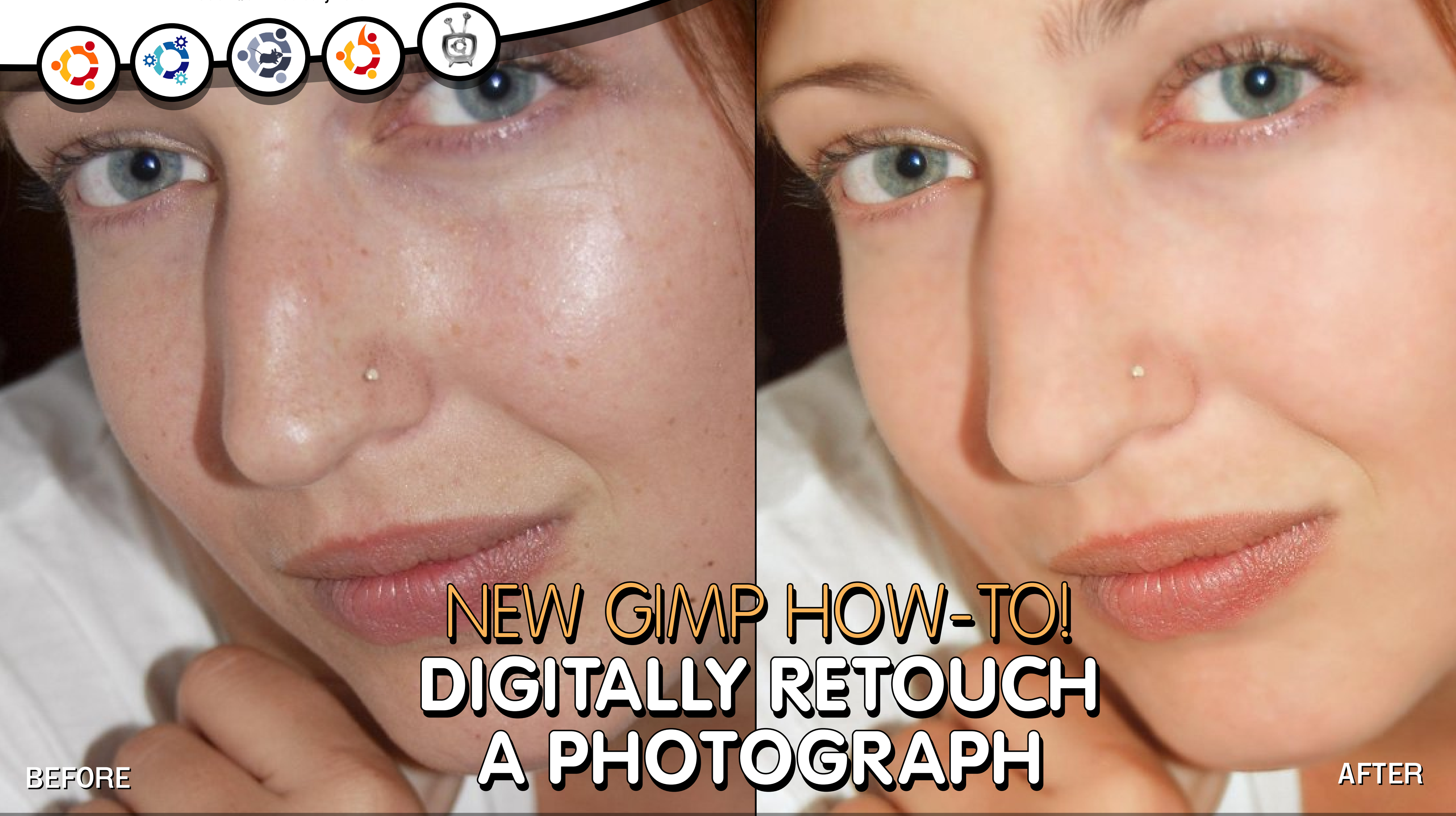
full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #34 - February 2010



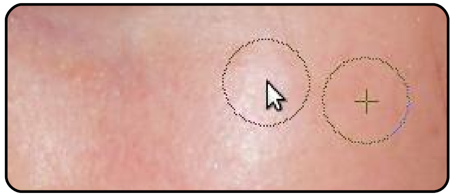
NEW GIMP HOW-TO!

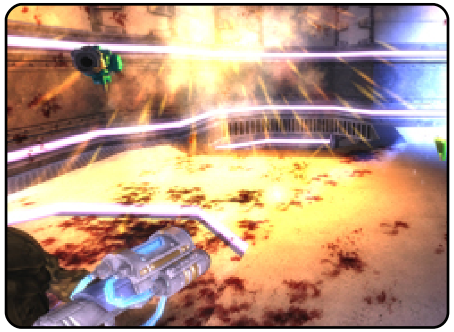
DIGITALLY RETOUCH A PHOTOGRAPH

full circle magazine is neither affiliated with, nor endorsed by, Canonical Ltd.

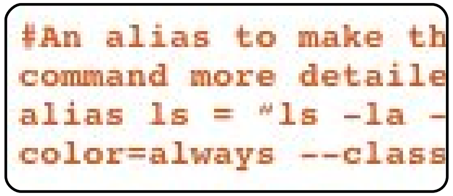
full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

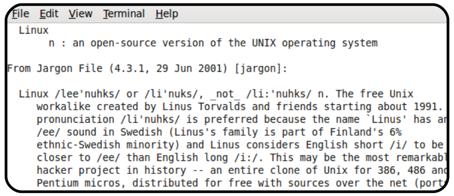




11:31.19, 13 April 2010
127.0.0.1
CPU usage
0...........50...........100
Intraware Inc International Bank
Uplink Credits Machine
Atlantic Plc Public Access Ser
Uplink International Bank
Uplink Test Machine
Walker Systems Access Terminal
Protovision
FreeService Plc Public Access Server
FourCom Technology Public Access Server
Atlantic Plc Internal Services Machine
Gateway
OCP remote monitoring system
Centura Computers Access Terminal
Datatek Networks Public Access Se
InterNIC
Unity Corp Public Access Server
Government Public A
HighEnd Industries International Bank



```
#An alias to make th
command more detaile
alias ls = "ls -la -
color=always --class
```



```
 Linux
      n : an open-source version of the UNIX operating system

From Jargon File (4.3.1, 29 Jun 2001) [jargon]:

 Linux /lee'nuhks/ or /li'nuks/, _not_ /li:'nuhks/ n. The free Unix
    workalike created by Linus Torvalds and friends starting about 1991.
    pronunciation /li'nuhks/ is preferred because the name `Linus' has ar
    /ee/ sound in Swedish (Linus's family is part of Finland's 6%
    ethnic-Swedish minority) and Linus considers English short /i/ to be
    closer to /ee/ than English long /i:/. This may be the most remarkabl
    hacker project in history -- an entire clone of Unix for 386, 486 and
    Pentium micros, distributed for free with sources over the net (port
```

Full Circle magazine is entirely independent of Canonical, the sponsor of the Ubuntu projects, and the views and opinions in the magazine should in no way be assumed to have Canonical endorsement.

EDITORIAL

Welcome to another issue of Full Circle magazine.

This month we have a new GIMP How-To on digitally retouching photographs. Many people think GIMP isn't quite up to the power of Photoshop, but this tutorial proves that GIMP can easily equal Photoshop. Hopefully, we'll have another GIMP How-To next issue. Speaking of GIMP, we're still getting letters from Ubuntu users who are wondering what, if anything, will replace GIMP in future editions of Ubuntu since surely it needs an image editor of some sort, and I have to agree. Since it can be a bit overwhelming for the beginner, maybe it's time to have a cut-down GIMP with just the basic features?

Also in this issue, Lucas (our *Command and Conquer* writer) has written a review of his new Asus UL30-A laptop, and in Ubuntu Games, Ed reviews *Uplink* from *Introversion Software*. Download the demo and give it a try, but do consider buying the full version and support those who are creating native Linux games, £20 ($33 / €26) is a bargain for a full game.

This month sees the return of the ***Full Circle Podcast***! Hosted by Robin Catling, Ed Hewitt and Dave Wilkins. Many thanks to them for taking up the challenge. Page four has the details and download links. You can contact them via: podcast@fullcirclemagazine.org

Enjoy the issue, and keep in touch!

All the best!
Ronnie
Editor, Full Circle magazine
ronnie@fullcirclemagazine.org

This magazine was created using :

What is Ubuntu?

Ubuntu is a complete operating system that is perfect for laptops, desktops and servers. Whether at home, school or work, Ubuntu contains all the applications you'll ever need including word processor, email application and web browser. Ubuntu is and always will be free of charge. You do not pay any licensing fees. You can download, use and share Ubuntu with your friends, family, school or business, for absolutely nothing.

Once installed, your system is ready to use with a full set of productivity, internet, drawing and graphics applications, and games.

TIP: use the new 'contents' link to jump to the contents page from any other page!

contents ^

NEWS

Laptop for computer novices

People confused and frustrated by computers can now turn to a laptop called Alex built just for them.

Based on Linux, the laptop comes with simplified e-mail, web browsing, image editing and office software.

Those who sign up for Alex pay £39.95 a month for telephone support, software updates and broadband access.

Its creators hope the laptop and its simple suite of software proves to be a popular alternative to the Windows and Mac operating systems.

Source: BBC News

Linux kernel 2.6.33 released

Linus Torvalds has released version 2.6.33 of the Linux kernel just two and a half months after the release of Linux 2.6.32.

Like it's predecessors, the new kernel in the Linux main development line has thousands of improvements, such as an open source driver for GeForce graphics hardware, support for the kernel-based mode setting (KMS), the DRBD (Distributed Replicated Block Device) replication solution and support for the Trim ATA command, as well as several new and updated drivers.

Source: h-online.com

Ubuntu 10.04 LTS Alpha 3 Released

The Ubuntu developers have announced the availability of the third alpha release of Ubuntu 10.04 LTS, code named "Lucid Lynx". The latest development milestone is the third of three planned alpha releases, which will be followed by two beta releases and a release candidate.

Ubuntu 10.04 LTS Alpha 3 is based on the 2.6.32 Linux kernel and features a number of changes over the previous 9.10 release, including the latest GNOME desktop environment. Additionally, the developers note that the open source Nouveau video driver is now used as the default for NVIDIA graphics hardware, improving resolution detection, and built-in social networking integration now includes support for Twitter, identi.ca and Facebook.

Source: h-online.com

Full Circle Podcast - Episode 01

That's right folks, the Full Circle Podcast is back and better than ever! The podcast is in both MP3 and OGG formats. Runtime is 37 minutes.

Topics include, the reshuffle at Canonical, the online Canonical music store, Google Buzz, Good Old Games and the Acer Aspire Revo.

Your Hosts:
- *Robin Catling*
- *Ed Hewitt*
- *Dave Wilkins*

The podcast and show notes are at: http://url.fullcirclemagazine.org/88a985

COMMAND & CONQUER

Written by Lucas Westermann

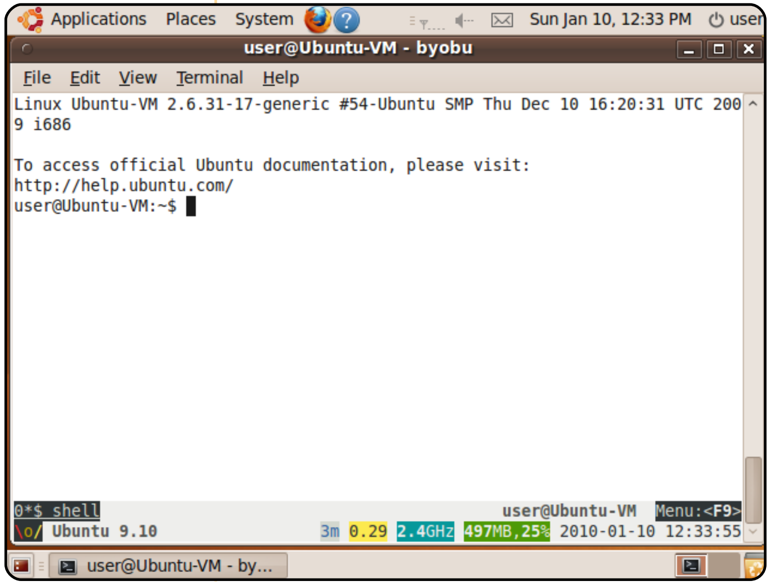
After Issue 33 was released last month, Ubuntu member Chris Johnston was kind enough to send me an email pointing me to Byobu (https://edge.launchpad.net/byobu). I admit I haven't had quite as much time to play around with it as I would like, but I have gotten a general idea of what it offers. Also, before I started this segment, I saw a lot of mentions of tmux on the Arch Linux Forums, so I'll be covering that here as an alternative. I won't be going into much detail on the features, as Byobu is similar to Screen's key bindings and settings, and tmux is extremely well documented, and my configuration file is commented and should be clear enough. Requests for a more in-depth article on either can be sent to me via email, since I'm not sure how much demand there will be for a thorough walk-through of their functions.

Byobu

“Byobu is a Japanese term for decorative, multi-panel screens that serve as folding room dividers. As an open-source project, Byobu is an elegant enhancement of the otherwise functional, plain, practical GNU Screen. Byobu includes an enhanced profile, and configuration utilities for the GNU screen window manager, such as toggle-able system status notifications.” (Taken from the Launchpad page.)

What this means is that Byobu isn't a re-write of Screen, but rather something that goes along with Screen, adding a few features. The main features I noticed that are different from Screen, is that by default, it comes with a status bar that offers more information on the system than Screen offers by default, as seen above right.

Also, Byobu offers an ncurses-based menu to create screen profiles. This can be accessed by hitting F9 (as seen on the task bar, “Menu: <F9>”). I won't add a screen-shot of the menu, since it's rather straightforward and self-explanatory.



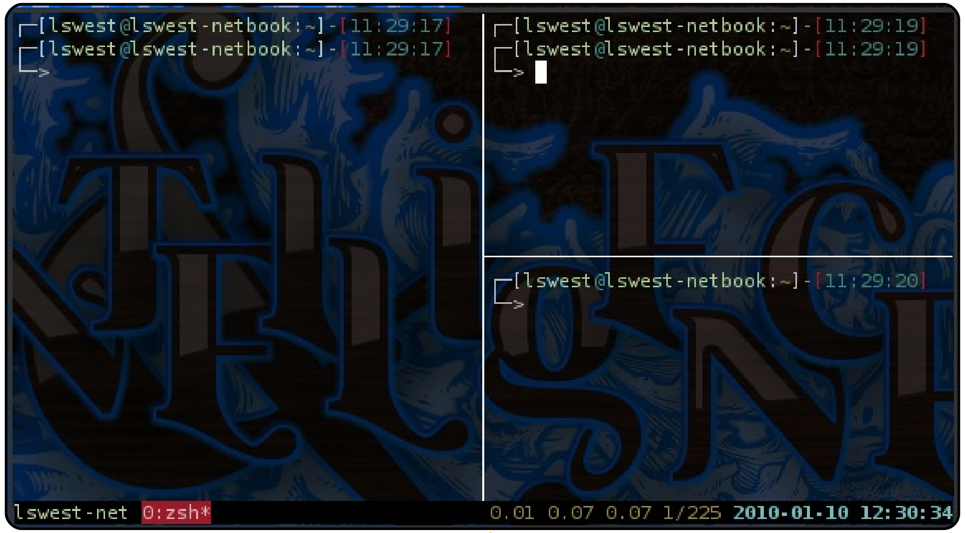
```
Linux Ubuntu-VM 2.6.31-17-generic #54-Ubuntu SMP Thu Dec 10 16:20:31 UTC 200
9 i686

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
user@Ubuntu-VM:~$ █









0*$ shell                                         user@Ubuntu-VM  Menu:<F9>
\o/ Ubuntu 9.10                 3m 0.29 2.4GHz 497MB,25% 2010-01-10 12:33:55
```

tmux

“tmux is a terminal multiplexer: it enables a number of terminals (or windows), each running a separate program, to be created, accessed, and controlled from a single screen. tmux may be detached from a screen and continue running in the background, then later reattached. tmux uses a client-server model. The server holds multiple sessions, and each window is an independent entity which may be freely linked to multiple sessions, moved between sessions, and otherwise manipulated. Each session may be attached to (display and accept keyboard input from) multiple clients. tmux is intended to be a modern, BSD-licensed alternative to programs such as GNU screen.” (From the homepage at http://tmux.sourceforge.net/)

What tmux offers—that is different from GNU Screen—are easier-to-understand commands, vertical and horizontal splitting, and sane defaults (status bar, etc.); moreover, it can be changed dynamically from the command-line, and it requires less memory than GNU Screen. It requires about 2.4 MB of memory for the first session, but each new window requires only 1 MB of memory. May not seem like much, but on an older box it can really make a difference. Also, by default, the key bindings all start with ctrl + b instead of ctrl + a —which can be changed, and is changed in my configuration file, which I'll post a link to at the end of the article. Due to the licensing, it is also included by default in BSD systems, for those who are interested in that kind of stuff. Also, a few key bindings are different, but the man page of tmux is extremely clear, and it offers a complete list of commands that can be accessed with the following command:

```
tmux list-commands
```



```
┌─[lswest@lswest-netbook:~]-[11:29:17]    ┌─[lswest@lswest-netbook:~]-[11:29:19]
┌─[lswest@lswest-netbook:~]-[11:29:17]    ┌─[lswest@lswest-netbook:~]-[11:29:19]
└─>                                        └─> █

                                           ┌─[lswest@lswest-netbook:~]-[11:29:20]
                                           └─>

lswest-net 0:zsh*                          0.01 0.07 0.07 1/225 2010-01-10 12:30:34
```

Above right is a screen-shot of tmux running (as you can probably tell, I actually use tmux on my netbook, as opposed to Byobu running in a Virtual Machine):

Configuration file (~/.tmux.conf): http://lswest.pastebin.com/fa64f955

List of commands for tmux: http://lswest.pastebin.com/f7d0cad21

I figured that I should include a few other choices in this series, since Linux is all about choice, and the freedom to use what you want.

