If you wish to enable the mouse, go to Applications > Multimedia > MythTV frontend > Utilities/Setup > Appearence > Next > Hide mouse cursor in Myth TV, and untick. If you go through the setup quickly, it's easy to miss this.

Idiosyncrasies Of Backend Setup

General:

I only use free-to-air TV. In Brisbane, Australia, a couple of different PAL standards are employed; however, just setting PAL works fine, i.e. Application > System > MythTV Backend Setup > Setup > General > Next > TV format: PAL. On the same screen, the Channel frequency table is changed to "australia" (sic, the "a" is lower case) since I live there.

Capture Cards:

For my tuner cards, I did the following; Applications > MythTV Backend Setup > Setup > Capture Cards > New Capture Card > Card type: DVB-DVT Capture Card (3.x). To do this requires down arrowing (twice) to "Capture Cards", then right arrowing to select the correct card type. Tab to "Finish", and Esc back to the setup menu.

Video sources:

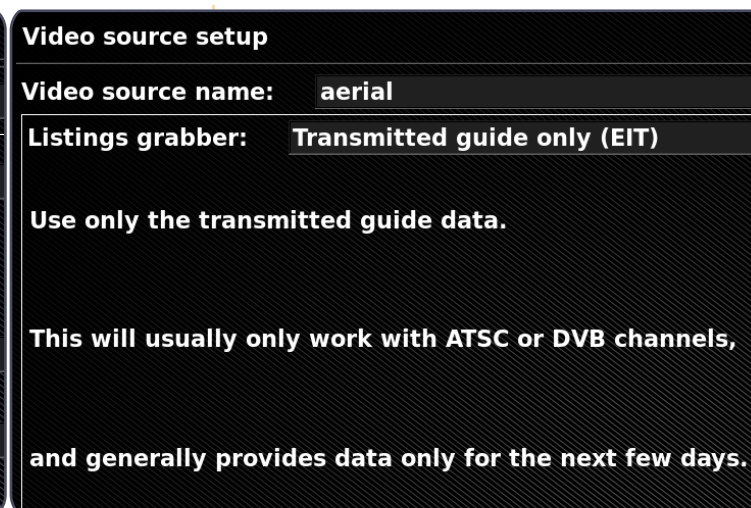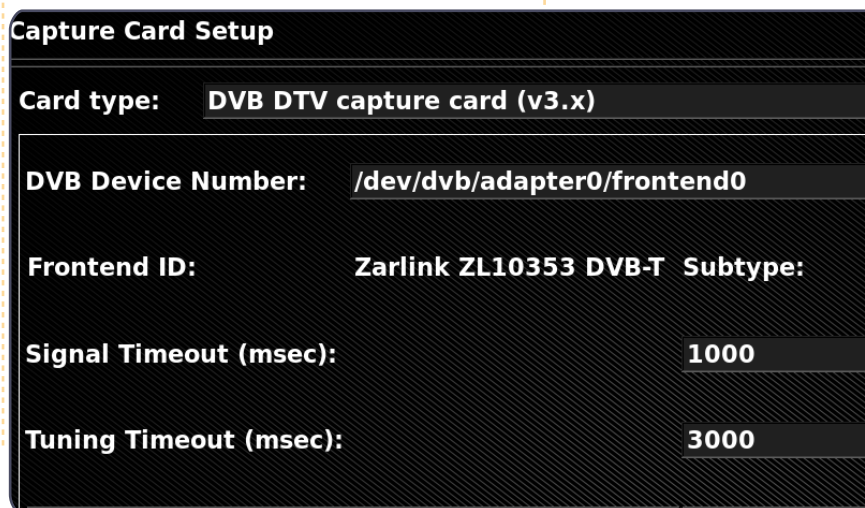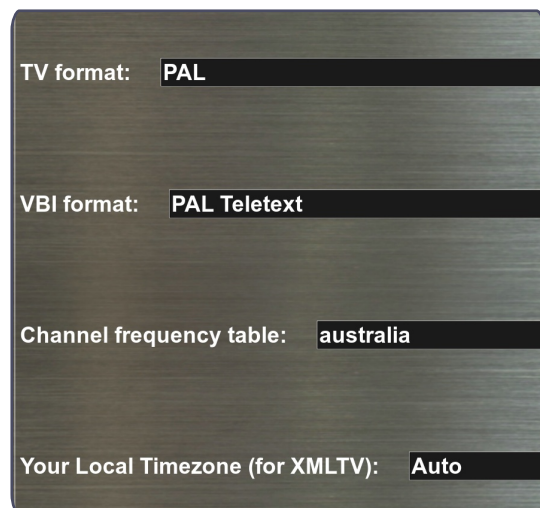Applications > MythTV Backend Setup > Video Sources > (New Video Sources) >

I add the Video source name "aerial", then change the "Listing grabber:" to "Transmitted guide only (EIT)". "Finish" saves these changes. I find it easy to get stuck in this screen (unable to reach "Finish"). If you get stuck, using shift-tab a few times might get you free and on to the "Finish" button (or you could just use the mouse if it has been unhidden).

Input Connections

This menu selection enables the tuning of the capture card(s), i.e. Applications > MythTV Backend Setup > Input connections. Although optional only if I have more than one card (or a dual card), I like to give them a name here - e.g. Card00, or Card10, etc. This next step is required if you are to tune your card, i.e. select the Video source: e.g. aerial.

Now you can tab down to Scan for channels, and then hit Enter.

From here we will be doing a full scan. Enter on next. When the scan is complete, just enter your way through the defaults and finally escape out.

TV format:	PAL
VBI format:	PAL Teletext
Channel frequency table:	australia
Your Local Timezone (for XMLTV):	Auto

Capture Card Setup

Card type:	DVB DTV capture card (v3.x)
DVB Device Number:	/dev/dvb/adapter0/frontend0
Frontend ID:	Zarlink ZL10353 DVB-T Subtype:
Signal Timeout (msec):	1000
Tuning Timeout (msec):	3000

Video source setup

Video source name:	aerial
Listings grabber:	Transmitted guide only (EIT)

Use only the transmitted guide data.

This will usually only work with ATSC or DVB channels, and generally provides data only for the next few days.

The Input connection procedure needs to be performed for each capture card.

Try going into Applications > Multimedia > MythTV frontend > Watch TV, and checking if all (or any) channels work. Some of the capture cards I have tried do not get all the channels at first, e.g. 1, 2, 7, 10, 11, 12. etc. (missing 3, and 9). However, if the channel changing was done in the reverse direction, e.g. 10, 9, 7, 3, 2 and 1, the missing channels would work. In such a case (actually in all cases), I set Applications > MythTV Backend Setup > Input connections > Use quick tuning:always. This fixes those problems (I find this very counter intuitive, but it works for me).

