The NEW PCLinuxOS Magazine

Volume 34

Happy Thanksgiving!

November, 2009
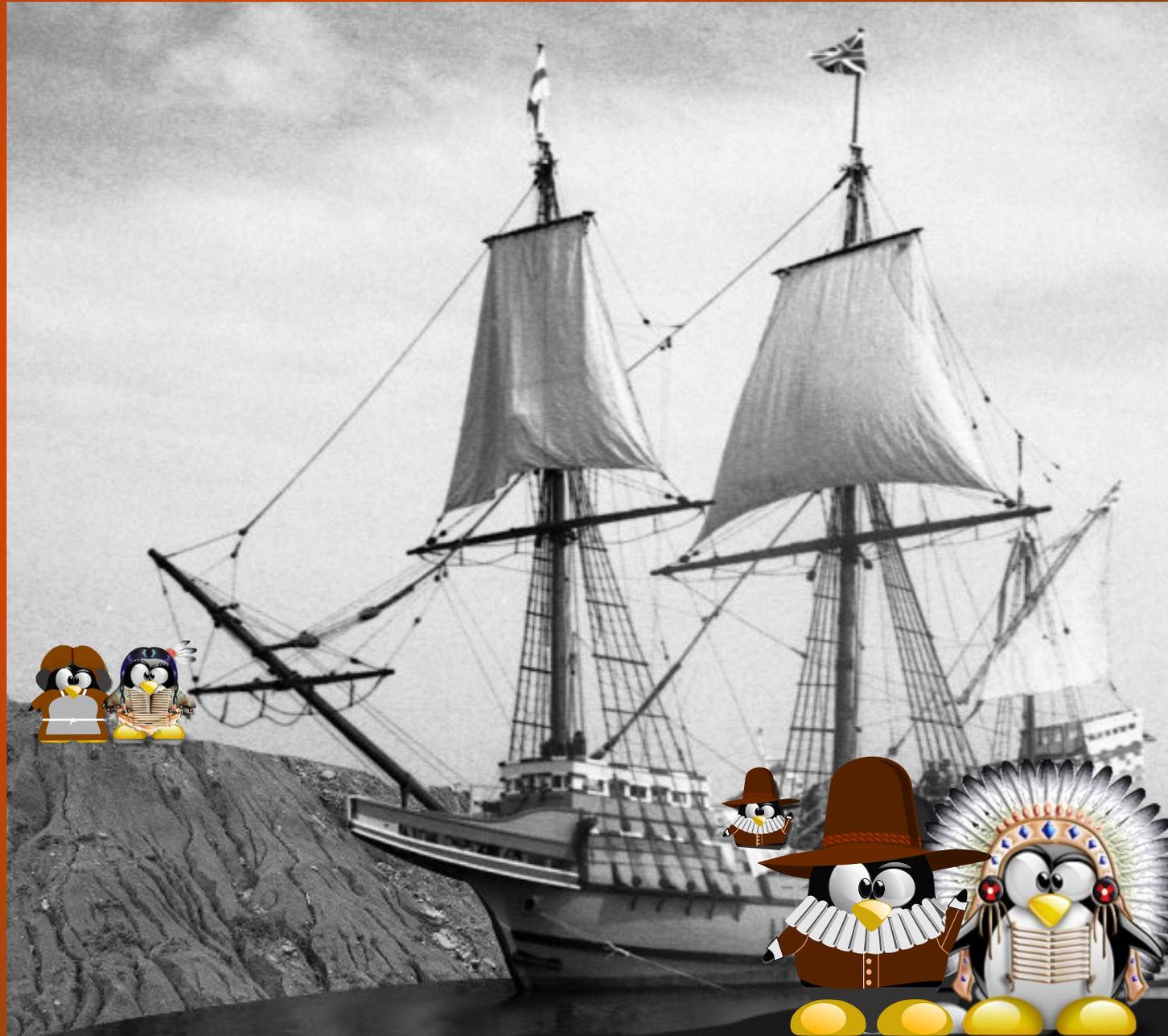
Table Of Contents

Welcome From The Chief Editor

Welcome to another exciting issue of **The NEW PCLinuxOS Magazine**! As is usually the case, there is a lot of exciting things going on with PCLinuxOS.

First, Texstar and the Packaging Crew are hard at work. Since updating the toolchain, new versions of all sorts of applications have been hitting the repository at the speed of lightning. Texstar has been putting his usual "midas touch" on things, and is working steadily towards a stable KDE 4.3.2. Work is progressing on the PCLinuxOS 2009.3, the third quarterly ISO release. Sproggy has created a test release with e17. And all of this is leading up to the PCLinuxOS 2010 release.

You are also likely to notice a few new things with the magazine this month, as well. The first thing you may notice is the new look for the magazine layout. We hope you like the new layout. We also hope that it helps keep the magazine's PDF file size down, by making better use of the available space on each page, without compromising on the content we bring you each month. As an added benefit, those of you who choose to print out various magazine articles for use as reference materials should experience less ink and toner usage. We carefully tried to mirror the colors used in the PCLinuxOS forum and those colors used by the popular Blue Caress theme that is the default theme of the 2009.2 quarterly ISO.

The second thing you may notice is the new section in the magazine: **Screenshot Showcase**. For as long as I've been with PCLinuxOS, folks have been posting their screenshots in the PCLinuxOS forum. Each month, we will pick out several of those to highlight in the magazine. You guys and gals post some very nice screenshots every month, and this is another way to show off a sampling of those to the "outside world." The Screenshot Showcase appears throughout the magazine, along with the name of the person who posted it, the date they posted it, and the desktop environment they are using. To kick Screenshot Showcase off this month, we have six screenshots featured. It wasn't easy to pick which six to run. I had to pick them from 18 finalists. I only wish there was more space to be able to run them all.
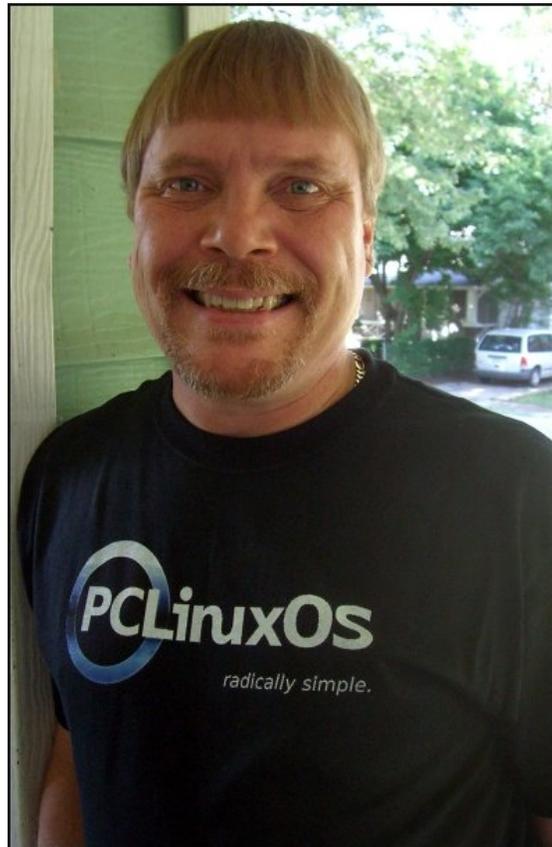
We have lots of other things for you this month, besides the new feature of the magazine. BobK54 went to **Ohio Linux Fest 2009** and reports back on his attendance there. We get to learn more about the people **Behind The Scenes**, as we get to know more about **Joble**. Meemaw continues her article on using your scanner, taking a look at using Kooka in **Scanner Saga: Part 2**. Meemaw and I also put our heads together to publish a rather long, but definitely not comprehensive, list of links where you can find **Wallpaper Sites** to download wallpapers to dress up your desktop. Ms_meme is back with another round of **Forum Foibles** and **ms_meme's nook**, as well as **Forum Puzzle**, a nice prose she penned about the PCLinuxOS forum. Georgetoon graces the magazine with another **Double Take** and **Mark's Quick Gimp Tip**. Kalwisti writes up **An Absolute Beginner's Guide To LaTeX On PCLinuxOS**, which should be enough to get anyone interested in learning LaTeX going in the right direction. Critter provides us with **Command Line Interface Intro: Part 2**, continuing his tutorial on getting users comfortable with using the command line. AndrzejL gives us this month's **Gadgets & Gear** column, where he reviews using his new wireless optical mouse with PCLinuxOS. Gary Ratliff, Sr. continues his series of articles, **Computer Languages A to Z**, where he sheds some light on **Elisp**.