Personally, I find tmux a bit easier to use and understand, but GNU Screen is the more widely-known program, which is why it was covered in-depth and tmux wasn't. However, the man page covers all the usual info, as well as keyboard shortcuts, configuration options, etc. I highly recommend looking at the man page before posing questions on how to do something, since most of what you'll need to know is in there (and pretty easy to find). Both of the programs are in the Universe repository in Ubuntu 9.10. As always, any article suggestions or questions can be sent to me at lswest34@gmail and I will do my best to answer the questions, and fulfil the requests.

Further Reading:

Byobu: https://edge.launchpad.net/byobu
and: http://blog.dustinkirkland.com/search/label/Byobu

tmux: http://www.openbsd.org/cgi-bin/man.cgi?query=tmux&sektion=1 (online man page)
and: http://tmux.sourceforge.net/ (homepage).

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.

Written by Greg Walters

Program In Python - Part 8

SEE ALSO:
FCM#27-33 - Python Parts 1 - 7

APPLICABLE TO:
ubuntu kubuntu xubuntu

CATEGORIES:
Dev Graphics Internet M/media System

DEVICES:
CD/DVD HDD USB Drive Laptop Wireless

We will continue programming our recipe database that we started in Part 7. This will be a long one, with a lot of code, so grab on with all your might and don't let go. But remember, keep your hands and feet inside the car at all times. We have already created our database. Now we want to display the contents, add to it and delete from it. So how do we do that? We will start with an application that runs in a terminal, so we need to create a menu. We will also create a class that will hold our database routines. Let's start with a stub of our program shown above right.

Now we will layout our menu. We do that so we can stub our class. Our menu will be a rather big loop that will display a list of options that the user can perform. We'll use a while loop. Change the menu routine to look like the code shown below right.

Next we stub the menu with an if|elif|else structure which is shown at the top of the next page.

Let's take a quick look at our menu routine. We start off by printing the prompts that the user can perform. We set a variable (loop) to True, and then use the while function to continue looping until loop = False. We use the raw_input() command to wait for the user to select an option, and then

```
#!/usr/bin/python
#------------------------------------------------------
# Cookbook.py
# Created for Beginning Programming Using Python #8
# and Full Circle Magazine
#------------------------------------------------------
import apsw
import string
import webbrowser

class Cookbook:

def Menu():
    cbk = Cookbook() # Initialize the class

Menu()
```

```
def Menu():
    cbk = Cookbook() # Initialize the class
    loop = True
    while loop == True:
        print
'=================================================='
        print '               RECIPE DATABASE'
        print
'=================================================='
        print ' 1 - Show All Recipes'
        print ' 2 - Search for a recipe'
        print ' 3 - Show a Recipe'
        print ' 4 - Delete a recipe'
        print ' 5 - Add a recipe'
        print ' 6 - Print a recipe'
        print ' 0 - Exit'
        print
'=================================================='
        response = raw_input('Enter a selection -> ')
```

contents ^

```
       if response == '1': # Show all recipes
      pass
    elif response == '2': # Search for a recipe
      pass
    elif response == '3': # Show a single recipe
      pass
    elif response == '4': # Delete Recipe
      pass
    elif response == '5': # Add a recipe
      pass
    elif response == '6': # Print a recipe
      pass
    elif response == '0': # Exit the program
      print 'Goodbye'
      loop = False
    else:
      print 'Unrecognized command.  Try again.'
```

```
/usr/bin/python -u
 "/home/greg/python_examples/APSW/cookbook/cookbook_stub.py"
===================================================
               RECIPE DATABASE
===================================================
1 - Show All Recipes
2 - Search for a recipe
3 - Show a Recipe
4 - Delete a recipe
5 - Add a recipe
6 - Print a recipe
0 - Exit
===================================================
Enter a selection ->
```

our if routine to handle whichever option the user selected. Before we can run this for a test, we need to create a stub inside our class for the __init__ routine:

```
def __init__(self):
    pass
```

Now, save your program where you saved the database you created from the last time, and run it. You should see something like that shown above right.

It should simply print the menu over and over, until you type “0”, and then print “Goodbye” and exit. At this point, we can now start stubs of our routines in the Cookbook class. We will need a routine that will display all the information out of the Recipes data table, one that will allow you to search for a recipe, one that will show the data for a single recipe from all three tables, one that will delete a recipe, one that will allow you to add a recipe, and one that will print the recipe to the default printer. The PrintAllRecipes routine doesn't need a parameter other than the (self) parameter, neither does the SearchforRecipe nor the EnterNew routines. The PrintSingleRecipe, DeleteRecipe and PrintOut routines all need to know what recipe to deal with, so they will need to have a parameter that we'll call “which”. Use the pass command to finish each stub. Under the Cookbook class, create the routine stubs:

```
def PrintAllRecipes(self):
  pass
def SearchForRecipe(self):
  pass
def PrintSingleRecipe(self,which):
  pass
def DeleteRecipe(self,which):
  pass
def EnterNew(self):
  pass
def PrintOut(self,which):
  pass
```

For a number of the menu items, we will want to print out all of the recipes from the Recipe table – so the user can pick from that list. These will be options 1, 3, 4 and 6. So, modify the menu routine for those options, replacing the pass command with cbk.PrintAllRecipes(). Our response check routine will now look like the code at the top of the next page.

One more thing to do is to set up the __init__ routine. Replace the stub with the following lines:

```
def __init__(self):
  global connection
  global cursor
  self.totalcount = 0
  connection=apsw.Connection("cookbook.db3")
  cursor=connection.cursor()
```

```
        if response == '1': # Show all recipes
            cbk.PrintAllRecipes()
        elif response == '2': # Search for a recipe
            pass
        elif response == '3': # Show a single recipe
            cbk.PrintAllRecipes()
        elif response == '4': # Delete Recipe
            cbk.PrintAllRecipes()
        elif response == '5': # Add a recipe
            pass
        elif response == '6': # Print a recipe
            cbk.PrintAllRecipes()
        elif response == '0': # Exit the program
            print 'Goodbye'
            loop = False
        else:
            print 'Unrecognized command.  Try again.'
```

First we create two global variables for our connection and cursor. We can access them from anywhere within the cookbook class. Next, we create a variable self.totalcount which we use to count the number of recipes. We'll be using this variable later on. Finally we create the connection and the cursor.

The next step will be to flesh out the PrintAllRecipes() routine in the Cookbook class. Since we have the global variables for connection and cursor, we don't need to re-create them in each routine.

Next, we will want to do a “pretty print” to the screen for headers for our recipe list. We'll use the “%s” formatting command, and the left justify command, to space out our screen output. We want it to look like this:

```
Item Name    Serves    Source

----------------------------
```

Finally, we need to create our SQL statement, query the database, and display the results. Most of this was covered in the article last time.

```
    sql = 'SELECT * FROM
Recipes'
    cntr = 0
    for x in
cursor.execute(sql):
        cntr += 1
        print '%s %s %s %s'
%(str(x[0]).rjust(5),x[1].lju
st(30),x[2].ljust(20),x[3].lj
ust(30))
    print '-------------'
    self.totalcount = cntr
```

The cntr variable will count the number of recipes we display to the user. Now our routine is done. Shown below is the full code for the routine, just in case you missed something.

Notice that we are using the tuple that is returned from the cursor.execute routine from ASPW. We are printing the pkID as the item for each recipe. This will allow us to select the correct recipe later on. When you run your program, you should see the menu, and when you select option 1, you'll get what's shown at the top of the next page.

That's what we wanted, except if you are running the app in Dr.Python or the like, the program doesn't pause. Let's add a pause until the user presses a key so they can look at the output for a second or two. While we are at it, let's print out the total number of recipes from the variable we set up a moment ago. Add to the bottom of option 1 of the menu:

```
def PrintAllRecipes(self):
        print '%s %s %s %s'
%('Item'.ljust(5),'Name'.ljust(30),'Serves'.ljust(20),
'Source'.ljust(30))
        print '--------------------------------------'
        sql = 'SELECT * FROM Recipes'
        cntr = 0
        for x in cursor.execute(sql):
            cntr += 1
            print '%s %s %s %s'
%(str(x[0]).rjust(5),x[1].ljust(30),x[2].ljust(20),x[3
].ljust(30))
        print '--------------------------------------'
        self.totalcount = cntr
```

```
Enter a selection -> 1
Item  Name                           Serves               Source
---------------------------------------------------------------------------------------
    1 Spanish Rice                   4                    Greg
    2 Pickled Pepper-Onion Relish    9 half pints         Complete Guide to Home Canning
---------------------------------------------------------------------------------------
===================================================
               RECIPE DATABASE
===================================================
1 - Show All Recipes
2 - Search for a recipe
3 - Show a Recipe
4 - Delete a recipe
5 - Add a recipe
6 - Print a recipe
0 - Exit
===================================================
Enter a selection ->
```

```
print 'Total Recipes - %s'
%cbk.totalcount

print '----------------------
-----------------------------
-'

res = raw_input('Press A Key
-> ')
```

We'll skip option #2 (Search for a recipe) for a moment, and deal with #3 (Show a single recipe). Let's deal with the menu portion first. We'll show the list of recipes, as for option 1, and then ask the user to select one. To make sure we don't get errors due to a bad user input, we'll use the Try|Except structure. We will print the prompt to the user (Select a recipe →), then, if they enter a correct response, we'll call the PrintSingleRecipe() routine in our Cookbook class with the pkID from our Recipe table. If the entry is not a number, it will raise a ValueError exception, which we handle with the except ValueError: catch shown right.

Next, we'll work on our PrintSingleRecipe routine in the Cookbook class. We start with the connection and cursor again, then create our SQL statement. In this case, we use 'SELECT * FROM Recipes WHERE pkID = %s” % str(which)' where which is the value we want to find. Then we “pretty print” the output, again from the tuple returned by ASPW. In this case, we use x as the gross variable, and then each one with bracketed index into the tuple. Since the table layout is pkID/name/servings/source, we can use x[0],x[1],x[2] and x[3] as the detail. Then, we want to select everything from the ingredients table where the recipeID (our key into the recipes data table) is equal to the pkID we just used. We loop through the tuple returned, printing each ingredient, and then finally we get the instructions from the instructions table – just like we did for the ingredients table. Finally, we wait for the user to press a key so they can see the recipe on the screen. The code is shown on the next page.

Now, we have two routines

```
try:
    res = int(raw_input('Select a Recipe -> '))
    if res <= cbk.totalcount:
        cbk.PrintSingleRecipe(res)
    elif res == cbk.totalcount + 1:
        print 'Back To Menu...'
    else:
        print 'Unrecognized command.  Returning to menu.'
except ValueError:
    print 'Not a number...back to menu.'
```

out of the six finished. So, let's deal with the search routine, again starting with the menu. Luckily this time, we just call the search routine in the class, so replace the pass command with:

```
cbk.SearchForRecipe()
```

Now to flesh out our search code. In the Cookbook class, replace our stub for the SearchForRecipe with the code shown on the next page.

There's a lot going on there. After we create our connection and cursor, we display our search menu. We are going to give the user three ways to search, and a way to exit the routine. We can let the user search by a word in the recipe name, a word in the recipe source, or a word in the ingredient list. Because of this, we can't just use the display routine we just created, and will need to create custom printout routines. The first two options use simple SELECT statements with an added twist. We are using the "like" qualifier. If we were using a query browser like SQLite Database Browser, our like statement uses a wildcard character of "%". So, to look for a recipe containing "rice" in the recipe name, our query would be:

```
SELECT * FROM Recipes WHERE
name like '%rice%'
```

However, since the "%" character is also a substitution character in our strings, we have to use %% in our text. To make it worse, we are using the substitution character to insert the word the user is searching for. Therefore, we must make it '%%%s%%'. Sorry if this is as clear as mud. The third query is called a Join statement. Let's look at it a bit closer:

```
sql = "SELECT
r.pkid,r.name,r.servings,r.so
urce,i.ingredients FROM
Recipes r Left Join
ingredients i on (r.pkid =
i.recipeid) WHERE
i.ingredients like '%%%s%%'
GROUP BY r.pkid" %response
```

We are selecting everything from the recipe table, and the ingredients from the ingredients table, joining or relating the ingredient table ON the recipeID being equal to the pkID in the recipe table, then searching for our ingredient using the like statement, and, finally, grouping the result by the pkID in the recipe table to keep duplicates from being shown. If you remember, we have peppers twice in the second recipe (Onion and pepper relish), one green and one red.

```
  def PrintSingleRecipe(self,which):
          sql = 'SELECT * FROM Recipes WHERE pkID = %s' %
str(which)
          print
'~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
          for x in cursor.execute(sql):
              recipeid =x[0]
              print "Title: " + x[1]
              print "Serves: " + x[2]
              print "Source: " + x[3]
          print
'~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
          sql = 'SELECT * FROM Ingredients WHERE RecipeID
= %s' % recipeid
          print 'Ingredient List:'
          for x in cursor.execute(sql):
              print x[1]
          print ''
          print 'Instructions:'
          sql = 'SELECT * FROM Instructions WHERE RecipeID
= %s' % recipeid
          for x in cursor.execute(sql):
              print x[1]
          print
'~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~'
          resp = raw_input('Press A Key -> ')
```

That could create confusion in our user's mind. Our menu uses

```
  searchin =
raw_input('Enter Search Type
-> ')

if searchin != '4':
```

which says: if searchin (the value the user entered) is NOT equal to 4 then do the options, if it is 4, then don't do

```
    def SearchForRecipe(self):
        # print the search menu
        print '-------------------------------'
        print ' Search in'
        print '-------------------------------'
        print ' 1 - Recipe Name'
        print ' 2 - Recipe Source'
        print ' 3 - Ingredients'
        print ' 4 - Exit'
        searchin = raw_input('Enter Search Type -> ')
        if searchin != '4':
            if searchin == '1':
                search = 'Recipe Name'
            elif searchin == '2':
                search = 'Recipe Source'
            elif searchin == '3':
                search = 'Ingredients'
            parm = searchin
            response = raw_input('Search for what in %s (blank to exit) -> ' % search)
            if parm == '1': # Recipe Name
                sql = "SELECT pkid,name,source,servings FROM Recipes WHERE name like '%%%s%%'" %response
            elif parm == '2': # Recipe Source
                sql = "SELECT pkid,name,source,servings FROM Recipes WHERE source like '%%%s%%'" %response
            elif parm == '3': # Ingredients
                sql = "SELECT r.pkid,r.name,r.servings,r.source,i.ingredients FROM Recipes r Left Join ingredients i
on (r.pkid = i.recipeid) WHERE i.ingredients like '%%%s%%' GROUP BY r.pkid" %response
            try:
                if parm == '3':
                    print '%s %s %s %s %s'
%('Item'.ljust(5),'Name'.ljust(30),'Serves'.ljust(20),'Source'.ljust(30),'Ingredient'.ljust(30))
                    print '--------------------------------------------------------------------------------------'
                else:
                    print '%s %s %s %s' %('Item'.ljust(5),'Name'.ljust(30),'Serves'.ljust(20),'Source'.ljust(30))
                    print '--------------------------------------------------------------------------------------'
                for x in cursor.execute(sql):
                    if parm == '3':
                        print '%s %s %s %s %s'
%(str(x[0]).rjust(5),x[1].ljust(30),x[2].ljust(20),x[3].ljust(30),x[4].ljust(30))
                    else:
                        print '%s %s %s %s' %(str(x[0]).rjust(5),x[1].ljust(30),x[3].ljust(20),x[2].ljust(30))
            except:
                print 'An Error Occured'
            print '--------------------------------------------------------------------------------------'
            inkey = raw_input('Press a key')
```

anything, just fall through. Notice that I used “!=” as Not Equal To instead of “<>”. Either will work under Python 2.x. However, in Python 3.x, it will give a syntax error. We'll cover more Python 3.x changes in a future article. For now, start using “!=” to make your life easier to move to Python 3.x in the future. Finally, we “pretty print” again our output. Let's look at what the user will see, shown right.