The Magic Of The Letter 'd'

Apart from superficial appearance, and the odd paper cut fixed, the menu system has changed little since 9.04 and it works very well once you have learned where stuff resides (except the minimum record time is now 5 minutes rather than 1 minute which is five times as

painful when doing test recordings). One key you should know about is the d key. Pressing d deletes (the delete key does not). This works while viewing a recording or in a menu. For instance, if used in the "Upcoming Recordings" menu, you may delete all future scheduled recordings of a highlighted program. It even works in the capture card list, allowing capture cards to be deleted individually (very handy when trying to diagnose strange behavior if using more than one capture card). In the channel edit menu, it allows the removal of unwanted individual channels. The only downside I can see to using d is that not deleting a recording via the Applications > Multimedia > MythTV frontend > Manage recordings > Delete recordings route does mean you will not get an indication of how much space is left on the drive - which could be bad if you have a small hard drive. Under Mythbuntu9.04, running out of space on the dedicated TV recordings partition meant just losing all recordings. Mythbuntu10.10 uses one partition for everything; what happens if this fills up I don't know.

Useful keys:

Esc	backs out of everything/anywhere, one step per key press.
Enter	selects a highlighted menu item.
f	allows left/right arrow keys to control volume level.
\	mutes the sound.
a	allows left/right arrow keys to slow or speed up playback.
End	move fwd 24hrs in program guide (Shift-End for -24hrs).
Rt arrow	move 30secs fwd in playback and live TV (if possible).
Lt arrow	moves 10 seconds back during playback and live TV.
p	pauses playback (hit it again to resume).
Numeric	select channel when watching live TV (Enter selects*).

Up/Down keys: browse channels in order when watching live TV (Enter selects*).
Up/Down keys: move 10mins back/forward when watching recording.

* if Enter is required depends on setup.

OK, enough for this issue. I have not covered shortcut icons for opening MythTV frontend, terminal, volume/mixer, etc., or adding Skype, Open Office, a printer, screen rotate, K3B and gedit, etc. Nether have I covered the configuration of special keyboard keys for controlling sound inside and outside of Myth (mine worked by default until I installed Skype, then I had to fix them). Nor have I dealt with doing a full hard-disk backup, or getting the marble mouse to behave. If anyone wants it, I am happy to do a followup on these.

Guidelines

The single rule for an article is that **it must somehow be linked to Ubuntu or one of the many derivatives of Ubuntu** (Kubuntu, Xubuntu, Lubuntu, etc).

Write your article in whichever software you choose. I would recommend OpenOffice, but **PLEASE SPELL AND GRAMMAR CHECK IT!**

Writing

In your article, please indicate where you would like a particular image to be placed. Please do <u>not</u> embed images into your Open Office document.

Images

Images should be JPG with low compression.

Regarding image sizes: if in doubt, send a full size screengrab and we will crop the image.

If you are writing a review, please follow the guidelines shown here.

For a more detailed list of the style rules and common pitfalls please refer to: https://wiki.ubuntu.com/UbuntuMagazine/Style - in short: US spelling, no l33t speak and no smilies.

When you are ready to submit your article please email it to: articles@fullcirclemagazine.org

If you can't write articles, but hang out in Ubuntu Forums, send us interesting forum threads that we could print.

Non-English Writers

If your native language is not English, don't worry. Write your article, and one of the proofreaders will read it for you and correct any grammatical or spelling errors. Not only are you helping the magazine and the community, but we'll help you with your English!

REVIEWS

Games/Applications
When reviewing games/applications please state clearly:

• title of the game
• who makes the game
• is it free, or a paid download?
• where to get it from (give download/homepage URL)
• is it Linux native, or did you use Wine?
• your marks out of five
• a summary with positive and negative points

Hardware
When reviewing hardware please state clearly:

• make and model of the hardware
• what category would you put this hardware into?
• any glitches that you may have had while using the hardware?
• easy to get the hardware working in Linux?
• did you have to use Windows drivers?
• marks out of five
• a summary with positive and negative points

You <u>don't</u> need to be an expert to write an article - write about the games, applications and hardware that you use every day.

Back in the early days, the first electronic computers were operated only by large corporations and government departments. Programs and data could only be loaded straight into memory because that was all the storage there was. Then engineers got smarter, using stacks of punch-cards and paper tape for programs and output, followed by half-inch magnetic tape: all serial-access storage.

Then some bright spark devised a magnetic disk drive, a Direct Access Storage Device (DASD) which could read and write to random locations, which is why they needed a file-system to organize the data and support a Disk Operating System (DOS). Move on a few years and enter the personal computer. When IBM needed a file-system and some way to access it, we got the Microsoft Disk Operating System, MS-DOS. Not the first or only DOS, but it was the main player for home PCs - like it or loathe it, DOS was the one you used.

Move on a few more years and find the smart computing set (that's us) using open-source software. Accept the default settings in most installers and get the default file-system. Do anything else and your first issue with Linux is choice. Which one do you choose?

Dear Diary...

Or more precisely, "Dear Journal..." Most modern file-systems employ journaling. Think of it as a low-level activity log. A file to be updated is first written to the journal, clocked-in, written to disk when ready, cleared from the journal, and clocked-out. If there's any interruption to the normal running of the computer – power cut, catastrophic crash – while the file is being written to disk, the file system has the journal entries for all operations not yet completed. If all is good, the operation can complete, if not, there is a log to aid file recovery.

Journaling pays a small disk/processor overhead for added data security. Some file systems reduce the overhead by not writing the full file to the journal, so you will see references to file meta-data, inode, or disk location in their journaling scheme.

Other crucial features of a decent file-system include consistent access controls (permissions, or authorities, according to your school), aliasing and symbolic links – multiple pointers referencing a single copy of a file.

Ext Family

We're on the fourth iteration of Ext, or Extended file-system. The original Ext is practically defunct, so Ext2 is the lowest version you'll see in general use. It's a non-journaling file-system, so it's fast, but not as secure as its successors. Since it writes less to disk (and erases less), it remains a good choice for flash memory, USB-sticks, and SD-cards - which have limited lifetime write-limits.

Ext3 and Ext4 remain backward compatible with Ext2, with the addition of journaling. They have years of optimizations to improve performance and data security, which is why Ext3 took off with large databases, but not with servers, and why Ext4 finally scored as a good all-rounder. Ext4 has many major improvements over Ext3, like larger file-system support, faster checking, nanosecond timestamps, and journal verification using checksums. It employs a technique known as delayed allocation to reduce file fragmentation, and means it can be used on flash memory and solid-state disks (SSD); however, delayed allocation has potential for data loss. I use Ext4 on all my desktop, laptop and external hard-drives with a noticeable increase in performance over Ext2 and Ext3. Ext4 is robust and efficient, but lacks some advanced features - such as support for disk snapshots and advanced scalability. Enter the next two contenders...

Rise and Fall

ReiserFS represented a radical leap forward in 2001, including many features that Ext still cannot implement. Reiser4 improved or completed many features over the initial release. Development since 2004 has been very slow, and remains under a cloud after the personal 'legal difficulties' of the original developer, Hans Reiser. Reiser4 is not supported in the main Linux kernel.

That said, ReiserFS performs well on systems requiring many small writes - say logs and indexes - such as in databases and email servers.

Better and Better

BTR-FS - can we agree to pronounce it 'better' and not 'butter'? I don't like butter, and I prefer not to keep my files on something slippery that's likely to melt - thank you! It stands for B-tree File-system, originally developed by Oracle (watch those licensing terms, open-source fans!). Having similar features to ReiserFS, it trades heavily on enterprise-level features such as drive pooling, on-the-fly

snapshots, transparent compression, and on-line defragmentation. All the major Linux distros plan to adopt it as the default file system eventually; however, you can't currently use it on a boot partition, only data partitions, so it isn't ready for exclusive use yet. Current performance benchmarks show it slightly slower than Ext4 in many uses, so bank on big-database vendor Oracle fixing that in the next couple of versions. The documentation makes it explicitly clear: it is "not suitable for any uses other than benchmarking and review."