With the recent release of Microsoft Windows 7, Ver Pangonilo shares a reprint from his blog on setting up a **dual boot with PCLinuxOS and Windows 7** -- after the Windows 7 upgrade messes up your GRUB boot loader. We also have a photo of **Linus Torvalds** on the Windows 7 launch day, where he was in attendance at the Japan Linux Symposium. JohnBoy shares with us an easy, step-by-step

method of **manually creating a Live USB**. I write up an article on using **mencoder**, and share my initial experiences with **Google Wave**, after receiving my invitation to participate in the Google Wave Preview. Katie gives us a **testimonial**, and **Flashback** features using **Grsync to create backups** of your personal data. This month's Wiki Wicket covers **migration from Microsoft Windows to PCLinuxOS**. And finally, this month's cover comes to us, yet again, from **Timeth**, of the PCLinuxOS Beautification Team.

November, in the United States, is the month when we Americans celebrate our **Thanksgiving** holiday, commemorating the first fall harvest of the 17th Century settlers on Cape Cod, in Massachusetts. Regardless of your country of origin or residence, I believe we can all find things in our lives that we are thankful for. For me, I'm thankful for my wife, my family, my job, good health, great friends, and all the things that make my day-to-day life as wonderful as it is. I'm also very thankful for Texstar and the

Packaging Crew for creating PCLinuxOS, a stable OS where my computer works for me, rather than me working for my computer. I am thankful to be able to contribute back to PCLinuxOS as this magazine's editor. So as we enter November, I'd like to urge each and every one of you look around you and acknowledge the things in your life that you are thankful for -- and to continue to acknowledge those things throughout all the year.

Want To Help?

Would you like to help with the PCLinuxOS Magazine? Opportunities abound. So get involved!

You can write articles, help edit articles, serve as a "technical advisor" to insure articles are correct, create artwork, or help with the magazine's layout.

Join us on our **Google Group mailing list.**

Reach Us On The Web

PCLinuxOS Magazine Mailing List:
http://groups.google.com/group/pclinuxos-magazine

PCLinuxOS Magazine Web Site:
http://pclosmag.com/

PCLinuxOS Magazine Forums:

PCLinuxOS Magazine Forum:
http://pclosmag.com/forum/index.php
Main PCLinuxOS Forum:
http://www.pclinuxos.com/forum/index.php?board=34.0
MyPCLinuxOS Forum:
http://mypclinuxos.com/forum/index.php?board=157.0

Ohio Linux Fest: 40 Years Of Unix

by Bob Krausen (BobK54)

The seventh annual Ohio Linux Fest was held at the Ohio Convention Center in Columbus OH on September 25th, 26th, and 27th. The theme for this year was "40 Years of Unix".

This was my second Ohio Linux Fest and I proudly wore my new PCLinuxOS t-shirt fresh from Cafepress.   OVER 1,100 people attended OLF2009 despite the economy and weak preregistration. A TOTAL success story! My camera died on me at OLF, so I've scavenged a few pictures from other sources and present them with attribution.

OLF2009 didn't disappoint. I attended only the Saturday open session. The Friday sessions had two very different main focus groups: the first was a tutorial class for professionals who wanted to take the LPI Certification exam for Linux System Administrators; the second was a Linux Basics Class sponsored by FreeGeek Columbus where participants not only learned their way around an Ubuntu desktop, they actually assembled their own computer from donated parts and got to take it home.   The Linux community gained 19 converts that Friday, and 19 people gained an understanding of how a computer works and how freedom really feels. Friday always holds the pre-party which I, again, had to bypass. The Sunday session seemed interesting, but I couldn't attend. It was a workshop entitled "End Elitism: Diversity in Open Source."

Saturday morning started off with a lively opening keynote on "Saving the Economy with Linux," by Shawn Powers, an Editor for Linux Journal. Shawn's "day job" is as Technology Director at a school district in northern Michigan. He told us we need to focus not only on adult converts to Linux, but the kids as well. Teach them the value of freedom. He told us to be honest while evangelizing about Linux. Don't bash MS, sell Linux's advantages. Be upfront with issues. From his point of view, he can see some disadvantage in all the desktop environments in Linux, while some others may see that as freedom of choice. He also faults the variety of package managers out there.

The rest of Saturday had 28 classes spread over eight sessions to cover a myriad of interests, from user to hardcore developer. I learned some of the Linux boot process as presented by Dann Washko of TLLTS fame; I learned some things about the new Gnome 3.0 coming out, about Moblin 2.0, about the legalities of FOSS, about building a community, about Drupal, and about KDE4. The list of classes was amazing. I wish I could have attended more.

Since the theme of OLF2009 was "40 Years of Unix," two sessions were especially interesting to me. Peter Salus, author of "The Daemon, the Gnu, and the Penguin," among many other books and generally regarded as the historian of Unix, gave a great presentation. His presentation was a montage of issues and events from the year 1969.   He pointed out the political, the ecological, and the technological events of the year. Most poignant of all the slides were one depicting the birth of Unix and another depicting the birth of Linus Torvalds. Those two slides brought together the message of why 1,100 of us might gather in Ohio in 2009 to celebrate something as mundane as a computer operating system.

The second session of interest was the closing keynote given by Douglas McIlroy. Mr. McIlroy's presentation was called "A Surfeit of Sophistication," and was a rather scathing critique on where Unix and Linux commands are today versus at the birth of Unix. Mr. McIlroy was in charge of part of Bell Labs in 1969. One of his direct reports needed to do some kind of computer calculations and he went home and
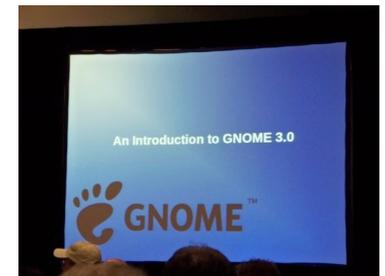
wrote the first stanzas of what we know today as Unix. Mr. McIlroy developed Unix pipes and Unix tools such as spell, diff, sort, join, graph, speak, and tr.

To illustrate his loathing of the current state of Unix/Linux tools, he noted the command line entry "less," which, if you consult the man pages via "less –help" gives you 220 lines of options for that command. His comment was "Why? How can 'less' be more than 220 lines of options long?". He also noted the command "ssh" which has 38 options and 64 configurations. He admitted there was probably a reason for each of those permutations but he bets many are unused dross in the system today. His point was he was concerned the operating system he and his cohorts gave birth to 40 years ago is not as svelte today as it was envisioned at that time. I think he was saying we have gotten sloppy over time because simply because we could get sloppy. In

1969 they HAD to be svelte, since they had minimal amounts of memory to play with.

The Linux elite are, thankfully, mere "fellow attendees" at OLF. Everyone is very much available, and will take the time to talk to you. Remember, I'm NOT a developer, just a end user! Last year I stood next to and chatted with Jono Bacon of Ubuntu community fame. This year I briefly met Bdale Garbee to talk about his lost "model" rocket he described in one of last years sessions. Bdale, by the way, is the Chief Technologist for Open Source and Linux at a little company called HP.

At lunch I ran into Steve Lake, author of Raidens Realm (raiden.net). Steve stopped me because of my PCLinuxOS shirt! He spoke very highly of our community and said he still highly recommends our distro to anyone wanting a painless Linux experience. The TLLTS guys were everywhere and did their usual podcast from their OLF booth. There were 37 booths to visit including IBM, Oracle, Digium (asterisk), Novell, Zenoss, TLLTS, Linux in a Ham Shack, Ubuntu LoCo, FreeGeek Columbus, EFF, FSF, Redhat, Peak 10, LinuxProMag, ZaReason, Ontario Linux Fest, Fedora, KDE, Drupal and tech hip-hop entertainment Dual Core. There was not as much schwag this year, and the economy played a roll in that I'm sure.

Ohio Linux Fest 2009 was a great success from my point of view and I would highly recommend attending a Linux festival near you, if you get the chance.

Photo of the main meeting area, courtesy of Brad McMahon.

Photo of the Gnome 3.0 presentation, courtesy of Steve Lake, Raiden's Realm.

Photo of seminar session, courtesy of Steve Lake, Raiden's Realm (raiden.net).

Visit Us On IRC

• Launch your favorite IRC Chat Client software (xchat, pidgin, kopete, etc.)

• Go to freenode.net

• Type "/join #pclinuxos-mag" (without the quotes)

The magazine's series of articles profiling the folks behind the scenes, doing the work that helps make PCLinuxOS what it is, and who help keep things running smoothly, continues. This month, we get to learn more about Joble, in his own words. Joble is a moderator on the main PCLinuxOS Forum, and is one of the administrators on the PCLinuxOS Magazine's Forum. Often known for his wit and for being sometimes wacky, there is a serious side to Joble – a serious side that comes out in his autobiographical "Behind The Scenes" article.

— Paul Arnote, PCLinuxOS Magazine Chief Editor

Joble: In His Own Words

I'm not on the magazine staff, but I like to read anything and everything related to my distro of choice, PCLinuxOS. So I was very happy to be invited to help administer the new magazine forum.