You can see how nicely the program prints the output. Now, the user can go back to the menu and use option #3 to print whichever recipe they want to see. Next we will add recipes to our database. Again, we just have to add one line to our menu routine, the call to the EnterNew routine:

```
cbk.EnterNew()
```

The code that needs to replace the stub in the Cookbook class for EnterNew() is at: http://pastebin.com/f1d868e63.

We start by defining a list named “ings” – which stands

```
Enter a selection -> 2
-------------------------------
Search in
-------------------------------
1 - Recipe Name
2 - Recipe Source
3 - Ingredients
4 - Exit
Enter Search Type -> 1
Search for what in Recipe Name (blank to exit) -> rice
Item  Name                           Serves               Source
--------------------------------------------------------------------------------------
    1 Spanish Rice                   4                    Greg
--------------------------------------------------------------------------------------
Press a key
```

Easy enough. Now for the ingredient search...

```
Enter a selection -> 2
-------------------------------
Search in
-------------------------------
1 - Recipe Name
2 - Recipe Source
3 - Ingredients
4 - Exit
Enter Search Type -> 3
Search for what in Ingredients (blank to exit) -> onion
Item  Name                  Serves            Source                         Ingredient

--------------------------------------------------------------------------------------
1 Spanish Rice                 4              Greg                           1 small
Onion chopped
2 Pickled Pepper-Onion Relish  9 half pints   Complete Guide to Home Canning 6 cups
finely chopped Onions
--------------------------------------------------------------------------------------
Press a key
```

for ingredients. We then ask the user to enter the title, source, and servings. We then enter a loop, asking for each ingredient, appending to the ing list. If the user enters 0, we exit the loop and continue on asking for the instructions. We then show the recipe contents and ask the user to verify before saving the data. We use INSERT INTO statements, like we did last time, and return to the menu. One thing we have to be careful of is the single quote in our entries. USUALLY, this won't be a problem in the ingredient list or the instructions, but in our title or source fields, it could come up. We need to add an escape character to any single quotes. We do this with the string.replace routine, which is why we imported the string library. In the menu routine, put the code shown above right under option #4.

Then, in the Cookbook class, use the code shown below right for the DeleteRecipe() routine.

Quickly, we'll go through the delete routine. We first ask the user which recipe to delete (back in the menu), and pass that pkID number into our delete routine. Next, we ask the user 'are they SURE' they want to delete the recipe. If the response is “Y” (string.upper(resp) == 'Y'), then we create the sql delete statements. Notice that this time we have to delete records from all three tables. We certainly could just delete the record from the recipes table, but then we'd have orphan records in the other two, and that wouldn't be good. When we delete the record from the recipe table, we use the pkID field. In the other two tables, we use the recipeID field.

Finally, we will deal with the routine to print the recipes. We'll be creating a VERY simple HTML file, opening the default browser and allowing them to print from there. This is why we are importing the webbrowser library. In the menu routine for option #6, insert the code shown at the top of the next page.

Again, we display a list of all the recipes, and allow them to select the one that they wish to print. We call the PrintOut routine in the Cookbook class. That code is shown at the top right of the next page.

```
cbk.PrintAllRecipes()
     print '0 - Return To Menu'
     try:
         res = int(raw_input('Select a Recipe to DELETE
or 0 to exit -> '))
         if res != 0:
             cbk.DeleteRecipe(res)
         elif res == '0':
             print 'Back To Menu...'
         else:
             print 'Unrecognized command.  Returning to
menu.'
     except ValueError:
          print 'Not a number...back to menu.'
```

```
     def DeleteRecipe(self,which):
        resp = raw_input('Are You SURE you want to
Delete this record? (Y/n) -> ')
        if string.upper(resp) == 'Y':
            sql = "DELETE FROM Recipes WHERE pkID = %s"
% str(which)
            cursor.execute(sql)
            sql = "DELETE FROM Instructions WHERE
recipeID = %s" % str(which)
            cursor.execute(sql)
            sql = "DELETE FROM Ingredients WHERE
recipeID = %s" % str(which)
            cursor.execute(sql)
            print "Recipe information DELETED"
            resp = raw_input('Press A Key -> ')
        else:
            print "Delete Aborted - Returning to menu"
```

We start with the fi = open([filename],'w') command which creates the file. We then pull the information from the recipe table, and write it to the file with the fi.write command. We use the <H1></H1> header 1 tag for the title, the

<H2> tag for servings and source. We then use the <li></li> list tags for our ingredient list, and then write the instructions. Other than that it's simple queries we've already learned. Finally, we close the file with the fi.close() command, and use webbrowser.open([filename]) with the file we just created. The user can then print from their web browser – if required.

```
cbk.PrintAllRecipes()
       print '0 - Return To Menu'
       try:
           res = int(raw_input('Select a Recipe to DELETE or 0 to exit -> '))
           if res != 0:
               cbk.PrintOut(res)
           elif res == '0':
               print 'Back To Menu...'
           else:
               print 'Unrecognized command.  Returning to menu.'
       except ValueError:
           print 'Not a number...back to menu.'
```

```
def PrintOut(self,which):
    fi = open('recipeprint.html','w')
    sql = "SELECT * FROM Recipes WHERE pkID = %s" % which
    for x in cursor.execute(sql):
        RecipeName = x[1]
        RecipeSource = x[3]
        RecipeServings = x[2]
    fi.write("<H1>%s</H1>" % RecipeName)
    fi.write("<H2>Source: %s</H2>" % RecipeSource)
    fi.write("<H2>Servings: %s</H2>" % RecipeServings)
    fi.write("<H3> Ingredient List: </H3>")
    sql = 'SELECT * FROM Ingredients WHERE RecipeID = %s' % which
    for x in cursor.execute(sql):
        fi.write("<li>%s</li>" % x[1])
    fi.write("<H3>Instructions:</H3>")
    sql = 'SELECT * FROM Instructions WHERE RecipeID = %s' % which
    for x in cursor.execute(sql):
        fi.write(x[1])
    fi.close()
    webbrowser.open('recipeprint.html')
    print "Done"
```

WHEW! This was our biggest application to date. I've posted the full source code (and the sample database if you missed last month) on my website. If you don't want to type it all in or have any problems, then hop over to my web site, www.thedesignatedgeek.com to get the code.

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.

HOW-TO

Digitally Retouch Photos in GIMP

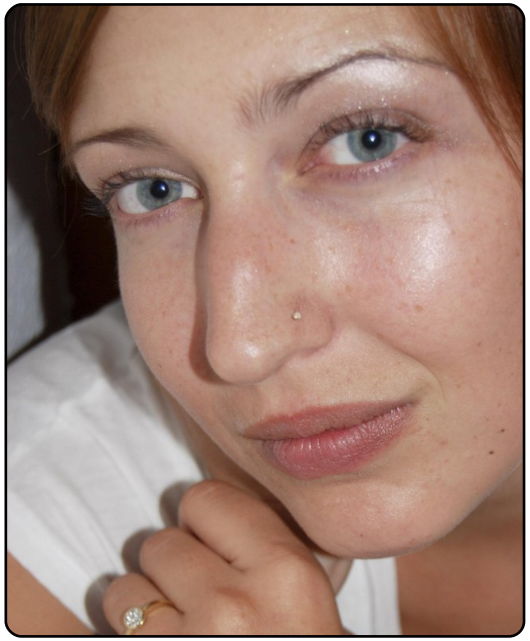
Written by Hüseyin SARIGÜL

SEE ALSO:
N/A

APPLICABLE TO:
ubuntu kubuntu xubuntu

CATEGORIES:
Dev Graphics Internet M/media System

DEVICES:
CD/DVD HDD USB Drive Laptop Wireless



This month, our subject is digital retouching of photographs. Nowadays, unblemished or even perfect faces are the rule on magazine covers and billboards. In this article I'll show you some general cosmetic retouching techniques.

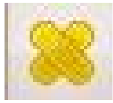
Professional photographers and artworkers digitally retouch in a few different ways. Some use Gimp or similar tools, Others prefer filters. Here, we will use both together. We'll retouch in a practical way. First, we'll see how to remove acne from photos.

Open the photo (my source image is shown left) in Gimp and check it for defects.

I've detected the following defects:

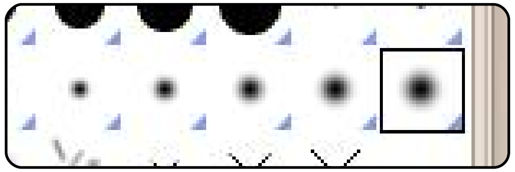
1. Flash reflections on the face.
2. Acne on the face.
3. Lines and wrinkles
4. Suboptimal colors.

Now, one by one, we will apply our solutions. We'll begin by removing the acne and spots. The healing tool will be used for this. Choose the healing tool (right) from the toolbox. Its shortcut key is H.

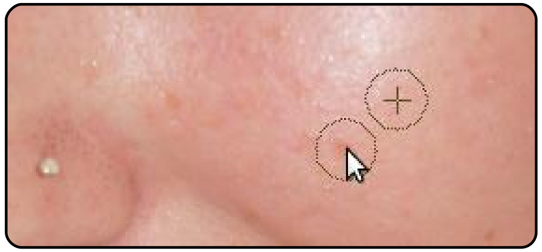


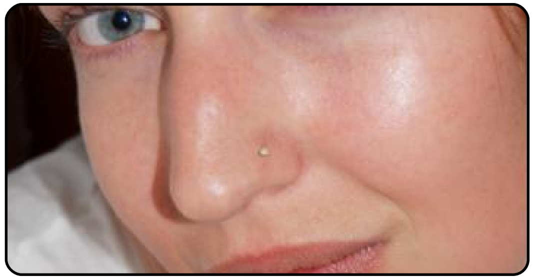
Healing Tool

Now we need to choose a brush. Circle Fuzzy (shown below, selected) is a suitable option. You can change the size of the brush by increasing / decreasing the scale value.



The healing tool is for removing acne and spots. First, with + and - zoom into the photo. Then mark the best color option - while pressing CTRL, then release the CTRL and click on the acne/spot.



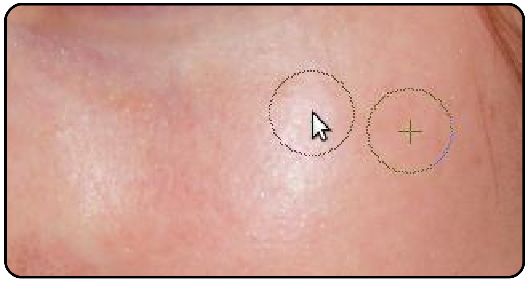


You should see that the acne has disappeared. You should perform the same process for all acne/spots.

All acne has been removed. Now it's time to remove the light from the flash - for this process we can use the clone tool or healing tool.

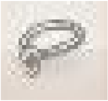
Clone Tool: Selectively copy from an image or pattern, using a brush.

In both tools, we maintain a low opacity value and we must do this by using different regions and for several attempts - not all at once.



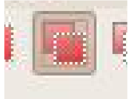
While the Ctrl key is pressed, click on the location which has the ground color close to a shiny place. Click a different place again and do the same process. Let this process continue until the light from the flash is gone.

Now we'll select some parts of the face. Choose the Free Select tool from the toolbox.

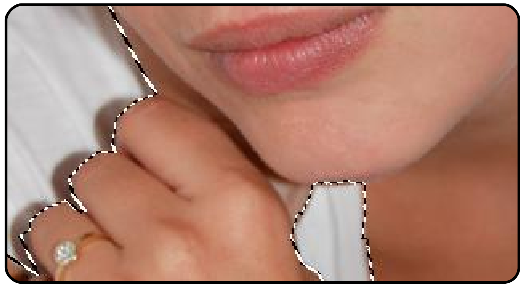


Free Select Tool: Select a hand-drawn region with free and polygonal segments

We will need to do more than one selection, so we'll choose the add to selection icon shown right.



We choose the face as our first selection, then, after clicking the add to selection icon we hold down SHIFT and draw a second selection around the hand.

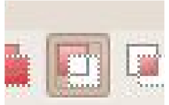


We choose the borders of the area that we would like to work on. If you make any mistake, you can use backspace to undo the previous action.

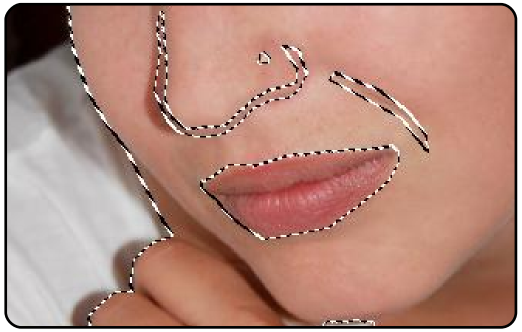
Now we should deselect some parts from our selection.

- Eyes
- Eyebrows
- Nostril
- Accessories

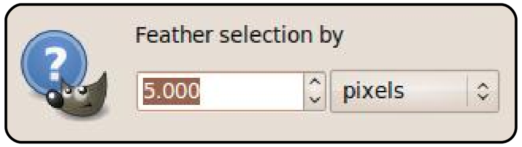
Choose the third option from the toolbox:



Deselect the parts that we have mentioned.



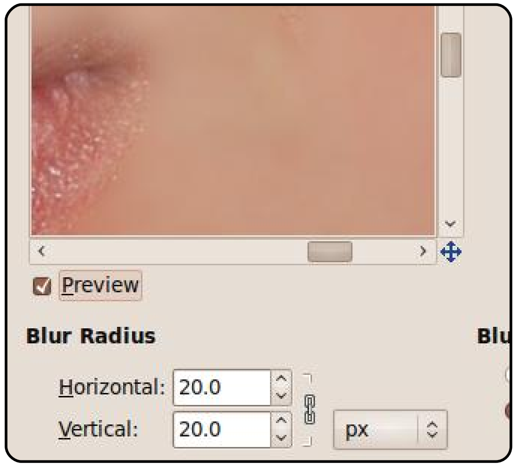
Don't forget to apply feather after each selection. Select > Feather and a value of 5 pixels.



Feather selection by

5.000 pixels

Then apply a filter. Click, Filters > Blur > Gaussian Blur.



☑ Preview

Blur Radius

Horizontal: 20.0

Vertical: 20.0 px

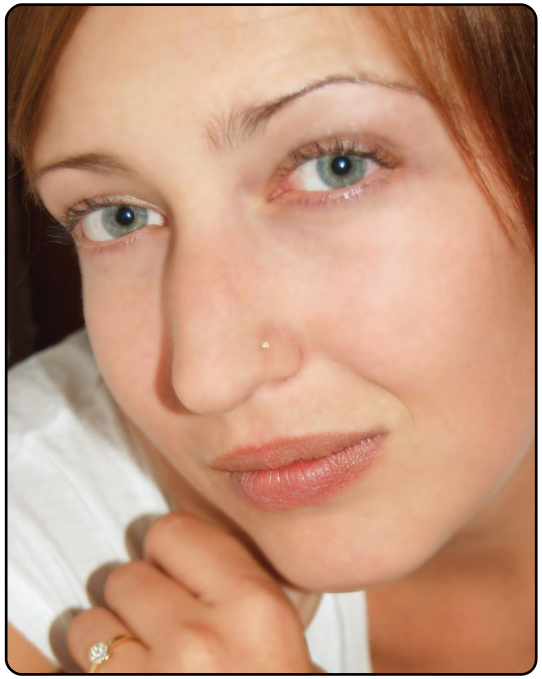
Choose suitable values, and click OK.

Press Shift + Ctrl + A or choose *None* from the Select menu.

Success! The last thing is correcting the colours. Click Colours > Colour Balance. The How-To for more information on this will be our next tutorial.

Translated from Turkish into English by Kaan Bahadır TERMELİ

This picture has a CC licence from Brenda Annerl http://www.flickr.com/photos/brendaannerl/2729960358/sizes/m/



contents ^

HOW-TO

Written by Falko Timme on HowtoForge.com

The Perfect Server - Part 4

SEE ALSO:
FCM09 - 16 : Server Series 1 - 8
FCM28 - 29 : LAMP Server 1 - 2
FCM31 - 33 : The Perfect Server 1 - 3

APPLICABLE TO:
ubuntu kubuntu xubuntu

CATEGORIES:
Dev Graphics Internet M/media System

DEVICES:
CD/DVD HDD USB Drive Laptop Wireless

To install the SquirrelMail webmail client, run:

```
aptitude install squirrelmail
```

Then, create the following symlink...

```
ln -s
/usr/share/squirrelmail/
/var/www/webmail
```

... and configure SquirrelMail:

```
squirrelmail-configure
```

We must tell SquirrelMail that we are using Courier-IMAP/-POP3:

```
SquirrelMail Configuration :
Read: config.php (1.4.0)
Main Menu
1.  Organization Preferences
2.  Server Settings
3.  Folder Defaults
4.  General Options
5.  Themes
6.  Address Books
7.  Message of the Day (MOTD)
8.  Plugins
9.  Database
10. Languages

D.  Set pre-defined settings
for specific IMAP servers
C   Turn color on
S   Save data
Q   Quit

Command >>
```

Enter: **D**

Now, you will see a list of IMAP server options entitled:

```
Please select your IMAP
server:
```

Enter the word: **courier**

```
imap_server_type = courier
default_folder_prefix =
INBOX.
trash_folder = Trash
sent_folder = Sent
draft_folder = Drafts
show_prefix_option = false
default_sub_of_inbox = false
show_contain_subfolders_optio
n = false
optional_delimiter = .
delete_folder = true

Press any key to continue...
```

Next, you will see a list of options and their settings; press the **Enter** key to continue.