Best of the Rest

If you can't get enough of file-system acronyms, there's a gaggle of niche flavors which still appear in Linux installers and disk tools:

XFS, from Silicon Graphics: much like Ext, good for large files, but not small; so render-farms and video processing good, databases and email not so good. If you need guaranteed data throughput rates, on-line resizing, built-in quota enforcement, and support for file-systems up to 8 exabytes in size, you can find XFS as an install option on many popular Linux distributions. You can tune your system to use variable block sizes, like a sliding scale for efficient use of space or high read-performance.

JFS, from IBM: showing its age now, but a good performer in its day on small drives and files. Find it on older hardware.

ZFS, from Sun Microsystems: think of it as the granddaddy of BtrFS.

SWAP Shop

Swap isn't itself a file system. This is virtual memory without a file system structure, used only by the kernel to write memory pages to disk. It's your swap-file or paging-file for when you run out of physical memory or when you set your computer to hibernate. You won't get through a Linux install without a Swap partition.

No Country for Old File-systems

That's the round-up, in no technical depth whatsoever of the common file-systems on our 'home-turf'. If you have the stamina, part two will take you over the border to foreign lands where the file-systems wear funny logos and speak in strange tongues. But if you want to be the Ambassador of Open, or the Emissary of Interoperability, you'll need to recognize the other tribes...

MY STORY

Written by Célio Ricardo Quaio Goetten

I am an electrical engineer working with industrial automation. My experience with Linux started almost ten years ago when I tried a version of Mandrake Linux (now Mandriva). At that time, it was very difficult to get the operating system running. A serial mouse didn't work out of the box, configuring an analog modem with PPP protocol was a terrible experience, and installing a new program was very difficult.

But three years ago I happened upon a magazine that contained a CD of Ubuntu 8.04. I decided to give Linux another chance! I was pleasantly surprised with Linux's evolution. The installation process was very easy with the live CD - with a graphical interface (no more text interface!), and almost everything worked out of the box (network, video, sound, and Internet).

The software packages that came with Ubuntu also surprised me; the default installation already had all the software that I needed (IM, Internet browser, office tools, P2P software, music and video players, etc.), and all were legal and free. Moreover, the installation of new applications was very easy - from the repositories or by downloading a .deb file.

It´s important to say that in Brazil, legal software is very expensive, and almost 60% of all domestic software used in Brazil is, unfortunately, illegal. It represents losses of roughly $2.21 billion.

Today, I'm using Ubuntu for 100% of my home tasks, and my mother and fiancé have been using Ubuntu for almost a year. One day I got a very nice surprise when my fiancé said: "This Win7 is very confusing. I prefer to use Ubuntu"!

Nowadays, I'm using Ubuntu to create intelligent devices with open-source hardware, such as Arduino. There are many experiences and devices that you can create by yourself with inexpensive hardware and free software. Imagine controlling your home appliances from an Internet browser, or storing data from your home - such as temperature, pressure, and electrical power - in a web server. But these matters are a theme for another edition of FCM!

There was a time when I had to see a shrink to get things out of my chest.
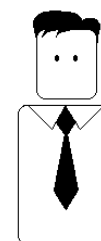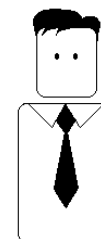
But today I can simply put them on Facebook.
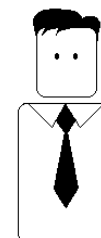
by Richard Redei

Boxee Box

You've probably heard the mumblings about a funny square set-top device that looks like it has partially slipped between two dimensions, and the majority of it landed in this one.

The Boxee Box is that product. But what is it? The Boxee Box is an Internet-enabled Social-Multimedia Set-Top Box. Before we get to the social portion, let's look at what it actually does.

Out of the box (so to speak), you have support for streaming internet video and audio from YouTube, Wired, Revision3, OpenUniversity, Pandora, Vevo, and many, many more. Premium sites like VuDu and Netflix are promised by end of year, and Hulu Plus is also in the works . It also handles local media such as .AVI, M4V, MP3, and others. It also displays pictures from your local network. The social part allows you to share your thoughts on music and videos with friends, and them with you.

Setup is a breeze, or at least it was for me. Turning on the Boxee Box, it automatically finds your local network, and, if it's protected, asks for the appropriate credentials. From there, you set up your local media shares, and, after a little while, you have access to all your shared media.

The Hardware

The Boxee Box features an Intel Atom processor, 802.11n wireless and 10/100 wired networking, 2 USB ports, SD Card slot and HDMI 1.3a output, S/PDIF for digital audio, as well as stereo RCA audio ports and an RF Remote (more about the remote in a moment). There's no local storage in the box, but you can use the USB ports for external USB drives.
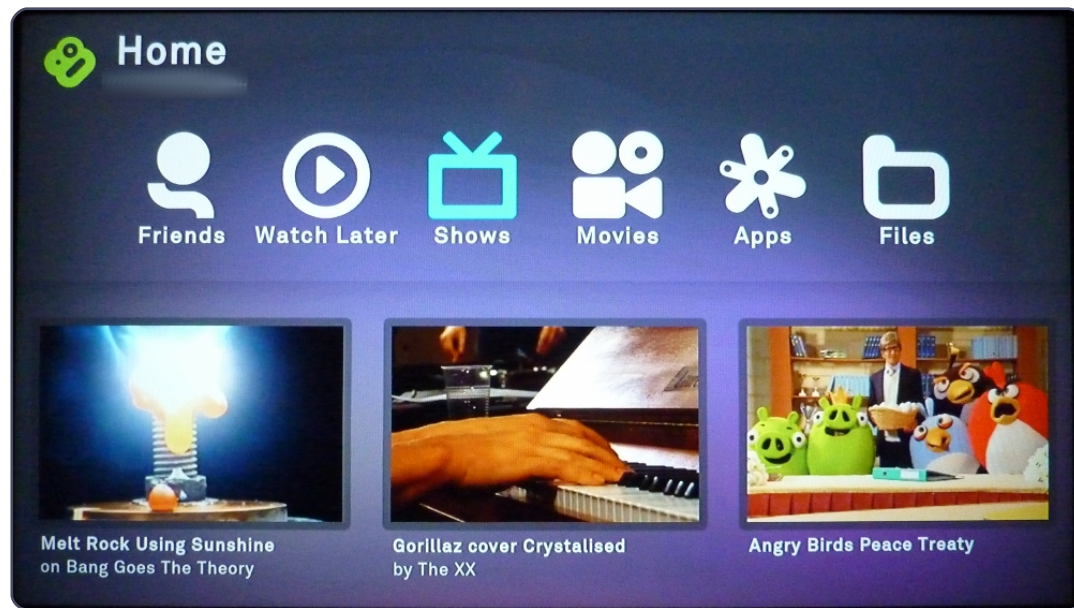
The Remote

On one side, you have a few buttons (play/pause, menu, select, and a 4-way navigation block). On the other side you have a QWERTY keyboard.