I am a retired USAF Security Policeman. My first experience with computers was when our electric typewriter was replace by a Z100 with no hard drive, and WordPerfect came on five floppy disks. I got pretty good at writing creative batch files and doing some basic programming in basic, to do the things I couldn't already do on the computer. I think they call those little files "Utilities".

I took some classes in computer science, learned a little Pascal and C++, and soon became known as the "Computer Guru". If you need it, and it doesn't already exist, I can write a "Utility" for you, that will get it done. For a while I was almost indispensable. Those days didn't last long. I was a cop, after all.

Some years later, long after I was the "Guru", Vista came out. Here I sit with a laptop and WinXP. I read that Windows will soon stop supporting security updates for XP, which I have known for a long time are essential to safe internet browsing. My brother was using Red Hat at the time, and he could host his own web page, run his own mail server and compile source code. Well, I know it's been a long time since Basic stopped shipping with Windows. I was familiar with Open Source Software, however. I was using Firefox, OpenOffice and a free version of Apache on my XP box. The move to Linux came naturally, but finding the right distro was not so easy.

I first found a version of Puppy Linux that would run from a USB stick in Windows. I liked what Linux had to offer.

I did my research and downloaded several distros from Distrowatch and burned them to CD. I tried Scientific Linux, based on Red Hat, cause I knew I could call my brother and get help. I also tried Ubuntu, Puppy, and a few others. Now some limitations came into play, and these limitations are the reason I use PCLinuxOS today.

After searching extensively how to get online with wireless (the only way I could get online), I soon discovered I needed something called ndiswrapper, and none of the distros I had tried had it. I read all about how to use it, how to find my windows drivers, how to install it, but of course I had to download it first and I had no way to do that, until I read one day about this distro called PCLinuxOS and saw what was installed by default on the Live CD, NDISWRAPPER! Having a Live CD that would go on-line, naturally this was the first distro that ever got installed on my system, which at the time it was dual-boot. I was such a chicken, but I was learning

Linux, and I began to learn as much as I could about ndiswrapper and wireless, in order to help others in the forum.

My next big challenge came when there was a serious need for packagers. I vowed to learn it and help the distro in any way I could. I'm still not a very good packager, but I did manage to get a few things into the repositories, and I'm still learning. I have made a lot of friends in the process and I love testing stuff out for them. I really enjoy testing pretty much anything, and take great pride in "breaking my install" and fixing it again, without reinstalling from a Live CD. I have done this now, many times.

My goal for the future is to learn kernel configuration, compiling and packaging, as I believe there is a need for that. Linux is to big to master, just as I specialized in wireless and continue to help where I can, I think my next step is to find something more advanced to "specialize" in.

Do one thing and do it well.

Joble

Screenshot Showcase

Johnr_1, Posted October 12, 2009. KDE 4.3.2

Katie's Testimonial

I have been an enthusiastic user of PCLinuxOS for sometime now, but each day as I peruse these forums I feel like I need to say, "Thank you" again.

My adventures in Linux began several years ago with the book, "Kiss The Blue Screen of Death Goodbye" and its included CD.

An acquaintance of mine then suggested I try SUSE 9.0, which I loved because of the way it updated. However, I have always been an advocate of using an OS you have a good local support group for -- and my friend moved away. The forum people were great, but it's not the same thing.

Next came MEPIS. A users group in a nearby city was pushing it through their installfest. It worked fine, until one time I made the mistake of accepting an update that wasn't able to download all the files needed. "OK" was far from OK. And the users group was really far above me and didn't seem to have time for beginners (for whom they had done the installs).

Then I heard about PCLinuxOS 2007-- WOW! I shadowed the forums for several months trying to learn all I could (& find out if they were beginner friendly). I ordered a CD from a provider and it installed easily & was beautiful-- and "just worked."

I went through the "big update" -- no big deal (except I tried to reboot before it was finished on my first machine--- so I re-installed and tried again -- perfect).

I play with older machines, virtual machines, and some newer ones -- and I am having a ball. Am I M$ free? Not quite, but mainly because I do so much OCR work and desktop publishing. I know there are Linux programs for those. And I am hoping that someday they will work for me. But they aren't quite what I need right this minute. However, I love the ability to keep a P4 machine online 24/7/365, and know it won't be infected or compromised. (I keep M$ as far away from the Internet as possible.)

Two observations, if I may (and this is really why I am writing):

1. I am constantly amazed at how friendly everyone is around here. Thank you from the bottom of my heart. This place is amazing!!! And every time I check in here I find that I have missed anywhere from 4 to 10 or 12 screens of new posts!! Amazing! Some forums you wonder if anyone is reading. This place is so busy, and I learn so much every day.

2. Tex and the Gang -- Thank you. Updates -- WOW: I installed the automatic updates and I cannot believe how busy you folks are (and how dedicated you folks are). Some days only a couple updates, but most days 6-8 updates (I had to be away for a couple days recently and came back to see 30-some updates-- I only missed 2 days). Tex and the Gang -- You folks are just amazing and I just wanted to tell you how much I appreciate all that you do for the community. I'm sure there are days you wonder if anyone notices. I notice every day, and I am so thankful for all you do.

Katie
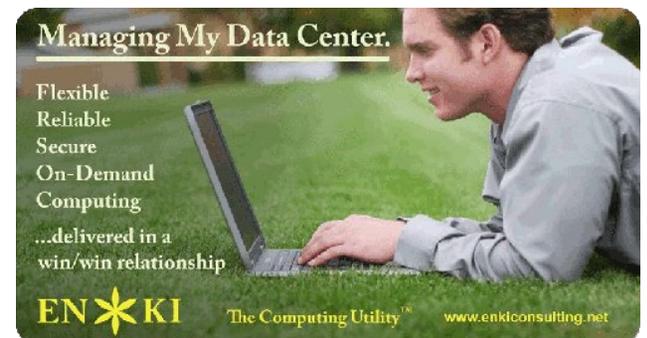
Forum Puzzle: The Pieces Make It Whole
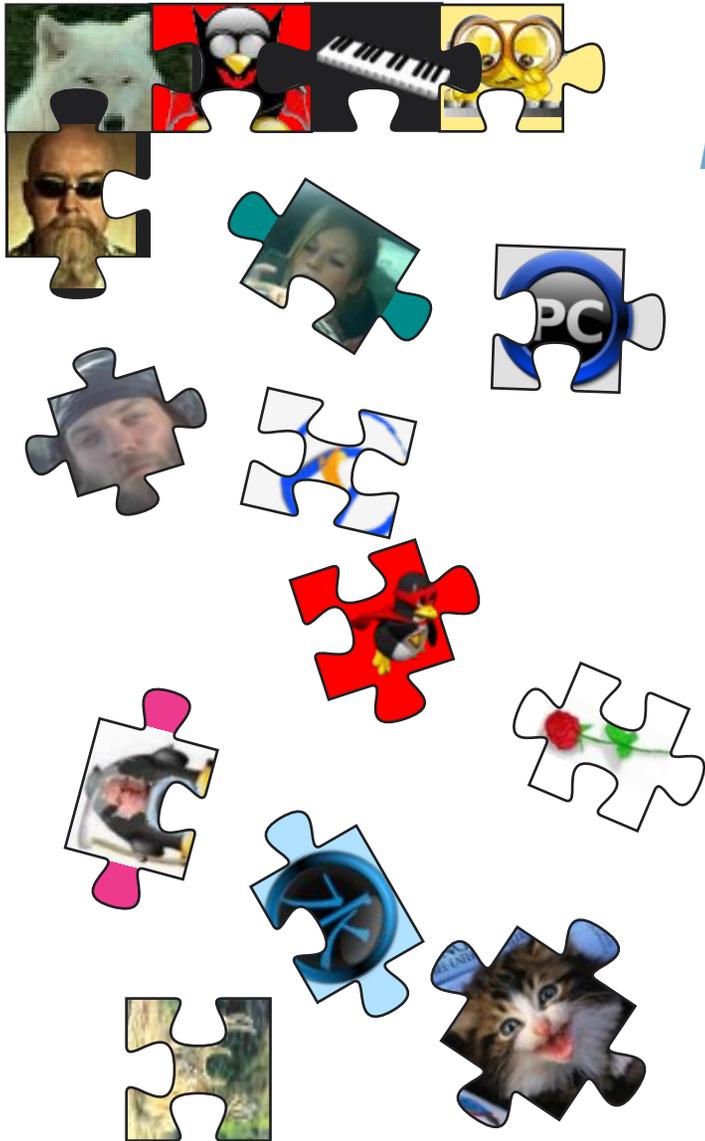
The forum is like a puzzle
Different talents of great size
All loving PCLOS
But that's really no surprise

Many come to the sandbox
To post more and more
Their jokes and their wit
Will never be a bore