Back at the Main Menu, enter: **S** to save data, and you will see:

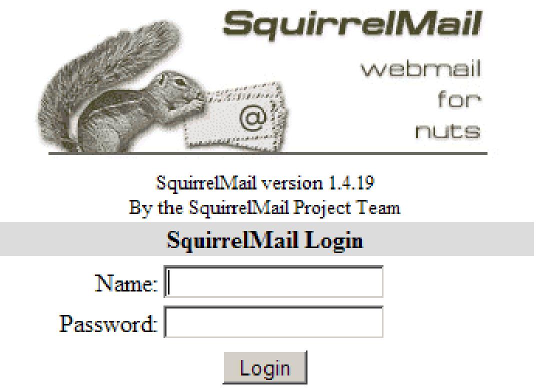
```
Data saved in config.php
Press enter to continue
```

Back at the Main Menu, enter **Q** to quit.

Afterwards you can access SquirrelMail under: http://server1.example.com/webmail

or:

http://192.168.0.100/webmail



SquirrelMail version 1.4.19
By the SquirrelMail Project Team

Install ISPConfig 3

To install ISPConfig 3 from the latest released version, do this (replacing ISPConfig-3.0.1.6.tar.gz with the latest version) :

```
cd /tmp

wget
http://downloads.sourceforge.
net/ispconfig/ISPConfig-
3.0.1.6.tar.gz?use_mirror=

tar xvfz ISPConfig-
3.0.1.6.tar.gz
```

```
cd
ispconfig3_install/install/
```

The next step is to run:

```
php -q install.php
```

This will start the ISPConfig 3 installer. Press **Enter** for each option - except when asked for your MySQL root password.

The installer automatically configures all underlying services, so no manual configuration is needed.

Afterwards you can access ISPConfig 3 under:

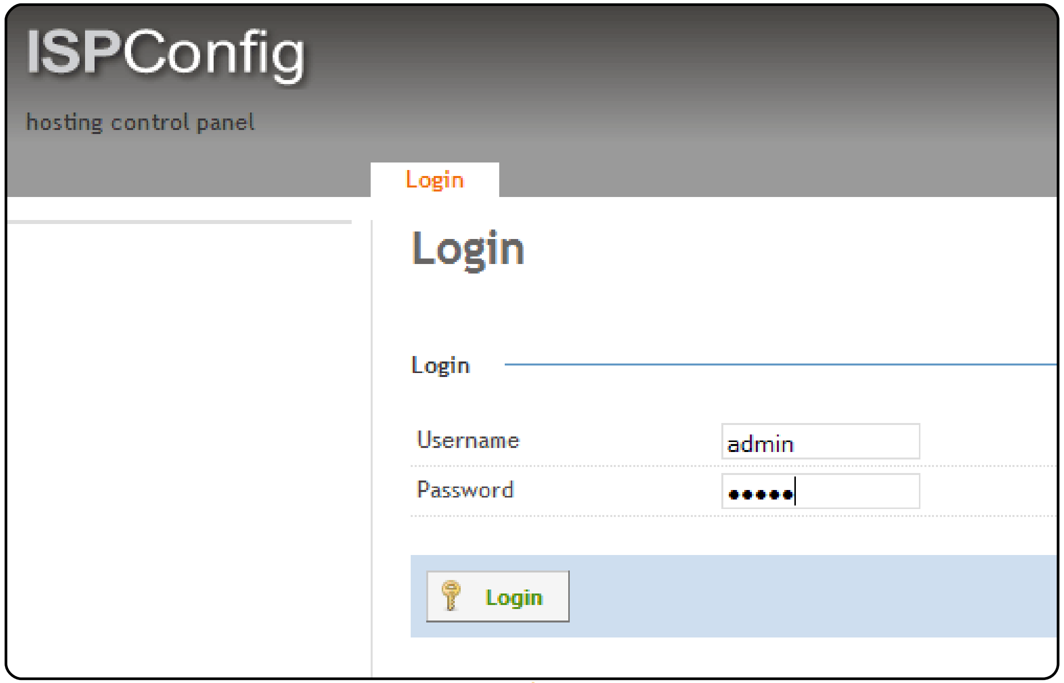
http://server1.example.com:8080/

or:

http://192.168.0.100:8080/

Log in with the username ***admin*** and the password ***admin*** (you should change the default password after your first login):

The system is now ready to be used.

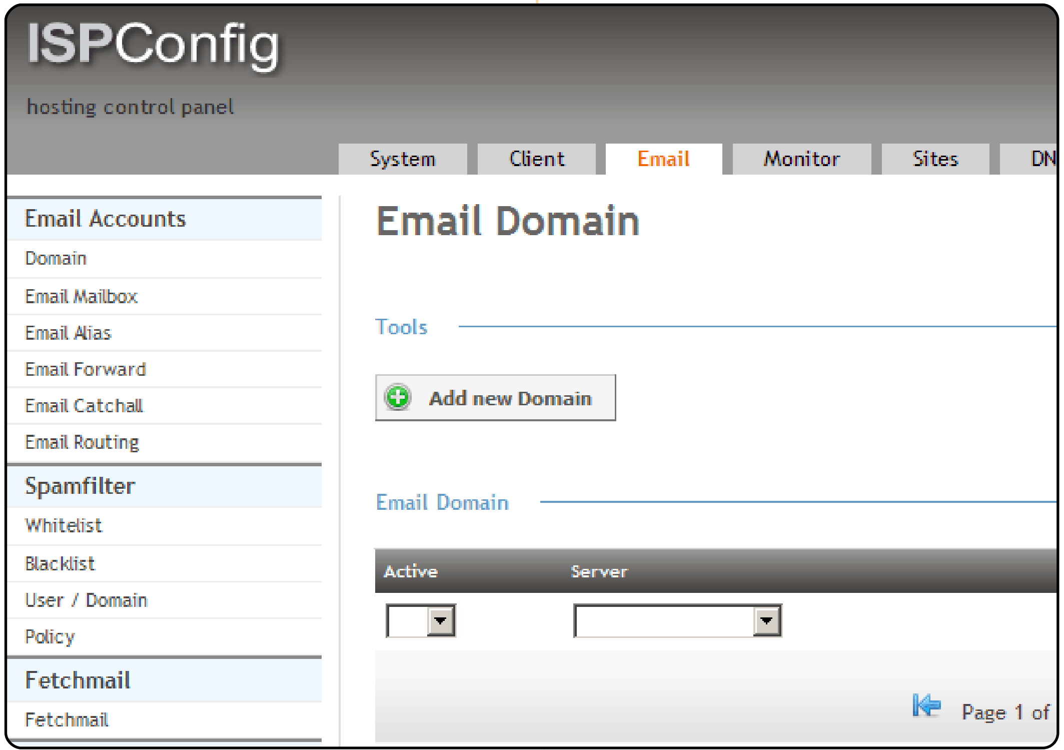


Login

Login

Username: admin

Password: •••••



Linux User

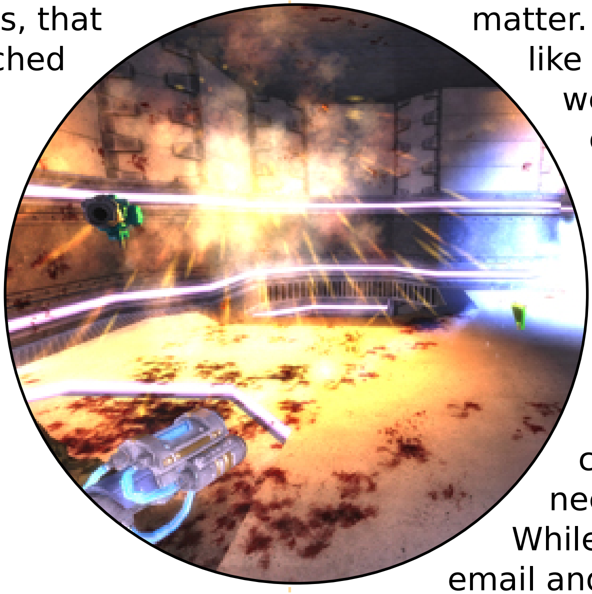
Written by Mike Creamer

My Linux story began with frustration. I am a high-school teacher who relies on his computer for everything. From typing tests to preparing lessons to keeping track of grades, my home computer kept my classroom cooking. But two years ago my desktop computer picked up a Trojan that left the system inoperable. Despite frequently updating antivirus- and spyware-protection software, my computer came to a crashing halt three times in a year from a nasty virus, leaving me unable to get my work done. Once I was supposed to give a test the next day, but without a working computer to type my exam that didn't happen. While my students weren't exactly upset about this (the little buggers actually seemed happy for some reason!), I was annoyed, to say the least. To say the most, I'd had it.

I had tried Linux before, but those experiments had been only an exercise in pleasing my inner geek. Years ago, I tried Red Hat and found it promising, but I didn't have the time or patience to face the learning curve. Over the next few years, I tried out many other distributions: Corel Linux, Fedora, Mandrake, Xandros, Debian, and Slack. While I was intrigued and impressed, none of them had convinced me to make the switch from my Redmond-centric OS.

But, thanks to my latest Trojan-induced disaster, I knew the time had come. Linux, after all, was virtually immune to viruses and spyware, two problems that had cost me far too much time and productivity. Teachers don't have enough time as it is. Preparing for class, grading assignments, attending meetings and trainings – a teacher's plate is full enough without having to wipe and reinstall your computer's operating system twice a year. And while I diligently backed up my data in anticipation of the next disaster, I always seemed to lose a few files, including some that had to be recreated at a cost of time I didn't have to spare.

I knew Linux could provide stability and stave off many of the dangers to which other operating systems fell prey. I also knew, thanks to the many bootable CD versions of Linux I'd tried over the past several months, that Linux had reached a level of usability that would more than meet my needs. But I had two problems to solve before I could switch my home computer to a Linux system.



First, I love computer games. My students always found it amusing that their teacher could kick their butts in online first-person shooters, or that he could lose himself in a good RPG. I've played PC games since the days of Wolfenstein 3D, so, unless Linux could scratch my gaming itch, it wasn't going to keep me happy.

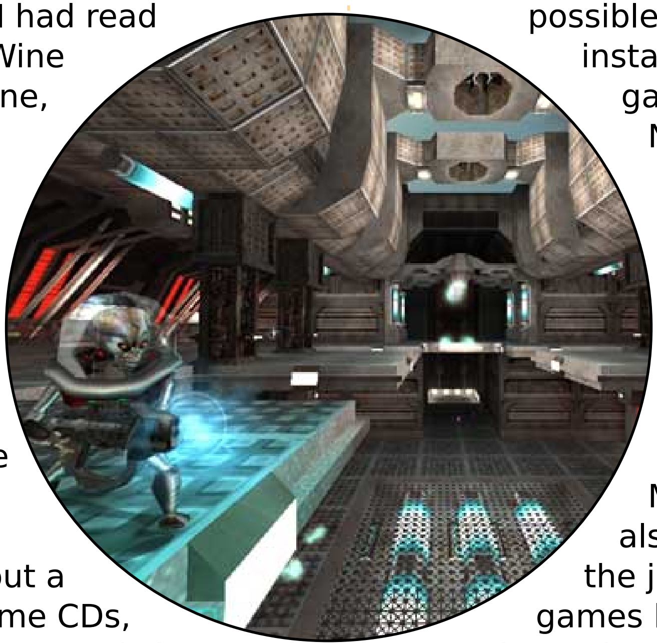
My second problem was my family. While I was geeky enough to tackle Linux, my wife and kids were another matter. My two sons, like their dad, were avid gamers who would need their gaming fix. My wife, however, who is also a teacher, saw the computer as a necessary evil. While she enjoyed email and online Scrabble, the computer was more of an irritant for her than a labor-saving device. Routine

contents ^

tasks - like typing a document in a word processor - often turned into an exercise in frustration. Clearly, if Linux was going to work for my family, it had to be user-friendly and a viable gaming platform.

Ubuntu Linux was my best hope. Of all the distributions I'd tried over the years, none had matched Ubuntu's elegance and ease of use. I downloaded Ubuntu 7.04 on a friend's computer, burned it to CD, took it home and booted it up. The installation was straightforward. My wired network connection was detected and ready to use, and, thanks to Synaptic, it was easy to install the needed plugins for Firefox that would ensure my wife's online Scrabble habit could be fed. In hindsight, my worries about my wife's ability to handle the switch to Ubuntu Linux were proven wrong. Once she realized that OpenOffice.org could handle her teacher tasks with aplomb, and that Firefox and Thunderbird made online tasks fast and easy, she never looked back. She is now completely comfortable with our Linux computer; in fact, a few months ago when her classroom computer crashed, she came home from school to tell me that when the computer tech came to fix the problem, she informed him that, if the school would just switch to Linux, this sort of thing wouldn't happen!

Gaming was my next challenge. I had read about the Wine project online, but I had never tried it. I used Synaptic again and installed Wine, which to my surprise took only moments. I then took out a stack of game CDs, did a Google search for Wine gaming how-tos, and set to work. To my surprise, I found that many games worked just as they did in Windows. Installing World of Warcraft, for example, was as straightforward as it had been in Windows. Guild Wars proved to be just as easy. Older games, however, such as my much-beloved Thief and Thief II games, were not responsive. After more online reading, I decided to download and install Transgaming's Cedega software. While Cedega isn't free, the five-dollar-a-month fee brings updates and features that have made it possible for me to install the Thief games, Neverwinter Nights II, Bioshock, LucasArts' Jedi Academy, and numerous other titles. My sons and I also discovered the joys of Linux games like Nexuiz (pictured on previous page) and Alien Arena (pictured), which have proven so entertaining that they have replaced Unreal Tournament. In short, while Ubuntu might not be capable of running every game new or old, between World of Warcraft and my Neverwinter Nights my kids and I are happy. Thanks to Wine and Cedega, my sons and I continue to frag, quest, and save the world.



That was two years ago. We now have two networked Ubuntu 9.04 computers, and I am typing this on my Dell laptop running Ubuntu 8.10. In two years, none of my computers has crashed or fallen victim to a virus or spyware program. My sons and I still game daily, and we are becoming increasingly adept at tweaking Wine and Cedega to run them well. My wife still plays online Scrabble, and finds her work on the computer less stressful than it was in our pre-Ubuntu years. In August, I am presenting a workshop for teachers in my district of Florida on the advantages of open-source software. Soon, I may not be the only teacher writing you to sing the praises of the penguin!

contents ^

MY STORY

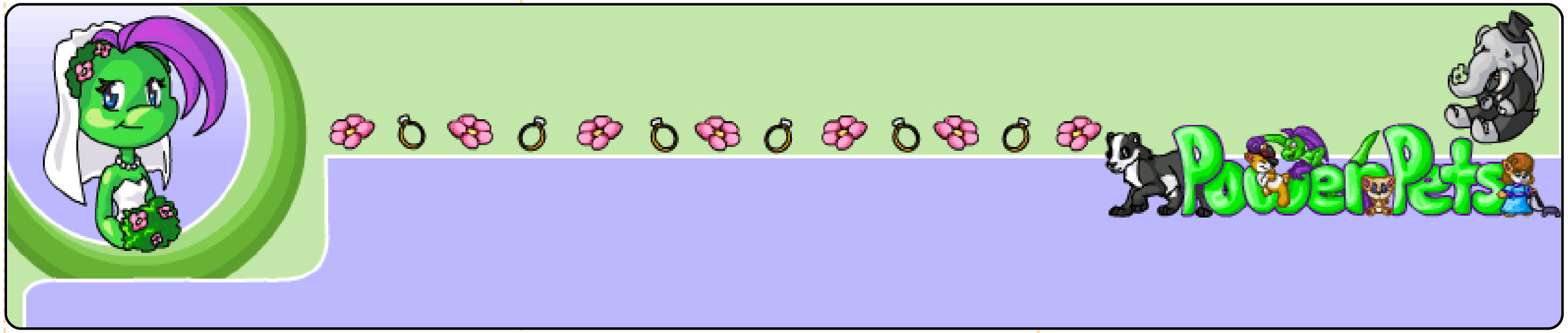
Written by Elizabeth Routliffe, President, Powerpets.com

Powerpets.com

In 2002, we started Powerpets.com, a Canadian company based on a small idea and big dreams: to create an entertaining virtual world with many educational aspects to support and spread awareness of abused and abandoned animals.

Less than half a year into the project, we were experiencing explosive growth and an equal number of problems - mainly e-mail based viruses. We just happened to be affected by the world’s next-big-virus threat weeks before any large anti-virus companies such as McAfee had a fix for it. More time was spent fixing our Windows office computers than working on our business.

When we switched to Linux we were keenly aware that we would face quite a few hurdles, but it was nice to know that once the system was set up we wouldn't have to worry about it. We wouldn't even need to restart it. Unfortunately, last year the flavor of Linux that we had chosen for the office was sold to another company who pretty much abandoned the OS.