My 16-year-old stepson, when he first saw the remote, said "This is AWESOME!". While some people have had problems with accidentally pressing the buttons

on the other side while trying to type on the keyboard, that hasn't been a problem for me. The good part is that you don't have to worry about pointing the remote at the box while typing since it's RF.

In Use

Using the Boxee Box is fairly easy. When the B-box starts up, you get a pretty user friendly home screen.

From here you can navigate to Shows (Internet TV-Shows), Movies (Internet based), various Internet applications, and local files such as locally stored pictures, music, and other local media files.

The applications section contains more than 100 applications to deal with various Internet-based sites, with more coming as time goes on.

From this quick screenshot from my TV, it's rather difficult to tell just how much there is. However, you can see that things like Pandora, Revision3, TWIT, YouTube, and more, are included. Many of the same applications are available for the PC (Windows, Linux, and Mac) version, and most of them work directly on the Boxee Box. However, there are a few that have some problems.
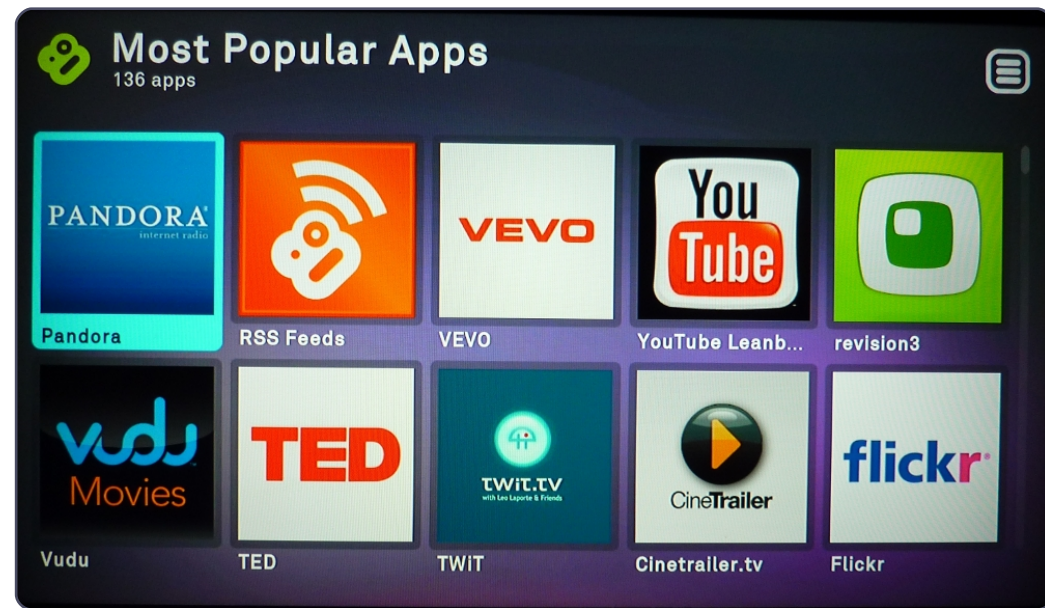
Those are being worked on by the developers who wrote them. Many more applications are being worked on, and I truly believe that, in the next few months, there will probably be over 200.

The Good

I've been on the Boxee alpha and beta teams for over two years for PCs. I must admit, as soon as I started working with Boxee, I was hooked. All of my local video and audio played correctly right away. The box is more than capable of playing Internet and local-based media. A few bugs have been found and are in the process of

being squashed right now. I've had only one or two extremely large local files that seem to have a problem transferring over my 802.11g network. The support structure for the Boxee Box, and Boxee in general, is wonderful. The forums have many knowledgeable people who try to help, and the moderators have the ear of the developers.

The Bad

OK. I'm torn by how to talk about this, since there are a lot of changes due out before this review sees the light of day. When the box was actually released,

there was a user interface that was a close clone of the ones that longtime Boxee users were used to. Almost immediately, a 1.0 version of the firmware came out, and the UI was completely changed. NOT for the better. The local media was placed multiple menus away from the main screen, and the Internet media was placed in the forefront. This caused a large backlash from users who had gobs of local media. Local pictures and music also took a back seat to the video portions of the box. The issue was heard by the Boxee team, and a change in the UI is in the works. There have been other issues including networking, media playback, and much more. However, the Boxee team jumped on the

issues and have (as of this date) already released two patches, and more are due out soon. As I'm writing this review (early January 2011), Vudu is out with another firmware update any day, and Netflix is due out by the end of January 2011. Both of these were

due to come out before the end of 2010, but things held both of them back. This has caused a number of early purchasers to regret their early purchase.

The Bottom Line

The Boxee Box is a wonderful device which has a huge amount of promise. Yes, today it has some (many) issues, but they are being actively addressed, and for the most part, won't keep one from enjoying local content and some Internet streaming. Many issues have been discovered and are also being worked on.

Should you buy one? I'd say yes. Just how soon you should wait until purchasing one is dependent on the amount of local content you have and how soon you would need to have premium (Netflix, Vudu, etc.) content. If someone were to hold my feet to the fire, I'd have to say wait until the first part of March - unless you have a ton of local content.

Greg Walters is owner of RainyDay Solutions, LLC, a consulting company in Aurora, Colorado, and has been programming since 1972. He enjoys cooking, hiking, music, and spending time with his family.

LETTERS

Every month we like to publish some of the emails we receive. If you would like to submit a letter for publication, compliment or complaint, please email it to: letters@fullcirclemagazine.org. PLEASE NOTE: some letters may be edited for space.

Missing Info

A critical piece of information is missing in the response the Q & A question writer's solution regarding Windows 7 access error. In my Window 7 64 bit Home Premium installation, there is no option: Local Security Policy. To what version of Windows 7 was this solution directed?

Chris Warren

Gord says: *"Local security policy" sounds like Enterprise to me, but I don't have access to a computer with Enterprise installed to check this. If Mr. Warren doesn't have Enterprise, I would doubt he got the error code 0x800070043.*

When someone suggests a solution, and another says it worked, I seldom check it. In this case, I couldn't.

OSX > Ubuntu

I switched away from Windows back in 2000, when Apple finally released an OS better than their outdated MacOS 9 - OS X (10.1). I also tried different Linux distros then, but none of them were ready for my daily use as home PC or laptop OS.

Last year, I also left Apple's OS behind. At my new office we use Ubuntu, so I sold my expensive MacBook Pro, on which I gave Ubuntu a test run in VirtualBox. OS X is, of course, a polished, reliable, and comfortable system. But, in my opinion, it became too commonplace, and I became interested in open source in both using and contributing to it.

I expected no smooth transition. ITunes and iPhoto after all are excellent applications, but now I make do with Rhythmbox and photo software like DigiKam just fine.

Igor Barinov

A PLEA ON BEHALF OF THE PODCAST PARTY

As you'll hear in episode #15 of the podcast, we're calling for opinion topics for that section of the show.

Instead of us having a rant about whatever strikes us, why not prompt us with a topic and watch for the mushroom clouds over the horizon! It's highly unlikely that the three of us will agree.

Or, an even more radical thought, send us an opinion by way of a contribution!