Others give of their time
To fix mistakes we've made
Knowing that in dollars
They never will be paid
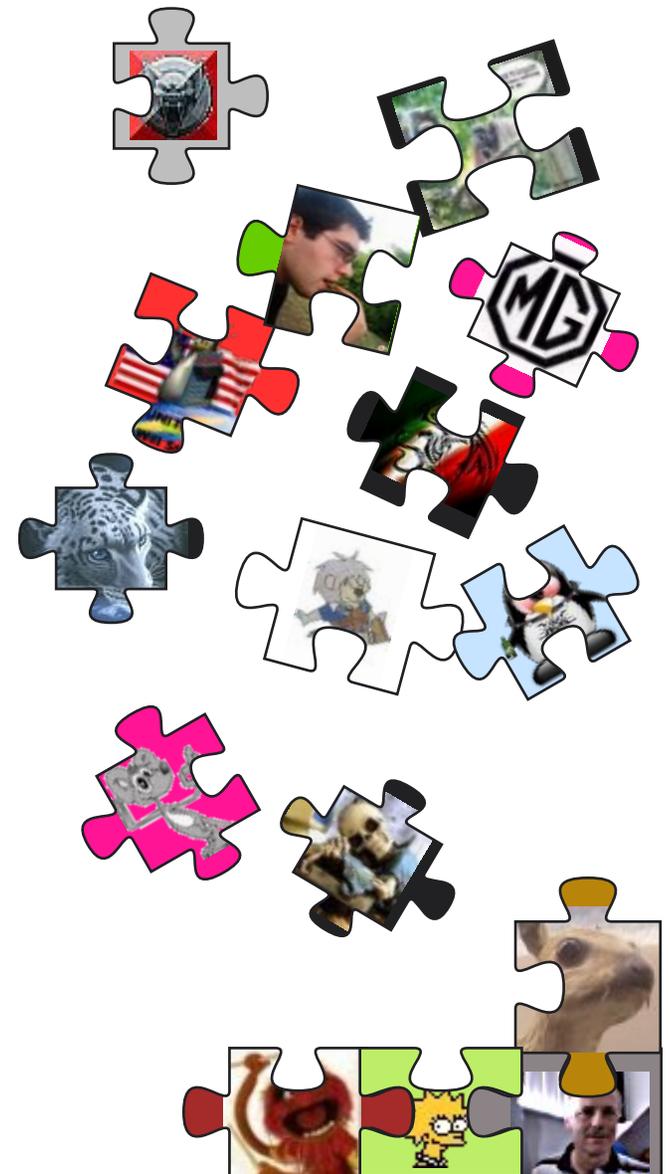
There are artists who give beauty
Everyday that we see
Just look at your desktop
And you will agree
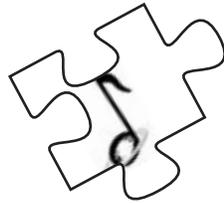
Others work behind the scenes
So everything will run smooth
Without their constant labor
We couldn't make a move
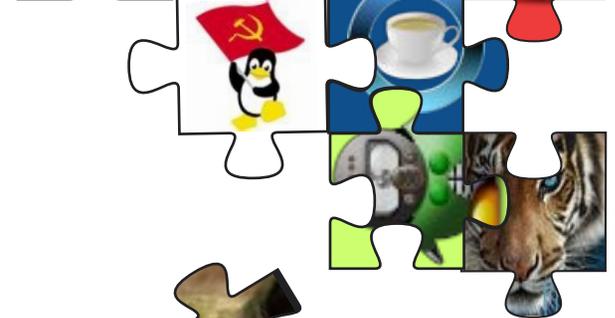
Although we are all different
You really must admit
No matter how you turn them
The pieces always fit

ms_meme
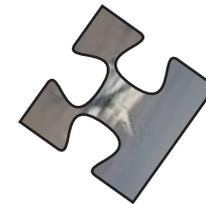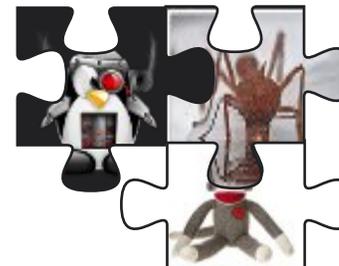
Dual Boot Windows 7 & PCLinuxOS

by Ver Pangonilo
Reprinted with permission from his blog
http://s.pangonilo.com/index.php/2009/10/linux-and-windows-7-dual-boot.html

The pre-installed Windows Vista Home in my laptop just kept irritating me. It came to a point where most shortcuts in "My Documents" do not work anymore, as it prompts me that I don't have the rights to those directories. My open source programs kept on crashing - until I decided to dual boot it with PCLinuxOS 2009.

The choice of PCLinuxOS 2009 was easy, since it was the only LiveCD distro (and Mandriva Live) that enabled my wireless connectivity out-of-the-box. It was unfortunate that Ubuntu, gOS, Fedora Core and Zenwalk failed in that respect.

The installation of PCLinuxOS was a breeze, with no glitch at all. The bootloader was able to detect the current Windows. I was able to dual boot Linux with Windows Vista. Now came Windows 7. I decided to upgrade my pre-installed OS with this much hyped Windows. Like the PCLinuxOS 2009 installation, I did not have any problem, not until I wanted to boot my PCLinuxOS. Windows 7 had overwritten the GRUB boot loader and failed to (or deliberately did not) include non-Windows operating systems.

Why do I still need Windows when, in fact, I have my Linux? Well, I still use some Windows only programs. I only use it for these programs, and nothing else.

Back to my problem: how can I reactivate my Linux. I don't want to re-install it as I did so much custom configuration and I don't want to lose them. I tried searching the net about GRUB re-installation. The simplest is **Doctor's Blog**, at http://saleem-khan.blogspot.com/2008/04/how-to-reinstall-grub-on-pclinuxos.html.

To summarize the steps in re-installing GRUB:

1. Boot with the PCLinuxOS 2009 LiveCD as root

2. Open terminal as a super user

3. Type "grub" then enter. The grub» prompt will appear.

4. Type grub»find /boot/grub/stage1 then enter. A list similar to this will appear.

(hd0,0)
(hd0,4)
(hd0,5)

5. You need to know which partition your Linux OS is located. In my case it is (hd0,4), thus I typed the command
root (hd0,4).

6. Next, enter setup setup (hd0) at the grub» prompt.

7. Exit GRUB by typing "quit"

8. Restart you computer. The GRUB bootloader should now appear.

In my case, the problem did not stop there. If I select Windows 7, it just fails to start. So, at this instance, my Linux works but my Windows not.

I remembered that during the installation of Windows 7, there was a menu about repair so I tried clicking it, then selected restore from a previous restore point. After this process, I restarted my laptop and was able to boot either PCLinuxOS or Windows 7 without any glitch at all.

Screenshot Showcase

Posted by Archie, October 16, 2009, KDE 4.3.2

PCLinuxOS Enlightenment e17 ISO

Coming soon!

Catch The (Google) Wave

by Paul Arnote (parnote)

A new wave is rolling towards your shore. And it's likely to completely change your whole concept of communication via the Internet. It's called Google Wave.

I recently received an invitation to participate in the Google Wave Preview program from my father-in-law, who works for a fairly good sized Internet provider in central Missouri. He and I have had conversations about Google Wave before, when we've gone to visit my wife's hometown, but this is the first time I've actually had the chance to see, beyond mere concepts, what all the buzz is about with Google Wave.

Google Wave's premise is simple, yet ambitious. Email, as we know it and use it today, was created over 40 years ago, before there were social networking sites, blogs, wikis, instant messaging, and many other communication tools we now tend to take for granted. It is one of the most popular and most used of all Internet applications. The Google Wave team decided to step back and ask, "What if email were created today?" The result they came up with is what we see with Google Wave.

Google Wave was rolled out in a public reveal at the Google I/O Developer Conference back in May, 2009. In fact, you can watch the entire announcement, a one hour and 20 minute video, online. Google Wave takes advantage of the proposed HTML 5.0 standard, and attempts to show what is possible within the web browser with the HTML 5.0 environment enabled. The early public reveal by Google was so that they could get developers onboard to create tools and components for Google Wave, using the Google Wave APIs (Application Programming Interfaces). In this way, it is hoped that there will be even richer content for Google Wave when it is rolled out later this year.

Google Wave development was started in 2007, and is written using the Google Web Toolkit. The Google Web Toolkit allows you to write HTML 5.0 applications in Java. Initially codenamed "Walkabout," the Google Wave project started out with a tough set of questions. From Google Software Engineering Manager Lars Rasmussen's blog, those questions were:

> **Email as we know it and use it today, was created over 40 years ago. The Google Wave team asked, "What if email were created today?"**

1) Why do we have to live with divides between different types of communication -- email versus chat, or conversations versus documents?

2) Could a single communications model span all or most of the systems in use on the web today, in one smooth continuum? How simple could we make it?

3) What if we tried designing a communications system that took advantage of computers' current abilities, rather than imitating non-electronic forms?