A great deal of research then went into selecting a new Linux OS to carry us through the next decade without headaches - or an overload of IT expenses. Ubuntu quickly caught our attention. An additional month passed before we requested an Ubuntu CD, because we couldn't find an option to purchase one and felt uncomfortable ordering a free CD. (With satellite Internet, we were unable to download the ISO image.) A few weeks later, our CD arrived in the mail. As soon as we had some free time we installed Ubuntu on an old (test) laptop. The installation was quick and without a hitch, despite the outdated hardware. Subsequently, everyone in our office installed Ubuntu and was using it in a matter of hours. Some changes were hard to get used to, such as the SAMBA shares, but the switch to Ubuntu was an overall success.

Many thanks Ubuntu for sending us the CD and for our great experience with your operating system!

contents ^

MY OPINION

Written by William Arledge

9.10 Issues

After reading Maciej Miller's letter in the November Full Circle, I just had to add my two cents. For me, the change from Ubuntu 8.10 to 9.10 has been a deep disappointment. After weeks of frustration, I'm pretty much unhappy with the whole thing. After backing-up and removing most of the hidden folders and files in my "home", I did a fresh installation of Ubuntu 9.10, Kosmic Kludge, to the same partitions I've been using for years.

Boot

The log-in screen is, at a minimum, corny. The whole scheme is flawed: first a mouse click to confirm the user, then change to the keyboard to enter a password, and then change back to the mouse to confirm log-in; but, if you use a left-handed mouse, the system can't figure out what you're doing even though it knows who the user is! If Ubuntu 9.10 boots faster than 8.10 I can't sense it. Of course, it is worlds faster than XP or Vista.

Grub

Apparently Grub2 replaces a program that has served well - simply for the sake of making a change. It is impossibly complex to configure. Grub2 still thinks that my HP/Compaq system restore partition is Windows 2K. But thinking that HP/Compaq's Windows XP partition is Linux is completely inexcusable!

Disk by uuid

I'm supposed to know that 409c796f-29c1-4336-8e7a-1ed18db8706f is really hda5! It really makes no sense, to me anyway, that resizing hda5 generated a completely new uuid for the same chunk of disk real estate. Of course that made the partition unavailable to Ubuntu.

If I plug in one of my pendrives, the desktop display shows "4 GB Filesystem" below its icon. Nautilus' Places->Computer displays roughly the same information. But Nautilus' folder display title bar shows "881A-9238". /ls/media gives 881A-9238 for my PNY pendrive, BCC4-4D1A for my Toshiba pendrive but CRUZER for my SanDisk. I don't find that informative.

Again, complexity replacing relative simplicity for the sake of change and "correctness".

Suspend and Hibernate

On Windows XP, my box suspends and hibernates without problems, but Ubuntu 9.10 can't handle these operations. At first, I thought the problem was that I had increased RAM from 500MB to 1GB leaving /swap at 900 MB. Finally, I worked up the nerve to resize the partition using Gparted Live. Wonderful application - simple, quick, it just works! That's when I ran into the uuid fiasco mentioned above - couldn't figure out why I had no /swap partition! So now, with a 1.5 GB /swap, Ubuntu still can't suspend or hibernate. Interestingly, a prior version of Ubuntu, maybe 7.??, was able to hibernate - now that's progress?

Video

Now I admit to having an "old" video card featuring an nVidia Gforce2 MX400 chip. But hey, I don't play video games - it handles Stellarium's needs just fine. I can't understand why I have to use nVidia's drivers and enable a bunch of silly desktop effects just to get a display that even works. If I don't, the "system" insists on using a VESA driver and a window manager that don't work - yes I've filed a bug. In that respect, 8.10 was head and shoulders above 9.10!

9.10 still happily paints files/drives on top of other files/drives in the initial workplace display.

Audio

The main reason I "upgraded" to 9.10. I can play music and hear the audio from videos, but, frankly, I'm afraid to reinstall Skype, or Wine/EchoLink, or even see if my microphone works. I just don't need any more hassle.

Internet

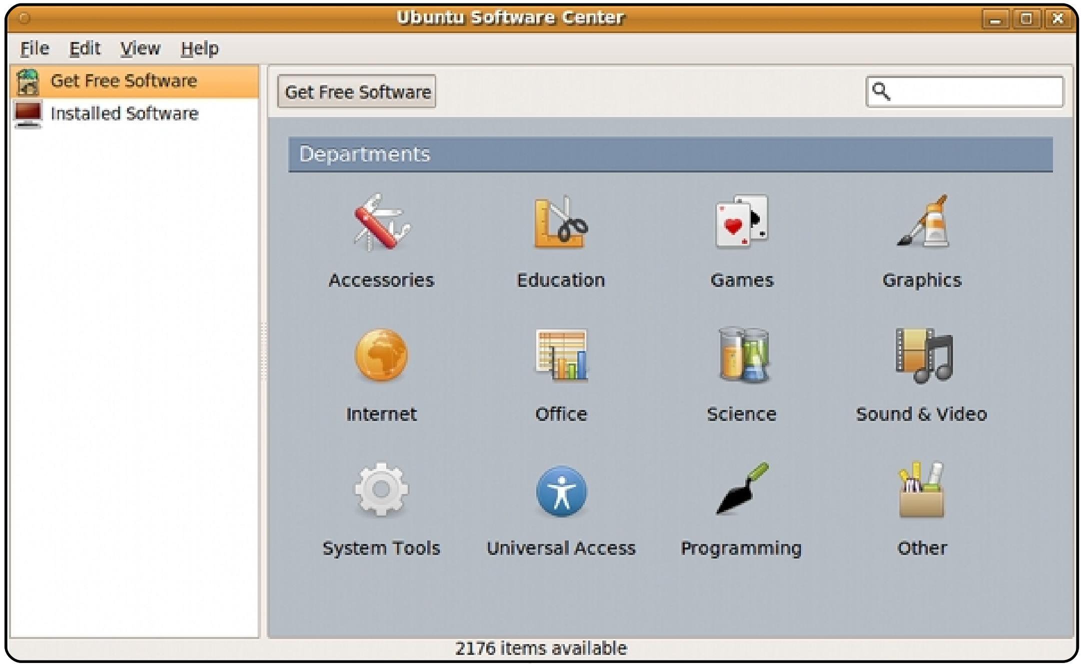
Somehow, 9.10 has introduced a 7-10 second delay when an application tries to access a URL - every URL. I use Thunderbird, Firefox, and Seamonkey - they all manifest the same problem. I spend way too much time watching 'waiting for ...' down on the status bar. 8.10 didn't have this problem. The version of eeebuntu on my EEEPC is based on 9.04, and it doesn't have the problem. I can't even figure out what to file a bug against for this issue.

This bug causes "collateral damage"! Any application (e.g. Wine) that tries to install the Microsoft Core Fonts complains that it times-out - trying after 10 minutes or so. A workaround removes part of the install script having to do with time-outs.

Default applications

Why anyone would choose Evolution for the default mail and calendar application is completely beyond me. Like most Swiss Army Knife solutions, it fails to satisfy needs in almost every respect. I tried - really tried - to use Evolution for several months on 8.10. The only thing going for it is that it can back-up and restore the mail folders. I won't even get into the mess that Evolution made of my Palm!

I've never been able to understand F-Spot - I can't get it to do anything. If I want to organize or edit photos that exist, I understand that they have to be "imported". The "import" function doesn't give me a choice of folders to import from - just my Windows partition or mounted USB drives. If I open a folder with F-Spot, I can view thumbnails and photos, but there are no editing features. G-thumb seems to be a more able and intuitive application for the purpose.



I do like the new Ubuntu Software Center, though (pictured).

To me it seems that Canonical has drifted off into the 'sell the sizzle not the beef' mindset. In my opinion, the developers should spend the entire time from now to release of 10.04 in resolving various issues such as these. I do understand that it is way more fun to do something new than it is to fix some niggling little issue - but it must be done. It just works - yeah right! Hey Maciej you forgot to mention openSuse as a viable alternative.

UBUNTU 9.10 KARMIC KOALA
BRAND NEW RELEASE!
UBUNTU user
EXPLORING THE WORLD OF UBUNTU
GET KARMIC
What's new in Ubuntu 9.10?
...ment Uncluttered:

Written by Lucas Westermann

ASUS UL30-A Laptop

SPECIFICATIONS:

- Intel Core 2 Duo SU7300 ULV processor @ 1.3GHz
- 4GB of DDR3 RAM (DDR3 1066 2GB*2)
- 320GB hdd (5400R, SATA)
- 802.11b/g/n + Bluetooth
- Ethernet (Atheros AR8132 – uses ath9k module) 10/100
- Intel X4500MHD
- 13.3” HD LED Monitor with integrated 0.3MP webcam
- German keyboard
- 8 Cell 5600mAh battery
- Model: UL30A-Q0229V
- Dimensions: 12.7" x 9.0" x 0.9-1.1" (323 mm x 229 mm x 22.9-27.9 mm)
- Weight: 3.7 lbs (1.7 kg)
- Ports: 3 USB 2.0, 1 VGA, 1 HDMI, 1 RJ45 10/100 Ethernet via Atheros AR8132, 1 analog headphone output, 1 analog microphone input

Cost: 679.00€

I recently took the plunge and bought a larger CULV (Consumer Ultra-Low Voltage) laptop (13.3” as opposed to my 10” Samsung N110 netbook). After having it for a little under a month, I can't really say I've tried out everything, but I have definitely checked the vitals in Ubuntu and Arch Linux. I figured that other people might be interested in the laptop, so here I am, writing this review. For a quick summary, check the conclusion of the article.

My first impression was that the laptop looked a lot nicer than any that I owned before it - it's silver brushed-metal on top, which makes it look and feel a bit sturdier than my netbook.

The OS it comes with is Windows 7 Home Premium 64bit, which I left on it, to dual-boot. In Windows, the battery offers me between 10 and 11 hours of normal usage (read: occasional video, lots of typing, surfing the Web, music, etc.). So, I was aiming for at least 9 hours of battery life in Linux (I've learned to expect a little less battery life). The version of Ubuntu I installed was 9.10 x86_64, and once I got it installed off the USB stick, I noticed that a) I had Bluetooth (the laptop was advertised as without Bluetooth, so that was a nice surprise!), and b) it was running at native 1366x768 resolution (with Compiz). So far, so good. Connecting to my WPA2-TKIP wireless network went off without a hitch, and ethernet had no issues running at its max (100mb/s) speed.

Once I was connected to the 'net and updated, I decided I'd install Ubuntu-restricted-extras, and check the Flash playback. Flash ran smoothly at 360, 480 and 720p You Tube playback (it did get a bit warm doing HD playback, but that's Flash's fault), and Flash sites seemed to run smoothly as well. While the video was playing, I decided to also try brightening/dimming the screen (and turning the backlight off), and each of the Fn keys responsible for those functions worked out of the box, as well as the Fn keys for volume. I didn't test the wireless on/off key (I never use it), nor did I try the custom Asus-software keys (I doubt they'd do anything). Media playback keys (play/pause, stop, back/forward) are recognized by Ubuntu, but it does require a media player that uses them - I use MPD, and a custom function bound to the keys for controlling playback. I also checked to see if the monitor model was correctly recognized - Ubuntu hadn't done that for any of my computers before - and was shocked to see that the model was listed there, and the default settings were chosen as recommended by the model number. It may seem like a small thing, but it was new to me.

contents ^

Suspend/resume was the next on my list to try and, lo and behold, both worked without a problem (I didn't try hibernate, as I skipped setting up a swap file/partition). I figured that since suspend/resume worked without a problem, it was time to try it on battery power, and see what Gnome-Power-Manager told me - so I removed the A/C adapter, and gave Gnome-Power-Manager a moment to update, then checked the reading. With the brightness as low as it went, Gnome-Power-Manager was telling me I had 10 hours of battery life left (sometimes going up to 10 hours and 30 minutes). I realize it's not accurate, but even if it's off by about an hour, that's more than enough time for a full day of use for most people, so I left it at that. I have tried it more extensively in Arch (since it's my Linux of choice), and it does indeed give me about 10 hours of battery life using custom ACPI handler.sh settings, so I imagine it's possible in Ubuntu as well.

I also tested the Bluetooth functionality with my phone (a Motorola RAZR V8), and it worked fine. The only thing I found an issue with is the webcam: Cheese, Skype, aMSN, etc. all recognized and used the webcam fine, but the resulting image was upside-down. I'm sure it's a module issue (doesn't happen in Windows), and it may be fixed sooner or later. It's not too big a deal, however, because the laptop's webcam of only 0.3 megapixels makes me doubt that anyone will buy the laptop for heavy webcam usage. If I remember correctly, Cheese offers the ability to rotate/flip the webcam input (similar to photobooth on Mac OS X), so taking pictures with it should be fine - you can rotate it after the fact as well, so no issue there.

The only device I didn't test (at least, the only device I know I didn't test) is the microphone. According to alsamixer and the Gnome sound applet, it's recognized as a valid input device, but I did not try it in anything, because I never use microphones. Chances are good it can be made to work - if it doesn't work out of the box. If anyone is thinking of buying this laptop, and needs to know if the microphone works, I would be willing to respond to an email query, and to make a short note in one of my articles for anyone else interested. However, I don't know how many people really use a laptop's integrated microphone in the first place.

One last note - which was a concern for me at first - is that Express Gate (Asus's 8 second boot OS) still works after installing GRUB, so you can still have quick and easy access to Skype, the internet, etc.

To sum up: Everything works out of the box except the webcam, which presents images upside-down (I didn't test the microphone). The 8-cell battery gives 10-10.5 estimated hours of usage (according to Gnome-Power-Manager). All in all, I'm extremely happy with the laptop. Since I bought it, I've found less and less reason to turn my PC on, which is nice. I also haven't found anything I do regularly that has really put a strain on it. HD video (HD Revision3 shows mainly) plays without a hitch, and it delivers a consistently smooth user interface. If anyone is looking for a thin, light-weight laptop, I highly recommend this one!

contents ^

MOTU INTERVIEW

Taken from behindmotu.wordpress.com

Roderick Greening

Behind MOTU is a site featuring interviews with those known as 'Masters of the Universe' (MOTU). They are the volunteer army of package maintainers who look after the Universe and Multiverse software repositories.

Age: 37
Location: St. John's, Newfoundland, Canada
IRC Nick: rgreening

How long have you used Linux, and what was your first distro?

Well, it all started back in University, and that would have been around '93. I was working in one of the computer labs, and a friend of mine (TC) said he had this cool new OS that I should try out. He indicated it was very UNIX-like, and, since we were using UNIX servers for most of our Computer classes, it would be useful. The distribution was Slackware, and I believe it came on approximately 15-20 floppy disks. I can't tell you how many hours I spent playing around in that environment. What I can tell you is that it was directly responsible for my current career path, and my first full-time job in the computer industry. I maintained a dual-boot system for many years at work, just to run Linux for 'real' work, and Windows for everything else I 'had' to.

How long have you been using Ubuntu?

I started using Kubuntu in 2006, though I dabbled with it a little bit earlier than that. Prior to that I was using Gentoo as my main OS, and did so up to about mid 2007. My main reason for looking into Kubuntu was I had a friend who was quite dissatisfied with Windows. He was constantly having to re-format and re-install his OS to get rid of mal-ware, viruses, and other cruft. I eventually convinced him to test-drive Kubuntu along with me. Since then, I have installed Kubuntu for many friends and family members, and they now use Kubuntu as their main OS, and are quite satisfied.

When did you get involved with the MOTU team and how?

One day, while using Kubuntu, my Knetworkmanager stopped working. In order to get back online, I needed to install the Ubuntu nm-applet. I decided then to see if there was something I could do to help troubleshoot it. I jumped onto IRC and joined the #kubuntu-devel channel and started asking some questions. I was amazed at how helpful everyone was, especially Jonathan Riddell, Scott Kitterman and Harald Sitter. These three devs really got me kick-started, and back into development mode. This was during the tail-end of the Intrepid development cycle. To quote Jonathan from his Kubuntu 8.10 release announcement: "Then there's rgreening who turned up one day, and fixes anything anyone asks him to since". Since then, I have packaged many releases and updates to KDE4, as well as other KDE Universe packages. I had really good sponsors, who were patient and helped me out a lot. In particular, Scott Kitterman was exceptional. On many occasions, he offered up little known (at least by me) tidbits on how to streamline my packaging and testing. Without him, I do not think I would have applied for MOTU status.

What helped you learn packaging, and how Ubuntu teams work?

It was a combination of reading documentation, asking the right questions via IRC devel channels, working with great sponsors (see above), and looking at other people's work. Having such a large community is definitely a real benefit to Ubuntu. It is quite easy to find the answer to your question, usually within minutes of looking or asking.

What's your favorite part of working with the MOTU?

contents ^

It's definitely the people. Not only is everyone really knowledgeable, they are also very approachable. Never be afraid to ask a question. It's how I got to where I am today.

Any advice for people wanting to help out MOTU?