You can post comments and opinions on the podcast page at fullcirclemagazine.org, in our Ubuntu Forums section, or email podcast@fullcirclemagazine.org. You can also send us a comment by recording an audio clip of no more than 30 seconds and sending it to the same address. **Comments and audio may be edited for length. Please remember this is a family-friendly show.**

It would be great to have contributors come on the show and express an opinion in person.

Robin

Elizabeth Krumbach: Please tell us a little about yourself.

Valorie Zimmerman: I'm a writer, wife, mother, and grandmother. Besides Free software and Free culture, social justice is important to me. Right now I'm finishing up the Amarok Handbook, which has been a year-long project. During the recent Google Code-In project, I mentored quite a few teens who helped finish writing and illustrating the manual. That was a great experience! I enjoy music, reading, art, genealogy research, traveling, making new friends, and helping people in IRC.

EK: What inspired you to get involved in the Ubuntu community?

VZ: I've been a member of LinuxChix for many years, and there I heard about the Ubuntu-Women project. Since I was using Kubuntu at the time, I was encouraged to hear that Ubuntu as a whole was welcoming and encouraging participation from women and other F/OSS minorities. And when I found out that there was a local group, the Washington State LoCo, I wanted to help out there also. When I volunteered for Amarok documentation, that involved adding Freenode to my list of IRC servers, so I joined the #ubuntu-women channel, and it's the first one I read every day.

EK: What inspired you to get involved with the Amarok project?

VZ: I had some time, and thought for a while about how I could contribute to F/OSS and have fun at the same time. Since music brings so much joy to my life, and Amarok is my favorite app ever in my almost 30 years of using computers, and they needed a new handbook, I volunteered. Rather than being ignored, I was welcomed, introduced around, and treated in a very friendly way. Even though my dad broke his hip a month or two later, and that resulted in putting the Handbook on hold for many months, I never met with impatience, but just helpful attitudes all around. It's been a grand experience, and that gave me the confidence to ask for Kubuntu membership and volunteer to help on documentation there too.

EK: What are your roles within the Ubuntu community?

VZ: Since I became a Kubuntu member, I attended UDS-N (thanks for sponsoring me, Canonical!), have been continuing my participation in my LoCo, and participate in the #ubuntu-women and #ubuntu-women-project IRC channels. I hope to help with Kubuntu documentation in the next few cycles.

EK: Is there anything you haven't done yet, but would like to get involved with in the Ubuntu community?

VZ: I thought I would already be learning how to do Kubuntu documentation by now, but still hope to ramp up my contribution there in the future. Once I learn how to do it, I hope to get others involved and trained also, since documentation is so valuable to the project. While my LoCo seems to be in a down phase right now, I hope to stand for leadership as part of a leadership team, following the excellent example of LinuxChix and Ubuntu-Women. I would like to see more Ubuntu activity all over the state of Washington.

EK: What other things are you interested in outside of open source and Ubuntu?

VZ: I love the growth of free culture around the world, where people share their art, music and other creative endeavors freely. I'm active in PFLAG (http://pflag.org) where we work on social justice for gay, lesbian, bi and trans people. One of my kids is gay, so this is very close to my heart.

It's back! The Humble Indie Bundle 2 kicked off just before Christmas. It seems only a short while since the last Humble Indie Bundle was out. Five Indie games were in the bundle, and again you could pay what you wanted! Braid, Osmos, Revenge Of The Titan, Cortex Command, and Machinarium were included. I will be reviewing three of these in the next few issues of Full Circle. I won't review Osmos, because I reviewed it in Issue 37, and I won't review Machinarium, since it is a Flash game.

To kick off my review series of the Humble Indie Bundle 2, I'll begin with Braid, a very popular Indie, originally released in 2008 on Xbox 360. You play as Tim who attempts to rescue a princess from a monster. The story is told through text passages that explain Tim's contemplations and motivations.

Braid is essentially like any other platformer; it involves moving from one side of the 2D map to the other, jumping across platforms, and dodging enemies. It has, however, a few especially nice gameplay mechanics. It requires you to collect puzzle pieces throughout the world and to put them together to create a completed jigsaw. There are 6 worlds to explore and find the puzzle pieces within.

An interesting mechanism is that Tim can go back or forth in time. This will help you if you die or must dodge enemies, and certain puzzles require you to use this technique. Another unique feature is being able to travel through all the different worlds of the game without collecting the puzzle pieces for the jigsaw that completes that particular world. So, if you get stuck at a certain point, you can carry on and enjoy the rest of the game.

Braid has only the single-player story, which you are dropped into immediately. It has no menus to speak of. The story should take no more than 6 hours to fully complete, but there is plenty of replay value to discover every bit of this great game.

Braid has beautiful artwork, consistent throughout all the different levels you play and interact with. It is always nice to see different art styles come through from Indie games. In Braid, bright colours in a watercolour style are a welcome change from the brown/gray game worlds we usually play. The soundtrack is also of high quality, and the score adapts well when you speed forward or backwards in time.

Braid is a fabulous Indie which has been very successful on every platform it has been released on, and it is a great honor that it is finally on Linux. The story, gameplay, atmosphere, graphics, and sound are excellent, a new level of detail for an Indie game. Braid is a platform at heart, requiring you to move through different worlds, collecting items and dodging enemies; but it brings some interesting new mechanisms to keep the game fresh and intriguing. The time travelling feature makes Braid easier by stopping you from failing, but it also adds challenging puzzles into the mix. Braid is an enjoyable platform with a high-quality polish, and it's definitely worth playing.

Score: 9/10

Good:
• Excellent Level design
• Interesting gameplay features
• High quality art style and musical score

Bad:
• Time travelling makes the game impossible to fail

Q&A

Compiled by Gord Campbell

If you have Ubuntu-related questions, email them to: questions@fullcirclemagazine.org, and Gord will answer them in a future issue. Please include as much information as you can about your problem.

Q I want to install the Gimp help documentation locally. I installed the gimp-help-common package, but this did not enable the help function in the program itself.

A Install *gimp-help-en* (or the version for your language).

Q In Maverick, a theme installation failed. Now when I try to click on certain "places", including home itself, instead of opening the folder in Nautilus, the appearances and preferences dialogue box opens with an error.

A (Thanks to *Frogs Hair* in the Ubuntu Forums) Right click the desktop and select "create folder." Right click the folder and select "open with another application." Select file browser from the menu, and make sure "remember this application" is checked. Remove the folder when finished.

Q Does anyone know how to get this video to display in Ubuntu? Link:

http://www.cbsnews.com/video/watch/?id=7166315n

All I get is a black empty box both in Firefox and Chromium.

A Disable AdBlock Plus and Noscript to get this to play in Firefox.

Q How can I listen to a radio station which streams its programming in a Windows Media format?

A Go to Administration/Software Sources and enable the Medibuntu repository. Close that and run Administration/Synaptic Package Manager. Click on "reload", then install the "non-free codecs". Start Firefox, and you should be able to listen to the streamed audio.

Q When I try to install Ubuntu, it gets part way, then just stops with a blank screen and the cursor blinking in the top-left corner.

A There are two articles in the Ubuntu Community Documentation which might help you. The first describes common problems with the CD, and is called BootFromCD. The second describes using boot options to handle quirks with your hardware, and is called BootOptions. From what I have seen, one or the other of them solves at least 90% of the problems.