Engineered by Lars and Jens Rasmussen, the creators of Google Maps, and headed up by Lead Product Manager Stephanie Hannon, Google Wave is much more than just email. It's a robust, open source system that allows realtime conversations with any, or all, of your contacts at the same time. It combines what we traditionally know as email with instant messaging and chats, and allows the user to include rich text content, maps, pictures, videos, and so much more. In fact, Lars Rasmussen calls it a "communication and collaboration tool."

Traditional email follows, more or less, the "snail-mail" model. You type in a message to a user (or group of users), hit send, and wait for a reply. If someone joins the conversation later on, the earlier messages in the "thread" aren't typically available to the user who has joined in, later in the conversation.

Google Wave changes that model, significantly. While you are participating in a wave, you can see

According to Hannon in the roll out video, because it is a hosted conversation, it's easier to keep track of structure. With traditional email, you can hit reply and manually hand edit the quoted text to insert your comments. With Google Wave, however, you can tell the server to simply "split" the original message and insert your comments at the appropriate spot. Overall, not too terribly different from traditional email, but (at least in the demonstration at the developer's conference), easier to do and to follow, since it ends up structured much more like a real conversation might be.

When you reply to a wave, your message appears instantly on the screen of the other members of that wave, if they are logged in and viewing, as you type it. There is a feature, still under development and not yet implemented, that allows you to type in your message without it being displayed on the other participants screens until you click on the "Send" button. This is good for those who may either be slow typists, or those who are not very good at spelling. And, speaking of spelling, Google Wave sports an on-the-fly spell checker, called "Spelly," that will correct many common spelling errors automatically, based on the context of what you are typing.

If a user is not logged in at that particular moment, the updated wave will be highlighted in bold type when the user returns to Google Wave, indicating to them that the wave has been updated. They can then re-open the wave, and continue the conversation. I know what you're thinking. "Nothing too revolutionary here." But wait, because it gets better.

what the other invited members of the wave are saying, at the same time as it is typed, in real time. You can comment or reply in the middle of a wave. And, to top it off, you can invite other users to participate in the wave, and the entire wave is then open to them, and they can add to the conversation, even parts of the conversation that occurred before they were invited or added.

Wave differs from traditional email in that you start out with the definition of a conversation, and users participating in that conversation. Those users, or participants in the conversation, can jump in at any point. The entire conversation is hosted on a server, and the users share equally in that conversation.

Adding users to a wave is as easy as drag-and-drop. Drag a user from your contact list to the user list in the wave, and that new user is added to that wave instantly, and they have full access to the entire conversation, as if they had been there all along. They can then contribute to the conversation as if they had been there since the beginning, including all parts of the conversation from before they were added.

A feature of Google Wave, called Playback, allows new users (those invited to a wave after it has already begun) to see how things progressed to their current state. By hitting "Playback," the new user in the wave will be able to see, message by message, reply by reply, how things have gotten to their current state. They can see where the wave has been split and "sub-conversatons" have been added, in the order in which they occurred. The new user can add their reply to any part of the previous conversation, and the wave will be marked as updated in the other users' Google Wave screen by being displayed in bold face type.

There is even a feature of Google Wave that allows you to create what they term a "private reply." With a private reply, not everyone in the wave can see what you type. Instead, only those you designate or choose can see your reply. Say you, your mom, and your sister were chatting about plans for Christmas dinner. You and your sister want to talk about gift ideas for mom. You and your sister can discuss gift ideas for mom privately, between only the two of you, without that part of the conversation appearing on mom's screen. So mom has no idea of the private conversation you and your sister are having, during the discussion about Christmas dinner.

Google Wave supports drag-and-drop from your desktop. Say you want to share photos with other users in the conversation. All you have to do is open the folder containing the photos you want to share, and drag them into the open wave window. The thumbnails of those images appear on the other users screens before you have fully finished uploading the image files. Currently, the drag-and-drop feature is supported by Google Gears, but Google has applied to the proposed HTML 5.0 standards committee to make this a supported feature of the new HTML standard.

The wave window will then show a button at the bottom of the screen, allowing you to download the photos of the current wave. When you select it, the button applies the action to all of the photos in the wave, allowing you to download all the embedded photos. This makes it easy to create a collaborative photo album, with photos from all participants of the wave. Once you (collectively) have shared all of your photos with one another, you can extract the photos from that wave and insert them into another wave, that you can share with yet others, without the need to share all of the conversation that took place while you were creating the collaborative photo album.

Yet another nice and innovative feature of Google Wave allows for you to embed the wave on your blog or website. As the wave conversation is updated, so is the blog or web site entry. Visitors to your blog can comment, and their comments are added to the wave. Once the wave is published to a blog, all participants in the wave are notified by a message bar at the top of the wave of it being published. This is a nice feature, since you may not want particular comments or feelings about a subject to become publicly known.

Similarly, you can reply to comments on the embedded wave right from Google Wave, without having to go to the blog site to post your reply. The conversation can continue from Google Wave, and the blog is instantly updated with the new material in real time.

Just as you can share photos and create a collaborative photo album, you can do the same thing with documents. You can embed a document into a wave, and those you invite to the wave can

edit, contribute, and discuss the document as a whole, or in part. Comments can be embedded into the document, allowing you to discuss individual points within that document. Google Wave provides a "widget" which allows you to collapse all comments, so you can read the document without it being broken up by comments on the screen.

You can also use the playback function of Google Wave to go to any previous revision of the document, and it will also show who, from among the wave participants, made specific changes to that document. This will allow you to copy a document at any point through its collaborative changes, and save that document at that stage.

What's even more amazing, and what Lars Rasmussen termed as one of the most difficult things that the team had to solve, is that multiple users can edit the same wave at the same time. In the roll out at the developer's preview, four users were shown editing the same wave at the exact same time. And all changes were instantly reflected on all the other users' screens. And even more astounding, those edits can occur, simultaneously, in multiple languages. Using Google Translate, those edits are instantly translated to the end user's native language. In the demo, simultaneous edits were occurring in English, Chinese, and Hebrew.

Once you get into using Google Wave, you will want to organize your waves in some fashion. Similar to email, you can organize your waves into folders. Or, you can organize your waves with tags, and those tags apply to everyone who has access to those particular waves. So even if you didn't start the wave and didn't have the opportunity to give it a clever

title, you can give it a clever tag, which will be applied to the wave for all users who have access to that wave.

Even more clever, is that you can use waves to organize waves, using what the team calls "wave links." All you do is create a wave, and within it, embed links to other waves. Clicking on a wave link takes you directly to that wave. When you are finished, click the Back button, and you are taken back to the first wave that contains the wave link. And, the wave links are inserted just like pictures are -- drag-and-drop the wave from your search panel into the new wave. It couldn't be any easier.

Speaking of the search panel, it behaves in a dynamic fashion, much as all of Google Wave does. Say you are searching for waves over a particular subject matter. While you are choosing which waves to include and create wave links from, a new wave comes in about the same subject. Your search panel is updated instantly to include the new wave that just came in. All without the need for you to refresh your browser window.

Remember "Spelly," the spell checker we talked about earlier? Well, Google Wave also sports two other related cousins. First, is "Linky." Linky automatically recognizes links as you type them, and highlights them as links for you, without any additional intervention from you at all. Second, is "Searchy." Searchy allows you to launch up a pop

up window with a Google search bar at the top. You type in your search term or phrase, and a list appears below of pertinent web content. When you click on the link in Searchy, the link is automatically places in your wave. If the content is a picture, a thumbnail of the image is placed in your wave, which other users can click on to view the full size image.

Of course, Spelly, Linky, and Searchy are all possible in Google Wave via extensions. And, extensions is what the Google I/O 2009 Developer's Conference was all about. Google hopes to attract many more developers to develop extensions to Google Wave, in the hopes that users will have an even richer experience when it is finally opened up to the public.

Developers have access to external APIs of Google Wave, which contain most of the functions of the internal APIs. "Tools" created with the external APIs are called "gadgets." These gadgets can help users embed content from social networking sites in waves. They can also help create a poll of sorts, with a tool called the Yes-No-Maybe gadget. With this, say you and a group of friends are wanting to arrange a dinner at a nice restaurant that evening. So, you send out your wave to a group of friends, asking them to meet you for dinner at Hell's Kitchen at 7 p.m., and you want to know who is able to go. By using the Yes-No-Maybe gadget, your friends can reply Yes (they can go), No (they are unable to make it), or Maybe (they might be able to meet you for dinner).

And, since Google Wave has been created by the same creators of Google Maps, it only makes sense that you can also include Google Maps in your

waves. Users can do this with a gadget, created using the external APIs. Client side extensions are called gadgets, while server side extensions are called Robots.

Robots allow you to place forms in your wave, and those forms can be filled out collaboratively. You can also use the robot named "Polly" to create polls. The possibilities are astounding. "Tweety" is an extension that allows you to tweet (on Twitter) directly from Google Wave.