1. Don't be concerned whether you have the required skills. If you can install and use Ubuntu, you can help QA packages, submit bug reports, and test drive new releases.
2. Dive right in. The best way to learn is by joining a team you are interested in, and helping them - whether it's testing, writing documentation, etc.
3. Get a mentor to help guide you and sponsor your work.
4. Have fun.

Are you involved with any local Linux/Ubuntu groups?

There isn't a LOCO here in Newfoundland, though a couple of us are starting to seriously consider starting one up. I know there is a Linux community here, and it's been around for a while now, though I believe they mostly use RPM-based distros. At any rate, I'm sure I'll be contacting 'the Bacon Community' in the near future, once I get some spare cycles.

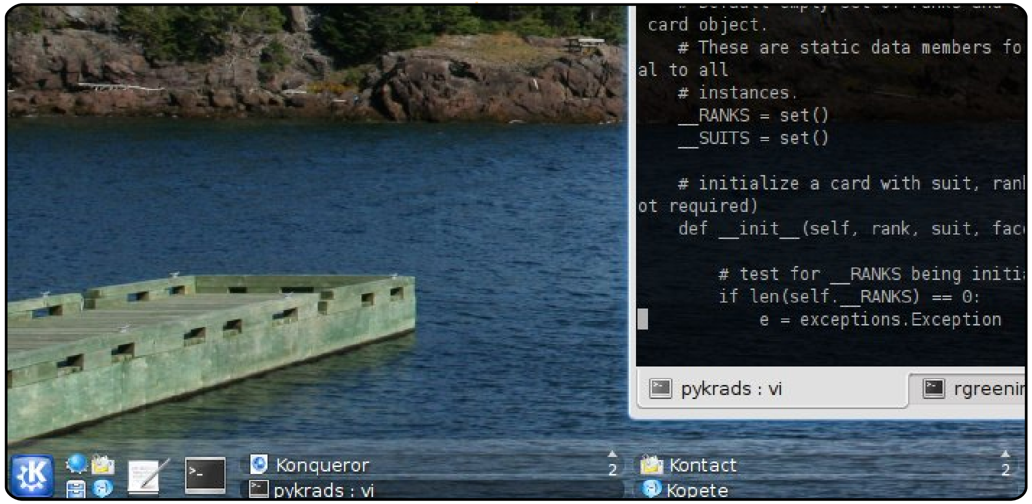
What are you going to focus on in Jaunty and Jaunty+1?

Well, right now I have been working on packaging KDE 4.2.1, and will also assist in KDE 4.2.2 packaging. Outside of that, there are improvements to KPackageKit I have implemented, and still some additional things to come. For Jaunty+1 (Karmic), I have a couple of packages I am developing myself, which I would like to see make it into Universe and/or Main. There are two applications which are security related (ufw-kde and clamav-kde), as well as a new card game written in PyQt. Additionally, I have been thinking about some work in the server-team, and would like to help them develop a package that would help new admins set up a server with proper TCP wrappers, fire-walling, logging, root-kit checking, and log analysis. Currently, these tools exist, but you have to install and configure each one separately. I'd like to bundle a set of these apps (meta package), and help the user configure some sane choices. At this point it's only an idea, but with UDS looming ahead, who knows where it might lead.

What do you do in your other spare time?

Spare time? Oh, you mean if I had a personal life outside Ubuntu? Well, I am an amateur photographer. I have a Canon Rebel XT, and a few lenses. I am mostly interested in scenic/landscape photography, but I also dabble in portrait shots. I also love to read. I have hundreds of books in my personal library, mostly Sci-Fi and fantasy. I am a really big fan of Star Trek, Star Gate, and Forgotten Realms. My collection also includes a substantial number of technical books on programming and networking. Outside of that, I spend time with my wife, and dog, Jewel, and hang with my friends playing Mario Kart (baby Luigi/quacker) and Settlers of Catan ("monopoly wheat").

My current desktop is running Kubuntu with the Glassified theme. The picture in the background was taken by me 2 years ago, while visiting my parents. It's a small inlet, called Deep Cove, and it's in Musgravetown, Newfoundland, my family's hometown.


```
card object.
    # These are static data members fo
al to all
    # instances.
    __RANKS = set()
    __SUITS = set()

    # initialize a card with suit, ranl
ot required)
    def __init__(self, rank, suit, fac

        # test for __RANKS being initi
        if len(self.__RANKS) == 0:
            e = exceptions.Exception
```

contents ^

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 reasons.**

Graphic Questions

Regarding Ubuntu planning to give up GIMP begs the question: what will they replace it with? Surely they must replace it with something as Ubuntu can't be without some sort of graphics application. I agree that the GIMP is perhaps too fancy for most of us non-graphics professionals, so perhaps we need an open-source version of Photoshop Elements?

On another related tack, it would be great if Ubuntu would automatically offer to reduce the size of photos when e-mailing as Windows does. At the moment, when you want to e-mail photos you have to open each one in GIMP, reduce the size and then attach it to your message. This is a bit of a pain when sending multiple photos.

Chris Burmajster

Windows 7

On my new Gateway SX2800 ($400) with Windows 7 Home Premium the system boots a good bit faster than Ubuntu 9.10 - that's with the AVG free anti-virus program, the Windows firewall, etc. Firefox 3.5.6 shows my homepage in about 2 seconds, Thunderbird 3 opens in about 3 seconds - yes loading current content takes time but nowhere as long as it does on Ubuntu 9.10. OO Writer and Calc both open in less than 4 sec. Oh, and may I mention that under Windows 7 the system will actually "sleep" and recover from it?

The assertion that Windows 7 won't run older apps is just plain wrong. I run FreeBASIC and FBIde, both relatively old 32 bit aps. The same version of Ancestral Quest 12 runs on XP, Vista and 7 without issues - same for PAF5.2. I run CADKEY 99 on 7 without problems - the

LETTER OF THE MONTH

Writer of Letter of the Month wins two metal Ubuntu case badges!

Regarding FCM#31, page 26, I can't avoid replying to this one, since I know the answer! What must be done for burning a DVD which is recognizable in this way is to have a folder where all contents of the DVD will be burnt. Inside this folder, we create a 'pool' directory. inside of it we have all the .deb files we need.

In a terminal at this directory, we do: (assuming apt-ftparchive command is installed)

```
apt-ftparchive packages
pool/ > Packages

apt-ftparchive packages
pool/ | gzip > Packages.gz

apt-ftparchive packages
pool/ | bzip2 >
Packages.bz2
```

It's important that all three of these files are created completely, since these files have all the database needed from apt-get (and Synaptic as well) and later, we will need more 3 files:

./.disk/info
./aptoncd.info
./README.diskdefines

A good way to check your layout is to compare it to some Ubuntu install CDs (also named Alternate CDs) - check the differences, and be sure about what to edit in the three files mentioned above. Of course, check if all info inside these files is correct, to avoiding some unwanted surprises. At this point, we may have at './' :

./pool (folder with .deb files inside)
./Packages
./Packages.gz
./Packages.bz2
./.disk/info
./aptoncd.info
./README.diskdefines

Now is the time to burn the CD or DVD - if burning the folder using Brasero it may show errors, so you may need to make an ISO, and burn that to CD/DVD. After burning the CD or DVD, it's time to check if it behaves just like an installable one, just like an alternate or extra Ubuntu CD or DVD - and having stuff there like all Ubuntu repositories inside!

Paulo

same program (CD) I ran on Windows 98 Second Edition. The only program I've tried that just flat wouldn't run is the old MS Professional Basic compiler.

Sorry, I disagree with most of the assertions made in the article. Personally I think that Ubuntu 9.10 was a disaster. So much of a disaster that I'm actually converting to Windows 7 after 4 years of exclusive Ubuntu use. What few Linux apps I have to use run on WUBI as well as anything - surprisingly well in fact!

David Rowell

Ubuntu Win

My daughter phoned me up one day and said her hard drive wasn't working. She said she couldn't get her all important baby pictures off the drive to save to a CD. I took over an Ubuntu 9.10 live CD and ran it, mounted her Windows hard drive and we were able to burn all her pictures to CDs. Ubuntu to the rescue. She and her husband were impressed. Thank you Ubuntu.

Robert Coulson

Windows Convert

I'm writing to you to share my experience with Ubuntu. I started using it at home, but because I share the desktop with the rest of the family, I was forced to move back to Windows XP. I really don't believe in dual-booting or Virtualization as a permanent solution, because I find them to be a waste of hardware resources.

Although I have a genuine Windows license, the XP charm began to wear off, faced with the evolution of its peers. I charged again with the Karmic Koala, added some "nice to have" desklets, the Cairo Dock, and I was able to dazzle my wife into Linux.

As far as the bundled apps go, I added only the Chrome browser and Picasa desktop tools. The hardest part with this migration was to get my printer working properly. My next challenge is to set up a video capture USB device I have.

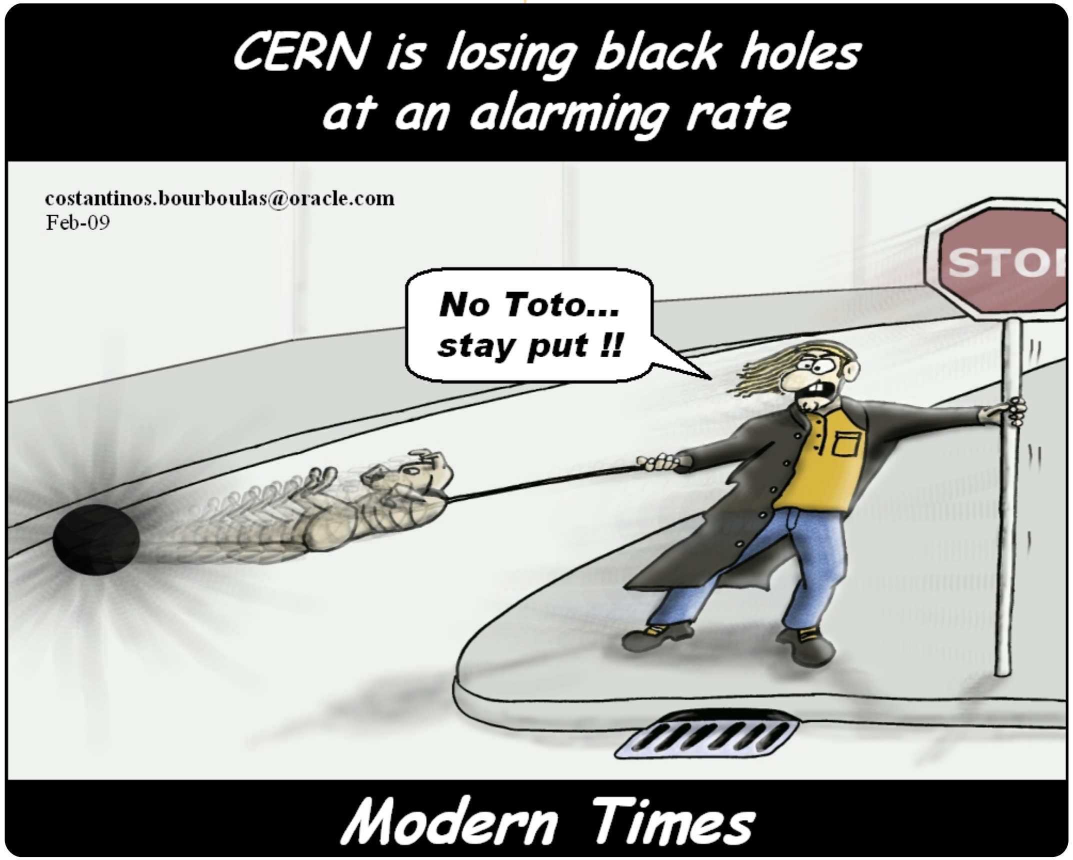
I agree with all the folks that state that Linux gives the fun back to computing.

Manuel João Osório

Spreading FCM

Whenever an issue is released, I spread the link to everyone I know via Facebook and Twitter using the handy 'Share links' on your site. I encourage other readers out there to do the same. Thanks for a great mag, can't wait for the next issue.

Wesley Werner


CERN is losing black holes at an alarming rate
costantinos.bourboulas@oracle.com
Feb-09
No Toto... stay put !!
STOP
Modern Times

Written by Penelope Stoew

Melissa Draper Interview

Melissa thwarts Fedora by stealing CDs at an educational expo in Sydney

Penelope Stowe: Please tell us a little about yourself.

Melissa Draper: I am Australian. I have lived my entire life in NSW, but I was born in Victoria.

Once upon a time, I wanted to be an environmental scientist, but, during one of the projects for a related course, I ended up with the task of creating a website. I learned HTML one weekend, and realized I preferred computers to trees.

PS: How did you get involved with Ubuntu, and free software in general?

MD: Whilst doing a web-development diploma, a teacher, a BSD fanatic and probably the most influential teacher I ever had, was in the habit of interpreting syllabi modules such as "install an operating system" fairly openly. He taught his students how to install Linux and BSD, as well as Windows (amusing aside: he also taught my mother to do the same!), and so began my obsession with Linux. I played around with live distributions for a while, then installed Ubuntu on an old computer. I soon found I did not need XP any more.

Another teacher was supposed to teach us ASP.net as per the whole web-development thing, but was unable to teach for peanuts. Essentially, the entire class taught itself PHP instead of ASP.net, and, following the lead of the OS teacher, argued that it was still valid under "dynamic websites." In the end, I don't think I even got around to buying the ASP software. Free software is accessible like that, and php.net as a resource is awesome.

PS: Why Ubuntu?

MD: When I decided I wanted to use Linux, I asked around. A friend had been using Ubuntu since Warty, so I installed Hoary on an old computer. I've tried other distros, and, despite using Fedora or CentOS for work, I've been with Ubuntu ever since.

PS: You've been around for several years in the Ubuntu community; what projects are you involved with or have you been involved with in the past?

MD: I was quite active on the marketing team for a while, writing for UWN, etc. I also did a lot of coordinating to lay the foundations for the LoCo ecosystem we have now. Now, I do IRC management and Membership Board for Asia/Oceania.

PS: You've worked hard with Ubuntu Women, including applying for Ubuntu Women leadership. What would you like to see from the project?

MD: I'm hoping to have more visible initiatives to balance out the current focus on commentary. Commenting on bad behavior is a critical part of the aim, and I'd rather not have the attitudes against it continue. I'd like to see folks within the team engaging more with matters such as enhancing the wiki pages and conference materials, and contributing the attributes they want from the team - rather than dragging down the stuff that others put time and effort into making happen.

contents ^

PS: Are you involved in other activities outside Ubuntu relating to women in open source or computing?

MD: I'm involved in LinuxChix, and used to organize the local chapter's pre-LUG meet-ups. I'm also my LUG's secretary. I was also on the Linux Australia Council (the group that oversees the organization of linux.conf.au every year, and helps FOSS-based groups with grants, etc.) for 2 years, but I chose to not run for a third consecutive year as I felt I needed to take a break from that. I'm also running a Linux retail and web shop in Sydney, Australia called Everything Linux.

PS: Where do you see yourself going from here with Ubuntu and free software? What are your goals - both personal and for Ubuntu/free software in general?

MD: There's some odd politics going on, and I've been pushed away from a few things within Ubuntu lately. It's a complicated matter and I feel like I'm left holding loose ends. I'm still trying to figure how and where I stand with things. At this point, I intend to continue with Ubuntu Women as it's an important part of the community, regardless of what others think.

Ideally I'd like to see the Free Software community in general set the tone for how online communities can be run.

PS; Do you have any other interests or activities you'd like to tell us about?

MD: Currently I'm rather infatuated with fountain pens and good paper. I think the act of hand-writing actual paper letters is something we ought to try to keep relevant.

Full Circle Podcast

AUDIO MP3 AUDIO OGG

The ***Full Circle Podcast*** is back and better than ever!

Topics this episode include:
- the reshuffle at Canonical
- the online Canonical music store
- Google Buzz
- Good Old Games

and the Acer Aspire Revo.

Your Hosts:
- *Robin Catling*
- *Ed Hewitt*
- *Dave Wilkins*

The podcast and show notes are at:
http://url.fullcirclemagazine.org/88a985

UBUNTU GAMES

Written by Edward Hewitt

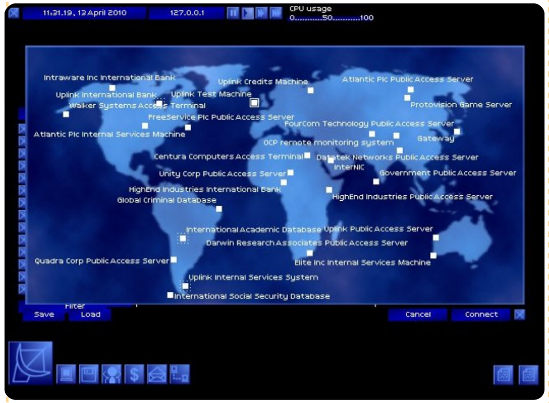
GAME NEWS

Rapid Wine Development – so far, in 2010, Wine has undergone major changes, which include Shader Model 4 support, Improvements to Direct3D, and tons of bug fixes.