Q I was installing Ubuntu 10.10, and got to the screen when it asks, "who are you." I filled in all the fields, but the "forward" button was still not available.

A Make the computer's name and username lower case.

Q I have Windows 7 and Ubuntu 10.10 installed, and I want to add Opensuse. What should I do about Grub?

A During the Opensuse installation, tell it not to install Grub, because it includes the old version. After the installation, boot into Ubuntu and issue the Terminal command:

```
sudo update-grub
```

You will need to run it again every time an Opensuse update installs a new version of the kernel.

Q & A

Q **Normally, I opened Spotify (the Linux version) and once I selected the music, I closed the window. Spotify kept on running in the background and a small icon appeared in the Panel at the top. Whenever I wanted to open Spotify, I would click the small icon at the top.**

Now, Spotify keeps on running in the background, but the icon doesn't appear. When I click the regular Spotify icon on my desktop, it says Spotify is already running and won't open. Now, I can't open Spotify any more. All I can do is kill the process and completely restart the program. A similar scenario happens with Skype.

A Right-click on an open area of the Panel, select "Add to panel", scroll down in the list and select "Notification Area". The icons for Spotify and Skype will then appear when you run those programs.

Q **I installed an OCZ Vertex 2 Solid State Disk in my Thinkpad T61p. The maximum transfer rate I am getting is 136.63 MB/sec. Why isn't it a lot faster than that?**

A That computer uses a SATA 1 connection for the hard drive. Your transfer rate is limited by the SATA 1 speed.

Q **Where can I find Ubuntu 10.10 for the PowerPC?**

A http://cdimage.ubuntu.com/ports/releases/10.10/release/

Q **I have an ATI Radeon 9200 video card driving an LG M237WA monitor via an RGB cable. I cannot see the background, it appears in black. Also, when I maximize the windows I get a black screen too.**

A Turning off "Visual Effects" solved the problem.

Q **Resizing windows by grabbing window borders is difficult.**

A Switch to the Dust or Clear Looks theme. System/Preferences/Appearance, then select the Clearlooks theme instead of the default, Ambiance.

Q **I just installed Ubuntu and the maximum screen resolution I'm getting is 1280x720. When I used windows it was 1400x900 or something like that. I installed the Nvidia drivers and now my screen sits low and to the left.**

A Open a terminal and type:

```
sudo nvidia-xconfig
```

Reboot and then open System > Administration > Nvidia X Server Settings to set your resolution.

Q **Is it possible to use the audio optical output on my computer?**

A Open Accessories/Terminal and enter this command:

```
alsamixer
```

Use the arrow keys to go across to the optical out, S/PDIF. Then use "m" to toggle it on and off.
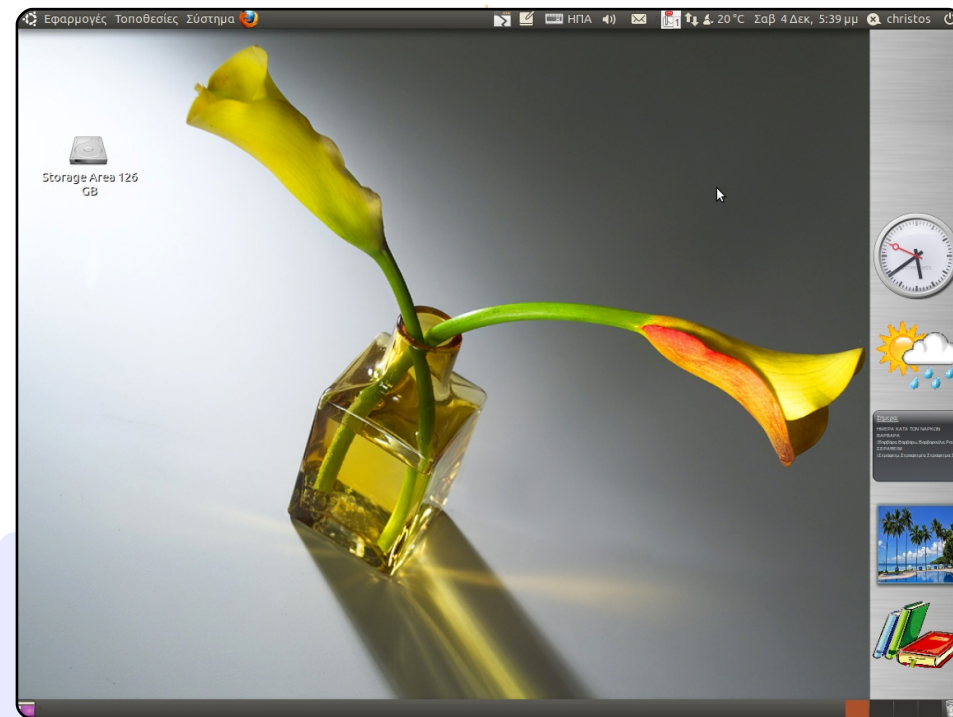
MY DESKTOP

I made some fissions in my laptop's desktop with AWN settings, screenlets, and cairo dock, along with the wallpaper provided with Ubuntu 10.10.

I am using a Toshiba L3100 laptop, Intel Centrino processor core 2 duo, 120GB hdd, 2GB DDR2 RAM. I wanted my desktop theme to have an environmental touch, so I selected a light green color and named the desktop "PATCH OF EARTH".

Anish Chirakkal Aravind

The wallpaper changes every time I open the computer (I use the program DesktopNova for that). In the sidebar I have the following screenlets: Clock, weather forecast, a screenlet that tells you the religious feast of the day (eg: St. Barbara), and the International day , a slideshow frame connected to Flickr, and a launcher to the e-book I read.