Just as users can email one another, regardless of who their Internet provider is, Google Wave also aims for this same functionality. They have developed an open protocol that allows users to communicate with one another, regardless of who their Internet provider is. It is called the "Federation Protocol," and can be freely used by anyone -- even those in direct competition with Google. This is in stark contrast to when Instant Messaging clients burst upon the scene a dozen or more years ago. Then, none of the competing Instant Messaging clients could talk to one another. It is the hopes of the Google Wave team to avoid this type of situation, and allow interoperability between all systems using the Google Wave Federation Protocol.

Additionally, Google Wave can be run on mobile platforms, although it's not yet implemented in the preview. When it is rolled out, Google Wave will run on the Apple iPhone and on devices running Google's Android. So, you will be able to "do the Wave" while you are on the go.

Conclusion

Google Wave is not email. Google Wave is not Instant Messaging. Google Wave is not chatting. Instead, Google Wave represents a radical shift in the Internet communication paradigm. Users will have to think of communication in a whole new fashion. And this is where the challenge comes in. Users will have to possibly "un-learn" old habits, and think of new ways to communicate via the Internet. Even the Google Wave team admits to initially being constrained by the old ways of doing things, simply because they were so accustomed to doing things in that manner. The same challenge faces you, as Google Wave rolls out to the public.

The Google Wave Preview is currently by invitation only. If you already have a GMail account, you already have completed one of the prerequisites to join in on the fun. Currently, not all features are implemented (for example, I can not yet change my profile picture; I receive a Server Error every time I attempt to change my profile picture.), but you should be able to get a pretty good feel for the direction that Google has planned for Google Wave.

Now all you need is an invitation to join in on the fun. I do not yet have the capability to invite others (all invites are reportedly used up while they determine if the Preview server can handle the load), but once I do have that ability (when Google decides to expand the Preview), I'll be more than happy to send out some invitations. Alternatively, you can go to https://services.google.com/fb/forms/wavesignup/ and sign up to be let in on the Google Wave Preview. There is already quite a lot of support

provided on the Google Wave Support pages (http://www.google.com/support/wave/).

Personally, I think Google Wave is going to be very successful. And, since Linux users appreciate rich and robust software -- and have, in fact, come to expect it -- Google Wave is definitely for you. The time is very near to ride the crest of the wave that Google Wave will create. All others are likely to be left behind, in the tidal wash.

Absolute Beginner's Guide To LaTeX on PCLinuxOS

by David Pardue (kalwisti)

1. Introduction

Would you like to create polished, professional-looking documents? Thanks to Texstar's packaging skills, we have a wonderful tool to do just that: LaTeX, which is part of the TeX Live implementation recently added to the PCLinuxOS repositories. This article has a twofold purpose: I would like to introduce LaTeX and encourage you to give it a try; I will also include a brief how-to which covers installing TeX Live on your system, a powerful yet user-friendly LaTeX editor (Kile) and a sample source file you can compile and preview.

In Part Two of the article (to be continued in a later issue), I will provide you with an annotated mini-bibliography of works on LaTeX that I found helpful when I began learning the program. It includes books and articles, some commercially published and some freely downloadable from the Web.

LaTeX suffers from a reputation of complexity which frightens some new users. Acquiring the fundamentals requires a time investment, but you do not have to delve into LaTeX's intricacies to produce straightforward documents.[1] I am proof that the learning curve is not insurmountable because I have been using LaTeX since around 2006. I am self-taught with it; my background is in the humanities and I am not a programmer, mathematician nor a Linux guru. When Dr. Leslie Lamport, the creator of LaTeX, was asked if LaTeX is hard to use, he answered:

"It's easy to use—if you're one of the 2% of the population who thinks logically and can read an instruction manual. The other 98% of the population would find it very hard or impossible to use."[2]

Although some might classify his statement as elitist, I would hope that as Linux users, we belong to that 2% of logical people who can — and will — RTFM.

Due to space limitations, I will only cover the highlights of LaTeX and some of the information will necessarily be incomplete. TeX and LaTeX have been used for a quarter-century; there is a vast body of published literature on them, so the problem I face as a presenter is not what to include but what to omit.

2. What is LaTeX?

LaTeX, usually pronounced "lay-tecks", is a computer typesetting system. Although its original purpose was to typeset mathematics, it can be used to produce a wide range of documents — everything from class handouts, PowerPoint-style slides and letters to journal articles, reports, theses and even books.

LaTeX is a large set of macros built upon the TeX computer program. It was initially developed by Lamport in 1985 and is now maintained by a group of international experts—both mathematicians and computer scientists. TeX (composed of the Greek letters tau epsilon chi [τεχ] and usually pronounced "teck") is the basic typesetting engine, written by Dr. Donald E. Knuth of Stanford University. Lamport draws the analogy that LaTeX can be thought of as "a house built with the lumber and nails provided by TeX."[3] LaTeX is the recommended system for all users except typographic programmers and computer scientists.

Knuth originally wrote TeX circa 1978 to produce Vol. 2 of his magnum opus, The Art of Computer Programming. He received the galleys from the publisher, and the quality of their mathematical typesetting was so poor that he decided he could create a better program himself. Ever the perfectionist, he set out to learn the traditional rules for typesetting math, what constituted good typography, and as much as he could about type design. He estimated this would take about 6 months; ultimately, it took nearly 10 years.[4]

In case you are curious about the unusual capitalization of "TeX" and "LaTeX," this is how their names are written in running text when it is not possible to precisely reproduce them with their logos.[5] Knuth chose "TeX" in order to distinguish it from other system names such as TEX (the Text EXecutive processor, a scripting language developed by Honeywell in 1979).[6] Lamport followed suit with "LaTeX" to avoid confusion with the word "latex."

3. LaTeX's Design Philosophy

We must keep in mind that LaTeX is not a word processor and does not behave like one. TeX and LaTeX are computer languages. We can compare them with other programming languages such as C or Java, in which source code is written with a text editor, then the code is converted to a binary

executable. However, TeX is a typesetting language, so the source code is converted into a typeset document.[7]

Word processors adopt a visual design approach (WYSIWYG), which does not separate the tasks of composition (i.e., writing) and typesetting. Allin Cottrell argues that this needlessly distracts authors by tempting them to "fiddle with fonts and margins." To illustrate the flaws in this model, he asks us to imagine Jane Austen wondering what font and size to make the chapter headings of Pride and Prejudice: a foolish situation, of course, because Austen was a novelist, not a trained typesetter.[8]

LaTeX adopts a logical design approach to document creation by focusing on structure. It allows you to concentrate on writing and lets the program do the design / layout, with TeX as its typesetter. TeX handles typesetting admirably, so you do not have to know much about typesetting in order to use it. For instance, LaTeX has a command for a first-level section heading (\section{Your Section Title Here}), but in using this command you do not need to worry about the size and style of type (e.g., 16-point Roman, bold, upright typeface) that will be used to produce the heading in your output. All this is taken care of behind the scenes by LaTeX.

Lamport firmly believes that a writer's primary focus should be on content rather than visual layout. When asked about the three most frequent LaTeX mistakes that people should stop making, he replied:

1. Worrying too much about formatting and not enough about content.

2. Worrying too much about formatting and not enough about content.

3. Worrying too much about formatting and not enough about content.[9]

LaTeX is a markup language (somewhat similar to HTML markup), not a page layout program. Users who want to control the placement of every single dot on the page will be unhappy with LaTeX. But if you trust LaTeX and TeX and let them perform their intended tasks, you will be rewarded with elegant documents. The output resembles something produced by a commercial book publisher. TeX's output will contain ligatures if necessary (with letter combinations such as "fi" and "ff") and kerning (bringing letters such as "A" and "V" closer together for easy reading) will be done automatically. TeX will introduce an appropriate amount of inter-word spacing and it will not produce paragraphs with rivers of space running through them.

Hyphenation will be done sensibly and the traditional stock of printers' punctuation marks (such as em dashes and en dashes) will appear as they should.[10] TeX measures with incredible accuracy: its unit of internal measurement is smaller than the wavelength of visible light!

For those of us accustomed to word processors, LaTeX's approach requires mentally shifting gears. But it does not have to be traumatic. The typical work cycle is discussed in Section 5.2 below.

4. Why Use LaTeX?

LaTeX has many advantages over word processors; I cannot list them all but I would like to mention the most important ones[11] :

Completely free of charge.

Proven, stable and virtually bug-free: TeX and LaTeX have been tried and tested for over 20 years.

The de facto standard in scientific publishing, especially in the fields of math, physics and engineering. Used by major publishers such as Elsevier, Springer-Verlag, John Wiley & Sons, Addison-Wesley, Oxford University Press and Cambridge University Press.
Extensively supported via the Internet and user groups (TeX Users Group [TUG], Dante e.V. [Germany], NTG [Netherlands]), etc.[12]

The premiere typesetter for math (mathematical formulas).

Produces industry-standard PostScript and Acrobat PDF files.

Tables and figures follow the standard publishers' practice of floating.