Introversion Software has always provided excellent support for Linux. About a year ago, I reviewed one of their games, Defcon. Here is another of their games, Uplink.

Uplink is very different from any game you have ever played. The game is all about you being a freelancer hacker. You will be assigned jobs to infiltrate companies' computer systems to steal or change data. With the money you earn from the jobs, you will be able to buy tools and software to perform more challenging jobs. The aim is to become the 'l33t' (elite) hacker.

The game starts with you getting to grips with 'your new hacking operating system' - with all the tools you need to hack into computer systems, access your bank account, and access the store. To begin your career, you will be given a few simple missions, which will require you to use the password cracker and break into fairly simple computer systems - to steal data and then email it to your clients. Fairly soon, you will be attempting more challenging missions to change the data, crack a bank, hack a rival into jail, and crash the stock market. You will be needing a more powerful computer and a more advanced system. However, it's not as simple as cracking the password and performing your job - you also need to watch your back. Each connection you make is traced, so make sure you connect to plenty of different computers - the trace will take longer to track you down. Uplink is very good with progression - it is always moving you forward. However, after a few hours, and plenty of missions completed, it can become repetitive. All the missions are very similar - the way you begin the job, and what is involved with each job, are basically the same.


11:31.19, 13 April 2010
127.0.0.1
CPU usage
0...........50...........100
Intraware Inc International Bank
Uplink Credits Machine
Atlantic Plc Public Access Server
Uplink International Bank
Uplink Test Machine
Walker Systems Access Terminal
Protovision Game Server
FreeService Plc Public Access Server
FourCom Technology Public Access Server
Atlantic Plc Internal Services Machine
Gateway
OCP remote monitoring system
Centura Computers Access Terminal
Datatek Networks Public Access Server
InterNIC
Unity Corp Public Access Server
Government Public Access Server
HighEnd Industries International Bank
Global Criminal Database
HighEnd Industries Public Access Server
International Academic Database
Uplink Public Access Server
Darwin Research Associates Public Access Server
Quadra Corp Public Access Server
Elite Inc Internal Services Machine
Uplink Internal Services System
International Social Security Database
Filter
Save
Load
Cancel
Connect

The atmosphere of the game is excellent, with the dark desktop screen giving you a sense of the hacker working in a dark room, hidden away. The game music is '90s retro music, similar to the year the game is set in. The interface of the game is easy to manage, though connecting to different computer systems, and moving and deleting files, can be a bit clunky.

Uplink is unique. It can be enjoyed by all, but some people may see it as too repetitive and more like a day's work at the office rather than a game. I found it enjoyable to live out a 'geek's fantasy' as a computer hacker in the comfort of the game.

Score: 7/10

Good Points:

- Becoming a hacker!
- Unique game

Bad Points:

- Clunky interface
- Repetitive

Ed Hewitt, aka chewit (when playing games), is a keen PC gamer and sometimes enjoys console gaming. He is also on the development team for the Gfire project (Xfire Plugin for Pidgin)

contents ^

Q&A

Written by Tommy Alsemgeest

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

Q Whenever I login to Linux, two new files (New Document.ott and New Spreadsheet.ots) are automatically created with no content inside them. If I delete these files, and again login, these files are again created automatically. Why did this happen?

A Chances are you have something in your startup list. Take a look for anything unusual in *System > Preferences > Startup Applications*.

Q I am fairly new to the whole Server and Linux use, and could use your help. I wish to set up a box as a server with dual NIC cards, after my DSL Modem, and put my wireless/4 port switch behind it to which all my other computers connect. I would love the ability to shut off, or limit access time for my kids, and be able to better block their viewing content, as well as access my music files and pictures from any computer in the house, such as my media PC running Windows 7 Home Premium 64bit. Please help, or give links to where I can find such info.

A Your best bet for restricting access time is IPtables, which should let you restrict based on the IP address of the clients. It has quite a steep learning curve, but there is a very comprehensive tutorial, thanks to bodhizazen, which should have all the info you need: http://bodhizazen.net/Tutorials/iptables/. For blocking the viewing content I would suggest OpenDNS (http://www.opendns.com/), which will allow you to block based on category (phishing, pornographic, time-wasters), as well as add custom sites to your blocklists. And as for sharing with Windows computers, Samba is the most commonly used. This page should get you up and running: https://help.ubuntu.com/community/SettingUpSamba

----

Q I want to control my laptop (Lenovo R61i, Ubuntu Karmic Koala) via Bluetooth using cell phone (Motorola Q9H) and / or wifi using PDA (Palm Tungsten C), for example for presentation (OpenOffice Presentation) and multimedia (Rhythmbox / VLC). What applications should I install on my laptop and cell phone / PDA?

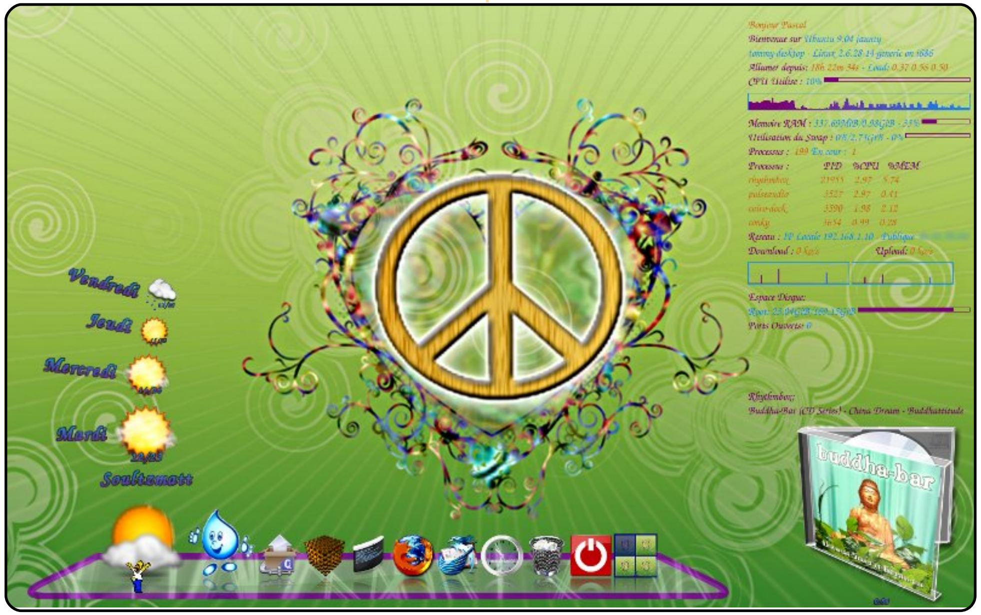
A The anyremote project (ganyremote and kanyremote in the Ubuntu software centre) should do what you need. Simply install it into Ubuntu, then install the Java remote into your Bluetooth phone and you should be able to control your computer.

Q I have a problem: I got the Ubuntu 9.10 Install CD and I want make an upgrade from my Wubi Ubuntu 9.04 desktop. How can I do it?

A Unfortunately, the Ubunu Live CD doesn't have the tools needed to upgrade, so you will need to either run the online upgrade, or download the alternate install CD, which does contain the needed files to upgrade. This page should have everything you need: http://www.ubuntu.com/getubuntu/upgrading

MY DESKTOP

Your chance to show the world your desktop or PC. Email your screenshots and photos to: **misc@fullcirclemagazine.org** and include a brief paragraph about your desktop, your PC's specs and any other interesting tidbits about your setup.



```
Bonjour Pascal
Bienvenue sur Ubuntu 9.04 jaunty
tommy-desktop - Linux 2.6.28-14-generic on i686
Allumer depuis: 18h 22m 34s - Load: 0.37 0.56 0.59
CPU Utilise : 10%
Memoire RAM : 337.69MiB/0.98GiB - 33%
Utilisation du Swap : 0B/2.73GiB - 0%
Processus :  199 En cour :  1
Processus :          PID    %CPU   %MEM
rhythmbox          21955   2.97   5.74
pulseaudio          3527   2.97   0.41
cairo-dock          3590   1.98   2.12
conky               3654   0.99   0.28
Reseau : IP Locale 192.168.1.10  Publique [illegible]
Download : 0 ko/s                Upload: 0 ko/s
Espace Disque:
Root: 23.04GiB/169.15GiB
Ports Ouverts: 0
Rhythmbox:
Buddha-Bar (CD Series) - China Dream - Buddhattitude
```

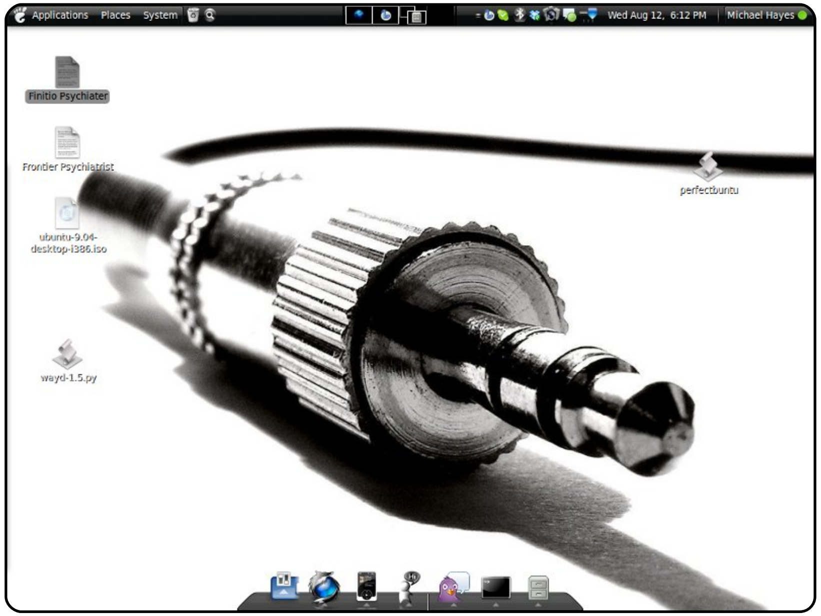
Hello, I present my 70's desktop (wallpaper found on devianART) with conky, Compiz Fusion, Emerald (fonts "Hippy Participants") and Cairo-dock + display album art 3D installed on Ubuntu 9.04.

My hardware configuration:

- Packard Bell PC P4 2.93 GHz
- Ati X300 graphics card
- 1GB of RAM and 1 HDD 200GB

So now everything is working perfectly even without the proprietary drivers for my card.

Pascal Olry

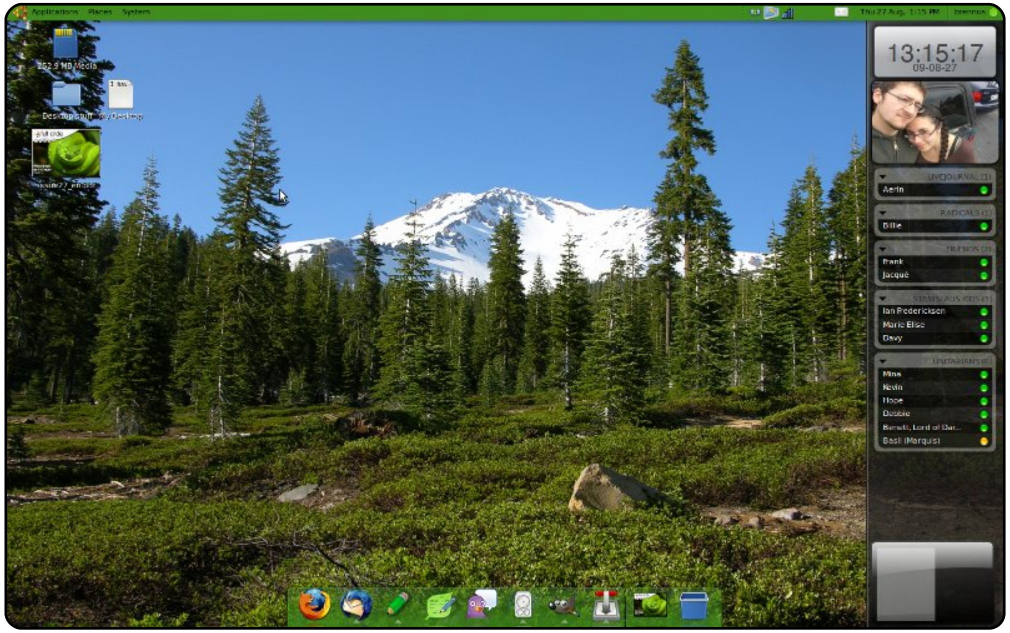


I am using the Moomex Metacity theme (from gnome-look.org) with Black-White 2 Gloss icons and _jack wallpaper (also from http://gnome-look.org). Also in the mix is the Avant Window Navigator and Compiz.

All this runs on Jaunty Jackalope on a six-year-old Dell 4600 with a Pentium 4 (single core!), Radeon 7000, and 2.5GB RAM.

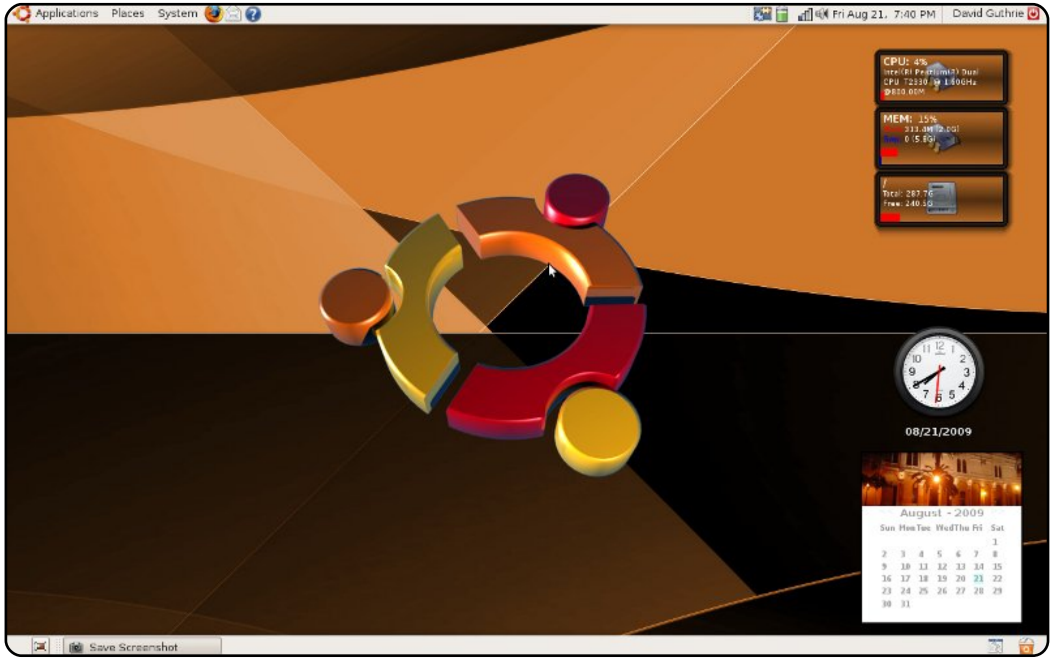
Thanks to the community for making such an awesome (and the best on the planet) distro!

Mike Hayes



I have a Dell Inspiron E1405 with dual-core 1.60GHZ processors and 2GB of RAM. I'm running Ubuntu 9.10, Jaunty Jackelope. I use Compiz. I use screenlets on my right (a clock, a picture frame, a pager, and the pidgin screenlet, all docked to a sidebar), and the Avant Window Navigator (AWN) as a dock. I use a custom theme built off of Clearlooks, using the Mist icon set and Gilouche window borders. My terminals are transparent. The applications on my dock are Firefox internet browser, Thunderbird email client, Drivel journal/blog client, Leafpad text editor, Pidgin instant messenger, Rythmbox music player, GIMP, and Transmission bittorrent client, with a trash bin on the ends.

Kenneth Sime



I started using Linux with Ubuntu 8.04. I have tried other Linux distributions, but Ubuntu is my absolute favorite. I use a Gateway laptop computer M-6309 Intel Pentium Dual-core T2330 GHz processor, 2GB memory, 320 GB hard drive, and I am using Ubuntu 8.10. The wallpaper is called Ubuntu 2.0 - available from http://www.gnome-look.org/. I tried using the Avant Window Manager dock, but I prefer to launch programs from the Applications Menu on the top panel. I am also using a desktop PC with Ubuntu 9.04. I love all the great applications available in Ubuntu. I never want to pay for software again.