My PC specs:
CPU Intel Core Duo 2GB, 2.83 GHz, Ubuntu 10.10, HDD 250 GB

Christos
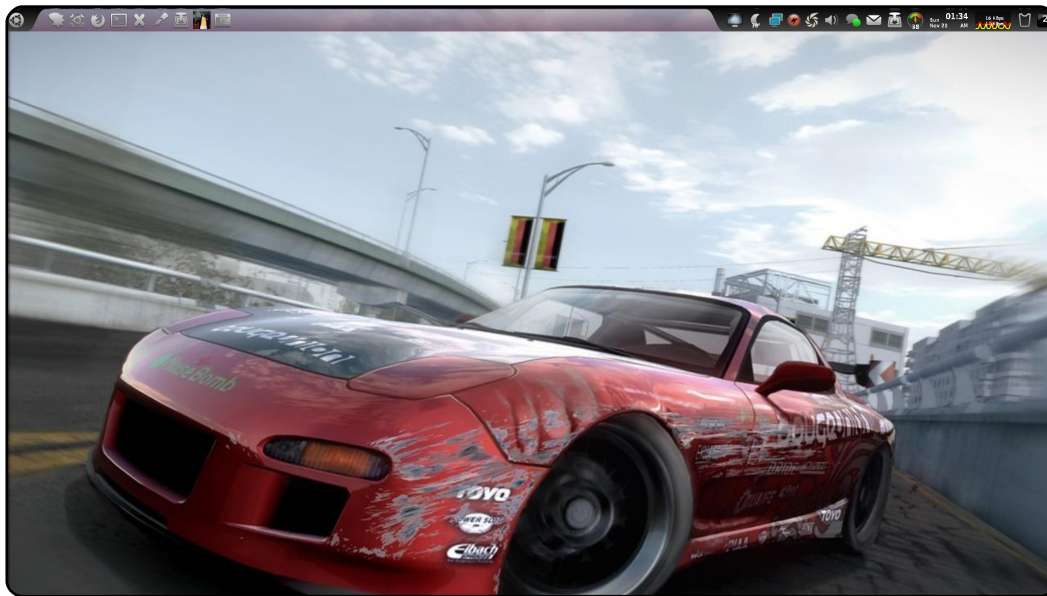
MY DESKTOP

I am running Ubuntu 10.10 Maverick Meerkat 64-bit. I don't like desktop icons, so I've disabled show desktop icons in gconf to give my desktop a clean look. Although you can't see it from the screenshot, also used is a wallpaper background slideshow created with "crebs" (create background slideshow). The Gnome Theme is a customized Elegant Gnome using the Aw0ken Icons. I replaced the default Ubuntu toolbars with the single AWN Dock (trunk version) using the Lucido style with the menu, launcher/task manger, show desktop, weather, notification area, indicator, hardware sensors, clock/calendar, bandwidth monitor, garbage, and desktop switcher applets - so it mimics the removed panels; also running is compiz because awn requires compositing to work. I chose compiz over Gnome's default compositing because Gnome's still has that video tearing bug, since the refresh rate still can't be adjusted to anything over 50mhz.

Corry Douglas

Here is a screen shot of my Ubuntu 10.10 desktop. The desktop was modified by installing 'cairo', several 'screenlets', and modifying a few compiz settings. I run the current nVidia video drivers at 1400x900 resolution on a Samsung 19" LCD flat panel monitor.

The wallpaper is from the Deviant Art web site (located in the "HD" section), and was #106.png at the time of download, and I think the 'watermark' picture is of Grace Park and from a site forgotten.

My basic system specs are: Gigabyte Ga-965P-ds3 motherboard, Intel Q-9400 processor, 4GB Gskill pc-6400 RAM, EVGA Geforce-9500-GT PCIE video card, and three SATA hard drives. The system is over-clocked from 2.6GHz to 3.2GHz, and the processor is cooled by an all copper Thermaltake Mini Typhoon heatsink/fan.

Larry Rudolph

gTune

Home: http://sourceforge.net/projects/gtune/

The first thing a guitarist needs to do, obviously, is tune his or her guitar. There are, of course, physical tuners that plug into your instrument and detect whether or not you are on pitch. However, if you don't feel like spending the money, there's a wonderful free alternative in the form of gTune. gTune is very simple: just download the binary, run it, and select the instrument and note. You can also set the length ("short", "long", or "everlasting") of the tone. It's minimalistic and it does its job well.

To install gTune, just download the appropriate file from the Sourceforge page. There's a pre-built .bin, if you don't want to deal with compiling.

eTktab

Home: http://etktab.sourceforge.net/
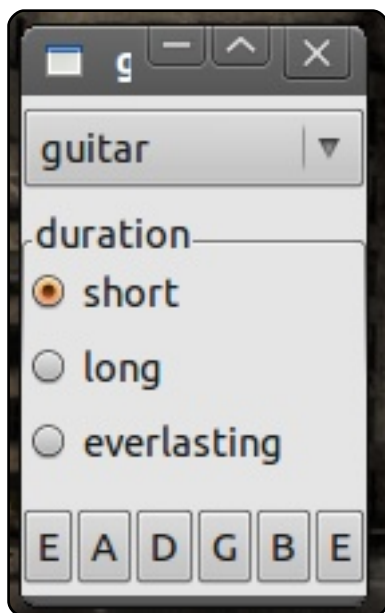
If you're a songwriter, you may appreciate eTktab. This open-source app, developed by Jason Sonnenschein, allows you to generate tab sheets using your keyboard. Essentially, you get thirty fret positions (five per string), each with a key assigned to it on your keyboard. You can also add additional strings (up to three more), tune the instrument, and even add lyrics. When you're done, you can export to an ASCII file, or simply copy and paste the tabs from the program to a textfile.

To install eTktab, use the Debian package on the official homepage.

contents ^

TuxGuitar

Home: http://tuxguitar.herac.com.ar/

If you want a more advanced (or more elegant) solution for music composition, try TuxGuitar. This Java-based program supports a huge variety of features, including different instruments, a built-in software keyboard and fretboard, various composition features (including time signature and tempo adjustment), and a sound previewer. Best of all, unlike most Java software, TuxGuitar sports a beautiful interface that fits into both KDE and Gnome.

To install TuxGuitar, use the **tuxguitar** package in the universe repository.

KGuitar

Home: http://kguitar.sourceforge.net/
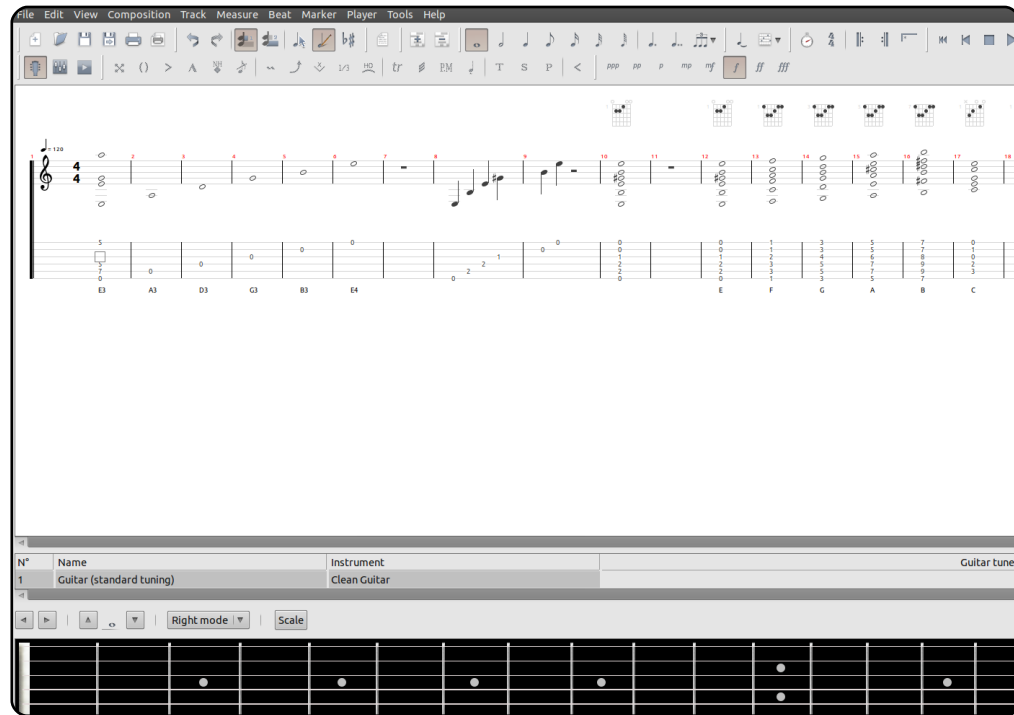
Another more advanced program, KGuitar was created out of Mikhail Yakshin's frustration at a lack of affordable yet effective solutions for guitarists using open-source operating systems. Several years later, Yakshin's project supports a wide range of features, including tab editing, a fretboard, lyrics, MIDI import/export, and support for GuitarPro, TablEdit, or ASCII formats. It also supports other instruments beyond guitars.