The .tex file format is portable: plain text (ASCII or Unicode). It can be created with any text editor and moved between different operating systems with no information loss.

The file format is more compact and more secure (less likely to be corrupted by viruses, etc.). Also, because it is plain text, your source files will still be readable 20 years from now.

Internationalized: LaTeX and TeX support a wide range of fonts and languages, including non-Latin languages such as Japanese, Chinese, Hindi, Thai, Vietnamese, Coptic, Greek, Arabic, Hebrew, etc.

Typefaces: Contrary to the popular misconception that "TeX only has the Computer Modern font," hundreds of typefaces are available—both free and commercial. It includes specialist fonts for technical matter, linguistics and mathematics.

Platform-independent: TeX is guaranteed to produce identical results no matter on which system it is run.

For example, you may type the first version of your source file on your office PC running Microsoft Windows; your co-author may make improvements to the same file on a Macintosh running OS X; and the journal publishing your article may use a Linux machine to prepare the manuscript for printing.[13]

There are default styles for common document types, such as article, report, letter, beamer [similar to PowerPoint].

Auto-formatting of chapters, sections, footnotes, etc.

LaTeX can automate the handling of cross-references, bibliographic citations, tables of contents, indexes and glossaries. It takes the drudgery out of many of the necessary tasks that are involved in the composition of a large or complex document.

For example, if you need to alter the structure of your nearly completed report by moving chapters or sections, then LaTeX will automatically renumber all the chapters and sections appropriately and produce an updated and correct table of contents. If your book contains cross-references to chapters, sections or pages, then all this cross-referencing information will be updated as well.

It's great fun to use!

5. How-To

5. 1. Installing the Required Software

5. 1.1. Installing TeX Live

Thanks to Texstar, we are fortunate to now have TeX Live (TL) available in the repositories as a replacement for its outdated predecessor, teTeX.[14] Your first step will be to install the necessary TL packages. It is a large download---over 200 MB in size---so it will take a while.

Start up the Synaptic Package Manager and install the following TL components:

texlive 2007-21.r6295.1
texlive-dvipdfm
texlive-dvips
texlive-fonts
texlive-latex
texlive-texmf
texlive-texmf-common
texlive-texmf-dvipdfm
texlive-texmf-dvips
texlive-texmf-fonts (This is by far the largest package---183 MB)
texlive-texmf-latex
texlive-texmf-usrlocal
texlive-xdvi (which will bring in texlive-mfwin with it)

Selecting these components will automatically pull in these packages also: bison, flex, libkpathsea4 and libteckit0.

5.1.2. Installing Kile

Although you can run LaTeX from the command line (Konsole) and edit .tex files with a text editor such as vi, Kwrite or Leafpad, it is much easier and more pleasant to use a dedicated LaTeX editor like Kile (KDE Integrated LaTeX Environment). Kile features niceties such as syntax highlighting, auto-completion of LaTeX commands, and it allows users to compile, convert and view a document with one click (the QuickBuild feature).[15]

Your next step is to install Kile via Synaptic. You do not need to install kile-doc or kile-extra. Be especially careful to avoid kile-extra because it will attempt to pull in six teTeX-related packages along with it.

Note: One oddity you will probably encounter is that Kile's menu entry will be under Start » Lost & Found

» Kile, so you will have to use the Menu Editor to move it back with the Office applications.

Kile's icon is shown below:

5.2. The Typical Cycle for Running LaTeX

Before we try to run LaTeX on a quick-and-dirty sample file, it will be helpful to have a basic overview of the typical LaTeX production cycle, which goes like this:

1. Create your file and give it the .tex extension. This file contains the text of your document, interspersed with commands (markup) that tell LaTeX how the file should be formatted.

2. Compile the file by running LaTeX on it. This will produce an output file with the extension .dvi (it stands for 'device independent') which you can view or send to a printer.

If you run the variant pdfTeX on your file, the output will be a PDF file. (As an aside, LaTeX will also generate files with the extensions .aux, .log, and may also include .toc, .idx and .bbl files. However, we do not need to worry about those files right now.)

3. View your output to see if it is correct. This step is called previewing.

4. If something is incorrect or you are unhappy with the result, return to the first step and edit your file.

The figure below illustrates this typical cycle:[16]

Start

Create or edit your source file, using a text editor or shell program. Then click on "save".

Run your compiler to create or update your output file.

Look at your output file to see what, if anything, still needs changing.

done   not done yet

convert

finished PDF
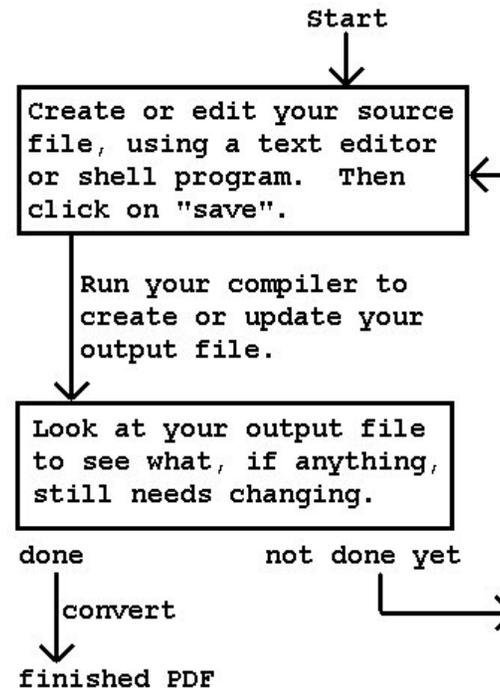
5.3. Your First Document

Now that you have installed TeX Live, Kile and understand the basic stages of LaTeX document

production, you are ready to create your first document. I will help you by providing a ready-made .tex file which will compile without errors.

Note: You will not see a LaTeX entry in your menu structure; instead, you will use Kile to create your .tex file, compile it with LaTeX and view it (both as a .dvi file and a .pdf file).

Follow these steps:

Start up Kile. If you moved Kile, as mentioned in Section 5.1.2 above, it can be found under Start » Office » Kile. If you chose not to move it, Kile will be under Start » Lost & Found » Kile.

Kile's main screen is shown below.
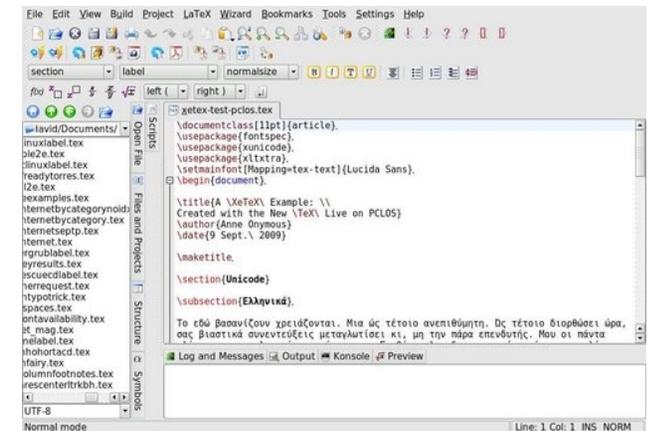
We will focus in a moment on four buttons in Kile's toolbar, which are illustrated in this screenshot:

Copy (or retype) the source file in Section 5.4 below, and create a new file in Kile (File » New » [Template] Empty Document). Then paste the source file from your clipboard into the new document.

Save your file (File » Save) with a meaningful name, such as latex-test.tex.

Use the button in Kile's toolbar (a blue KDE-style gear shown below) to run LaTeX on your source file. It should compile without any errors.

You will see text output scrolling by in the Log and Messages window at the bottom of Kile's main screen. It should end with the message [LaTeX] Done!

To view the resulting DVI file, press the View DVI button (the TeX lion's head superimposed on a sheet of paper, as seen below) in Kile. KDVI should pop up with a display of a one-page document.

Note: Although LaTeX's typographic display is asynchronous with the editor's window—i.e., there is not the immediate feedback you receive with a word processor—the DVI and PDF preview is more accurate than any word processor's WYSIWYG. Considering the speed of modern CPUs and monitors, it takes only a few seconds to compile your source file and preview the result.

If all goes well, you should see a one-page document that looks like this, set in 10-point Computer Modern typeface:

My First LaTeX Document

Yours Truly

September 28, 2009

1   Introduction

This is a short document to illustrate the basic use of LaTeX. Whether it's math, funny foreign accents, grocery lists, or haiku . . . LaTeX can typeset it!

2   A Math Example

LaTeX is especially good at typesetting math. Here's a restatement of the Pythagorean theorem:

$$c = \sqrt{a^2 + b^2}$$

2.1   A Sample Subsection

Does Æschylus have a soupçon of suspicion about Œdipus? Isn't he the fellow from København with the impressive résumé who measures in ångstrom units?
    Here's a short itemized grocery list:

- Milk
- Eggs
- Loaf of bread
- Cat food!