David Guthrie

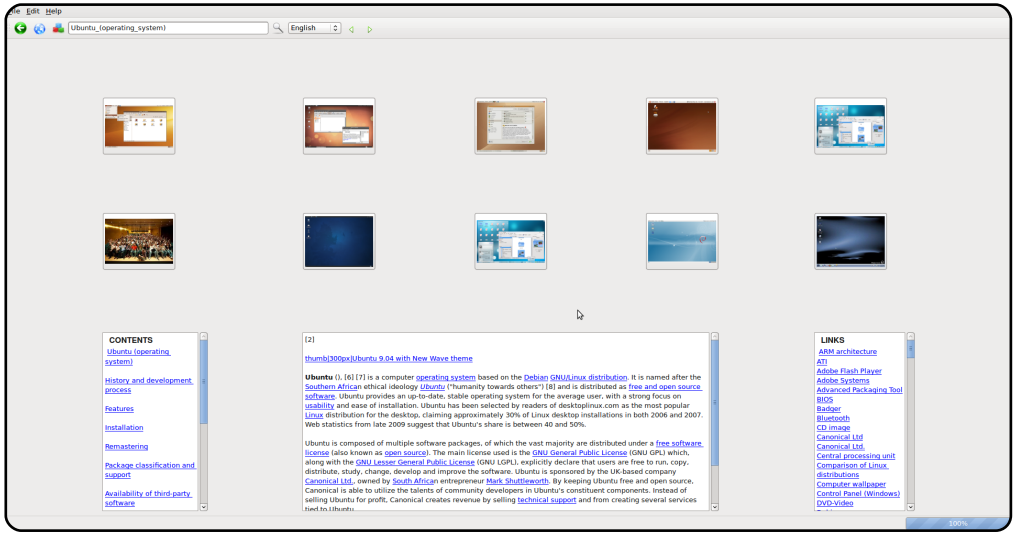
TOP 5

Written by Andrew Min

Reference Tools

Indywiki

http://indywiki.sourceforge.net/



CONTENTS

Ubuntu (operating system)

History and development process

Features

Installation

Remastering

Package classification and support

Availability of third-party software

[2]

thumb|300px|Ubuntu 9.04 with New Wave theme

Ubuntu (), [6] [7] is a computer operating system based on the Debian GNU/Linux distribution. It is named after the Southern African ethical ideology *Ubuntu* ("humanity towards others") [8] and is distributed as free and open source software. Ubuntu provides an up-to-date, stable operating system for the average user, with a strong focus on usability and ease of installation. Ubuntu has been selected by readers of desktoplinux.com as the most popular Linux distribution for the desktop, claiming approximately 30% of Linux desktop installations in both 2006 and 2007. Web statistics from late 2009 suggest that Ubuntu's share is between 40 and 50%.

Ubuntu is composed of multiple software packages, of which the vast majority are distributed under a free software license (also known as open source). The main license used is the GNU General Public License (GNU GPL) which, along with the GNU Lesser General Public License (GNU LGPL), explicitly declare that users are free to run, copy, distribute, study, change, develop and improve the software. Ubuntu is sponsored by the UK-based company Canonical Ltd., owned by South African entrepreneur Mark Shuttleworth. By keeping Ubuntu free and open source, Canonical is able to utilize the talents of community developers in Ubuntu's constituent components. Instead of selling Ubuntu for profit, Canonical creates revenue by selling technical support and from creating several services tied to Ubuntu

LINKS

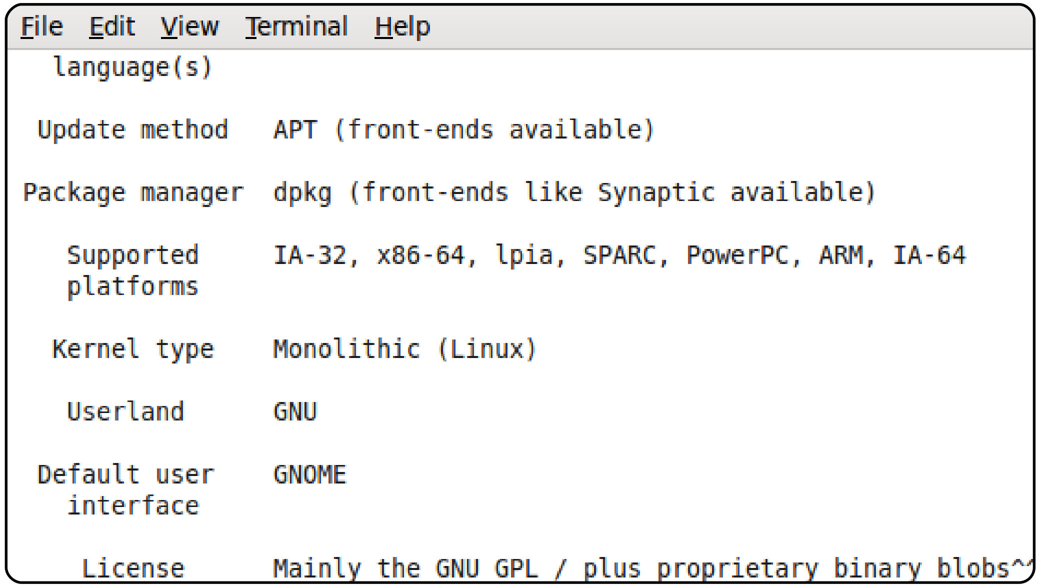
ARM architecture
ATI
Adobe Flash Player
Adobe Systems
Advanced Packaging Tool
BIOS
Badger
Bluetooth
CD image
Canonical Ltd
Canonical Ltd.
Central processing unit
Comparison of Linux distributions
Computer wallpaper
Control Panel (Windows)
DVD-Video

If Wikipedia is all the reference you need, check out Indywiki. It's a Qt interface to Wikipedia, with an emphasis on doing one thing well: graphic browsing. Open an article, and images from both the article and related articles pop up. Additionally, links to other entries referenced by the article show up on the right, while a nifty contents pane shows up on your left. The only bug I picked up was that it really didn't like my big screen resolution, it seems to have been built for smaller monitors. Setting aside this little quirk, though, Indywiki is a great alternative to using the more text-based browser client.

To install Indywiki, use the `***indywiki***` package in the `universe` repositories.

wikipedia2text

http://url.fullcirclemagazine.org/c9e5e4



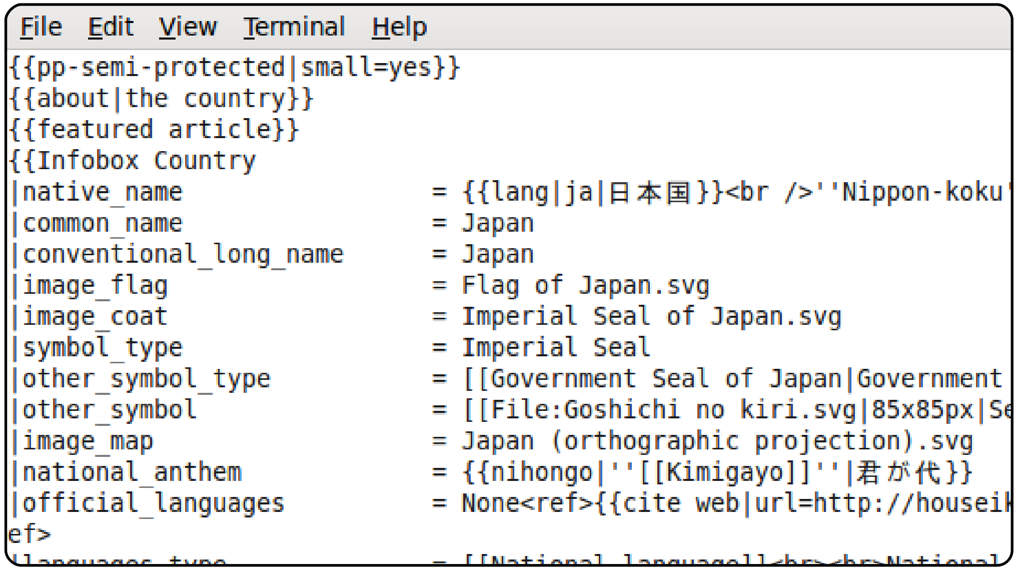
```
   language(s)

  Update method    APT (front-ends available)

 Package manager   dpkg (front-ends like Synaptic available)

    Supported      IA-32, x86-64, lpia, SPARC, PowerPC, ARM, IA-64
    platforms

   Kernel type     Monolithic (Linux)

    Userland       GNU

  Default user     GNOME
    interface

     License       Mainly the GNU GPL / plus proprietary binary blobs^
```

wikipedia2text is from the other end of the spectrum. Like Indywiki, it, too, searches Wikipedia, but it uses the most barebones method possible: the terminal. But don't be fooled - despite its command-line interface, wikipedia2text is certainly not devoid of features. It supports random pages, colorization, and summaries, as well as opening articles in external browsers. It also works well as a plugin; simply pipe in the arguments from your main program, and wikipedia2text displays the appropriate article.

To install wikipedia2text, use the `***wikipedia2text***` package in the `universe` repositories.

wikipediafs

http://wikipediafs.sourceforge.net/



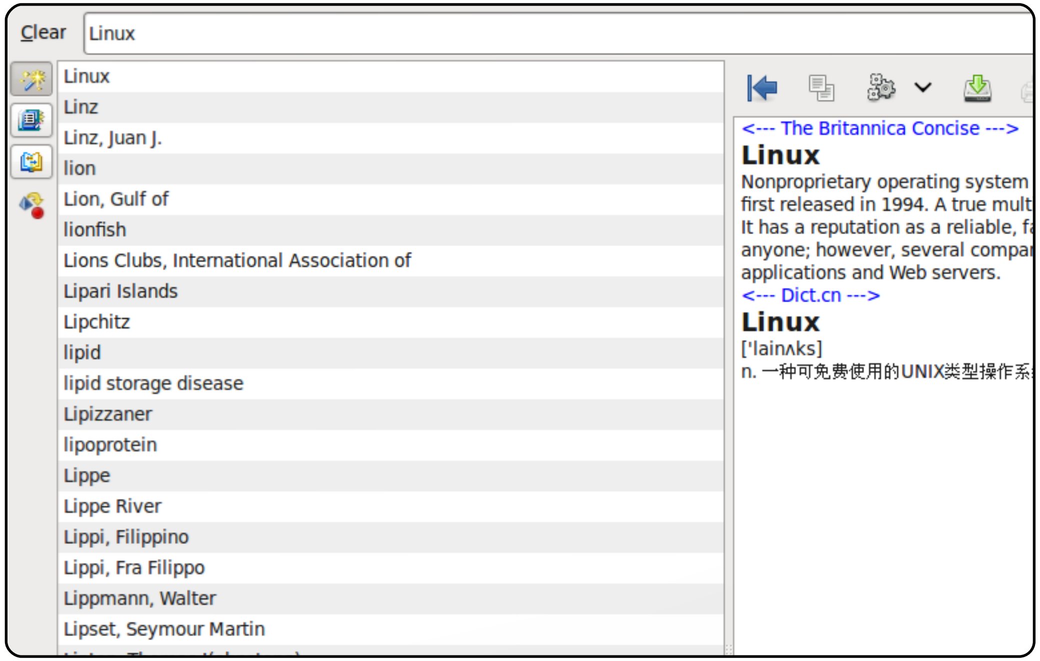
```
{{pp-semi-protected|small=yes}}
{{about|the country}}
{{featured article}}
{{Infobox Country
|native_name                 = {{lang|ja|日本国}}<br />''Nippon-koku'
|common_name                 = Japan
|conventional_long_name      = Japan
|image_flag                  = Flag of Japan.svg
|image_coat                  = Imperial Seal of Japan.svg
|symbol_type                 = Imperial Seal
|other_symbol_type           = [[Government Seal of Japan|Government
|other_symbol                = [[File:Goshichi no kiri.svg|85x85px|Se
|image_map                   = Japan (orthographic projection).svg
|national_anthem             = {{nihongo|''[[Kimigayo]]''|君が代}}
|official_languages          = None<ref>{{cite web|url=http://houseik
ef>
|languages_type              = [[National language]]<br><br>National
```

One of the coolest projects I've found is wikipediafs. Basically, this Python-based program allows you to access Wikipedia articles as if they were text files through fuse mount points. That means you can edit them through Gedit or OpenOffice.org, copy them with Nautilus, or simply view them with cat. This allows for a lot more flexibility than just viewing and editing with a browser, and has a lot of potential for developers. In short, it's like wikipedia2text on übersteroids.

To install wikipediafs, use the `**wikipediafs**` package in the `universe` repositories. Make sure you also check out the manpage (man mount.wikipediafs) for installation instructions.

StarDict

http://stardict.sourceforge.net/



Linux
Linz
Linz, Juan J.
lion
Lion, Gulf of
lionfish
Lions Clubs, International Association of
Lipari Islands
Lipchitz
lipid
lipid storage disease
Lipizzaner
lipoprotein
Lippe
Lippe River
Lippi, Filippino
Lippi, Fra Filippo
Lippmann, Walter
Lipset, Seymour Martin

<--- The Britannica Concise --->
Linux
Nonproprietary operating system
first released in 1994. A true mult
It has a reputation as a reliable, fa
anyone; however, several compar
applications and Web servers.
<--- Dict.cn --->
Linux
['lainʌks]
n. 一种可免费使用的UNIX类型操作系

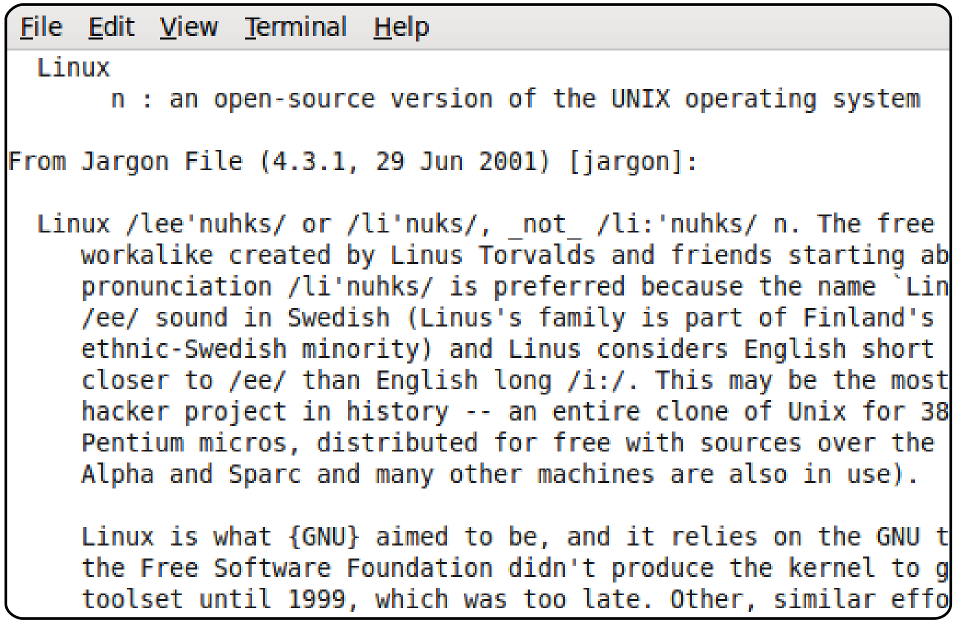
If browsing dictionaries is your thing, check out StarDict. It's a GTK-based dictionary viewer that allows you to view a ton of dictionaries, including English dictionaries like Oxford and Merriam-Webster, law dictionaries like Bouvier, and Bible dictionaries like Easton. It also packs in a lot of features, including advanced searching (fuzzy, wildcards, and so on), word translation, and text-to-speech.

To install StarDict, use the `***stardict***` package from the `universe` repositories. You can also get more dictionaries from the main website.

contents ^

dict

https://sourceforge.net/projects/dict/



```
  Linux
       n : an open-source version of the UNIX operating system

From Jargon File (4.3.1, 29 Jun 2001) [jargon]:

  Linux /lee'nuhks/ or /li'nuks/, _not_ /li:'nuhks/ n. The free
     workalike created by Linus Torvalds and friends starting ab
     pronunciation /li'nuhks/ is preferred because the name `Lin
     /ee/ sound in Swedish (Linus's family is part of Finland's
     ethnic-Swedish minority) and Linus considers English short
     closer to /ee/ than English long /i:/. This may be the most
     hacker project in history -- an entire clone of Unix for 38
     Pentium micros, distributed for free with sources over the
     Alpha and Sparc and many other machines are also in use).

     Linux is what {GNU} aimed to be, and it relies on the GNU t
     the Free Software Foundation didn't produce the kernel to g
     toolset until 1999, which was too late. Other, similar effo
```

Sometimes, simplicty trumps all. And that's where dict, a terminal-based client, comes in. It's fully compatible with StarDict (as a matter of fact, StarDict is based on dict), but without a GUI. You don't get all the bells and whistles, but you get a clean, powerful tool that supports a ton of services with unmatched speed. Sometimes, there's no school like old school.

To install dict, use the `***dict***` package in the `universe` repositories. You can also apt-cache search dict for looking up assorted dictionaries available for installation.

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

Full Circle Team

Editor - Ronnie Tucker
ronnie@fullcirclemagazine.org
Webmaster - Rob Kerfia
admin@fullcirclemagazine.org
Comms Mgr - Robert Clipsham
mrmonday@fullcirclemagazine.org
Podcast - Robin Catling
podcast@fullcirclemagazine.org

Editing & Proofreading
Jim Barlow
David Haas
Gord Campbell
Brian Jenkins
Mike Kennedy

And our thanks go out to Canonical, the Ubuntu Marketing Team and the many translation teams around the world.

Deadline for Issue #35:
Sunday 07th March 2010.

Release date for issue #35:
Friday 26th March 2010.

contents ^