To install KGuitar, use the **kguitar** package in the universe repository.

Frets on Fire

Home: http://fretsonfire.sourceforge.net/
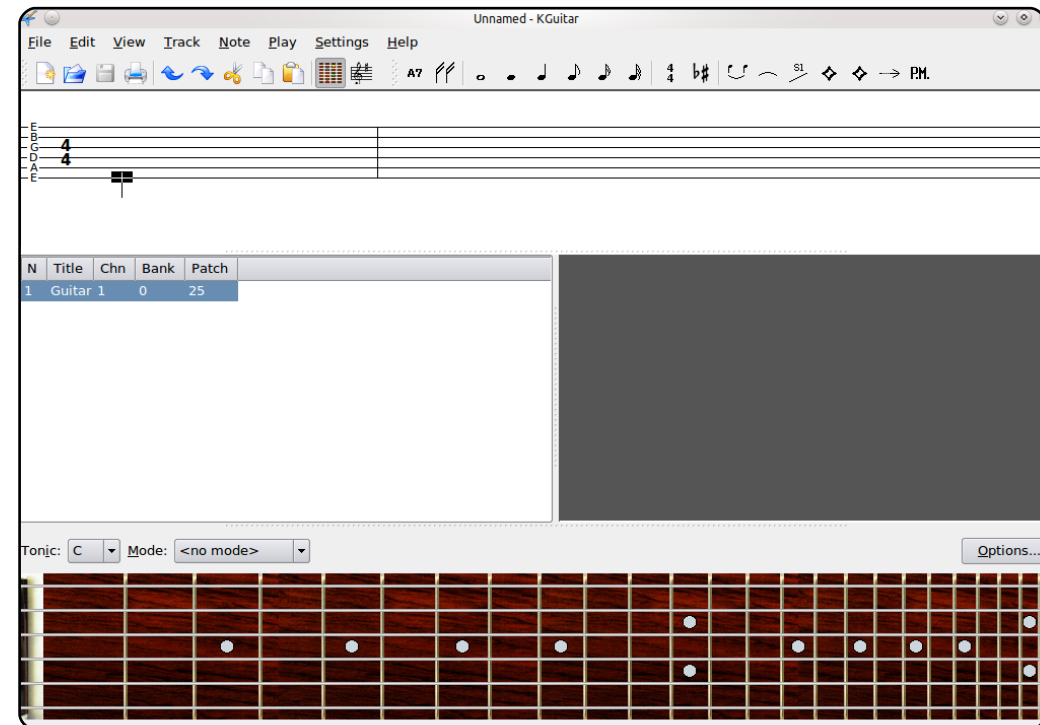
Whether you're a hardcore guitarist or a musical luddite, you'll most likely enjoy Frets on Fire. This open-source game shares many similarities with commercial hits Guitar Hero and Rock Band. Basically, you use your keyboard as guitar frets (F1-F5) and pick (Return) to try to hit the notes at exactly the right time. While FoF will probably never actually teach you anything musical, it will provide hours of entertainment for all ages and musical abilities.

To install Frets on Fire, use the **fretsonfire** package in the universe repository.
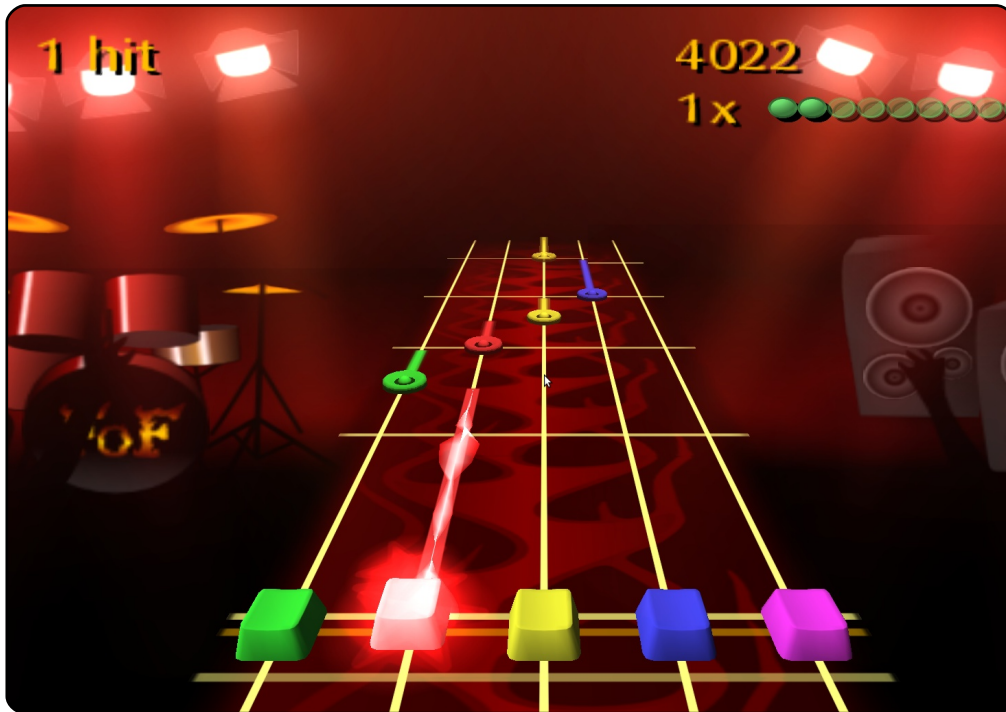
The Ubuntu UK podcast is presented by members of the United Kingdom's Ubuntu Linux community.

We aim is to provide current, topical information about, and for, Ubuntu Linux users the world over. We cover all aspects of Ubuntu Linux and Free Software, and appeal to everyone from the newest user to the oldest coder, from the command line to the latest GUI.

Because the show is produced by the Ubuntu UK community, the podcast is covered by the Ubuntu Code of Conduct and is therefore suitable for all ages.

http://podcast.ubuntu-uk.org/

Available in MP3/OGG format in Miro or iTunes, or listen to it directly on the site.

HOW TO CONTRIBUTE

We are always looking for new articles to include in Full Circle. For article guidelines, ideas, and for issue translation, please see our wiki:
http://wiki.ubuntu.com/UbuntuMagazine
Please email your articles to: articles@fullcirclemagazine.org

If you would like to submit **news**, email it to: news@fullcirclemagazine.org

Send your **comments** or Linux experiences to: letters@fullcirclemagazine.org

Hardware/software **reviews** should be sent to: reviews@fullcirclemagazine.org

Questions for Q&A should go to: questions@fullcirclemagazine.org

Desktop screens should be emailed to: misc@fullcirclemagazine.org

... or you can visit our **forum** via: www.fullcirclemagazine.org

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your Opinions, Desktops and Stories. We also need Reviews (games, apps & hardware), How-To articles (on any K/X/Ubuntu subject) and any questions, or suggestions, you may have.
Send them to: articles@fullcirclemagazine.org

Deadline for Issue #47:
Sunday 06th March 2011.

Release date for issue #47:
Friday 25th March 2011.