3   A Poem

After that headsplitting theorem, let's close with a haiku by Matsuo Bashō:

Temple bells die out.
The fragrant blossoms remain.
A perfect evening!

1

If you want to generate and view a PDF file, there are two buttons in Kile: one for running pdfLaTeX (a blue KDE gear with a PDF logo, shown below) and another for automatically viewing the created file (the button with the PDF logo). (You can hold your cursor's tooltip over the buttons and see their functions).
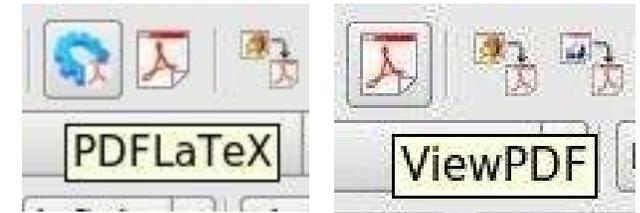
Note: If you plan on sharing your finished document with non-LaTeX users, PDF—rather than DVI—is the format of choice today, and pdfLaTeX can produce it directly. Another advantage of using pdfTeX is that in conjunction with Sebastian Rahtz's ingenious hyperref package, you can create hyperlinked PDFs.

When you finish working, close your .tex file by selecting File » Close from Kile's menu. You will see a message in the Log area that Kile is auto-cleaning the .log and .aux files.

5.4. A Sample LaTeX Source File

Below is a sample file containing the appropriate LaTeX markup; it has been verified to compile without errors. If you copy it exactly as is and follow the instructions in Section 5.3, you will successfully produce your first LaTeX document. (The document

contains a mathematical formula, some foreign-language accents, an itemized list and a haiku. It also illustrates how a document can be divided into sections and subsections).

```
\documentclass{article}
\begin{document}

\title{My First \LaTeX{} Document}
\author{Yours Truly}
\date{\today}

\maketitle

\section{Introduction}

This is a short document to illustrate
the basic use of \LaTeX. Whether it's
math, funny foreign accents, grocery
lists, or haiku

\ldots \LaTeX{} can typeset it!

\section{A Math Example}

\LaTeX{} is especially good at
typesetting math. Here's a restatement
of the Pythagorean theorem:

\bigskip

$c = \sqrt{a^{2} + b^{2}}$

\subsection{A Sample Subsection}
```

```
Does \AE schylus have a soup\c{c}on of
suspicion about \OE dipus? Isn't he
the fellow from K\o benhavn with the
impressive r\'{e}sum\'{e} who measures
in \aa ngstrom units?


Here's a short itemized grocery list:


\begin{itemize}
\item Milk
\item Eggs
\item Loaf of bread
\item Cat food!
\end{itemize}


\section{A Poem}


After that head-splitting theorem,
let's close with a haiku by Matsuo
Bash\={o}:


\begin{verse}
Temple bells die out. \\
The fragrant blossoms remain. \\
A perfect evening!
\end{verse}


\end{document}
```

5.5. Additional Source Files

If you would like more practice typesetting readymade .tex files, there are two standard test files written by Dr. Lamport himself to illustrate basic LaTeX usage: sample2e.tex and small2e.tex.

They are located in the following directory:

/usr/share/texmf-texlive/tex/latex/base

Navigate there, make a copy of the files and save them in your /home directory or your Documents folder. Follow the steps in Section 5.3 to compile and preview them, and you will be the proud creator of three LaTeX documents.

Wishing you happy TeXing!

Bibliography

1. In his introductory tutorial, Tobias Oetiker claims that LaTeX's essentials can be learned in 141 minutes. See his The Not So Short Introduction to LaTeX, ver. 4.26 (25 Sept. 2008). <http://bit.ly/DtFwX>.

2. Leslie Lamport, interview with Günter Ziegler, "How (La)TeX Changed the Face of Mathematics: An E-interview with Leslie Lamport," TUGboat 22 (2001): 21. <http://www.tug.org/interviews/>.

3. Leslie Lamport, LaTeX: A Document Preparation System: User's Guide and Reference Manual, 2nd ed. (Boston: Addison-Wesley, 1994) 5.

4. "Just What Is TeX?" <http://www.tug.org/whatis.html>. To learn more about Knuth's personal side, see Kara Platoni, "Love at First Byte," Stanford Magazine May/June 2006. <http://www.stanfordalumni.org/news/magazine/2006/mayjun/features/knuth.html>. Despite his brilliance, Knuth is well known for his modesty, kindness and devout Lutheran faith. NPR's Morning Edition broadcast a fascinating interview with him in 2005, titled "Donald Knuth, Founding Artist of Computer Science"; it can be heard at <http://www.npr.org/templates/story/story.php?storyId=4532247>.

5. The official "TeX" logo is properly typeset with the "E" below the baseline and reduced spacing between the letters; it is produced by the command \TeX. The command \LaTeX produces the logo "LaTeX."

6. Donald E. Knuth, The TeXbook (Boston: Addison-Wesley, 1986) 1.

7. Nicola Talbot, "What Is TeX?," comp. David Walden, PracTeX Journal 2005.3. <http://www.tug.org/pracjourn/2005-3/walden-whatis/>.

8. Allin Cottrell, "Word Processors: Stupid and Inefficient" 29 June 1999. <http://ricardo.ecn.wfu.edu/~cottrell/wp.html>.

9. Lamport, "How (La)TeX Changed" 22.

10. Antoni Diller, LaTeX Line by Line: Tips and Techniques for Document Processing, 2nd ed. (New York: John Wiley & Sons, 1999) 2.

11. Most of these points are taken from Peter Flynn's "The LaTeX Brochure" <http://silmaril.ie/downloads/>.

12. For a complete list, see "TeX User Groups around the World." <http://www.tug.org/usergroups.html>. TUG publishes the journal TUGboat three times per year, covering a range of topics in digital typography.

13. George Grätzer, First Steps in LaTeX (Boston: Birkhauser; New York: Springer-Verlag, 1999) xv.

14. teTeX is a TeX implementation for Unix-like systems which was maintained by Thomas Esser (hence his initials te in the prefix) from 1994 until May 2006. Esser stopped maintaing teTeX at that ime and recommended that his users switch to TeX Live as a replacement.

15. There are other LaTeX-capable text editors available in the PCLOS repositories: Emacs-AUCTeX, Geany, Gedit and AmyEdit. AUCTeX is probably the most powerful; it is an extensible package for handling .tex files in Emacs. (AUC is the initialism for Aalborg University Center [Denmark], where it was first developed). Gedit has a LaTeX plugin and Geany has a LaTeX mode. Some other Linux distributions, e.g., Fedora and Ubuntu, have Texmaker, the Vim-LaTeX suite and Winefish available.

16. Eric Schechter, "Beginners' Introduction to TeX and its Use." 13 Oct. 2005. <http://bit.ly/15cMZP>.

PCLinuxOS Beautification 2009 Onwards

Scanner Saga: Part 2

by Meemaw

Last month I gave a little instruction on how to use your scanner with Xsane. This time, we'll look at Kooka, which is KDE's scanner program.

Kooka is part of the package KDEGraphics. The whole package has many useful programs, including kamera, kcoloredit, kruler, ksnapshot, kuickshow, kview and several others. They can be installed separately or the whole KDEGraphics Suite can be installed. I have three or four of them. When it is installed, Kooka is in the Multimedia--> Graphics section of Kmenu. When it is first opened it looks similar to this:
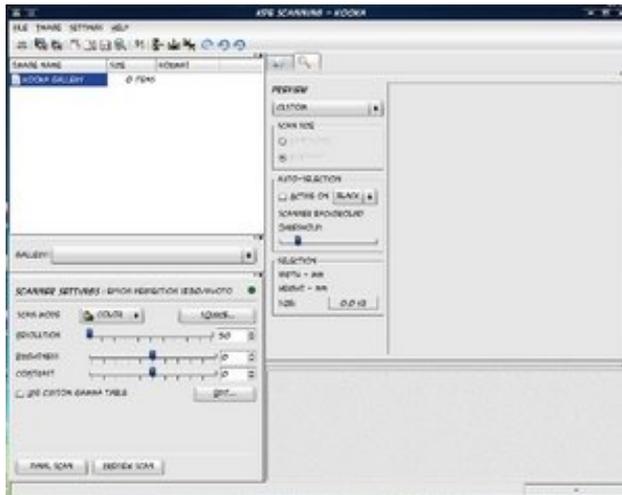
At top left is the "Gallery" list, at bottom left is the settings pane, at top right is the preview pane and at bottom right is the thumbnail pane.
The Gallery is where Kooka saves a file of each final scan: you can keep them or delete them when you

are finished. This is a handy option to have! If you are cleaning up your /home folder and delete a scan you needed, you can come back here and retrieve it, if you kept it. You can also try different settings to see which looks best before you save that picture you want to be really special.
The settings section doesn't seem to have as many choices as you have in Xsane, but notice that part of the choices are in the preview section rather than all settings being together.

The toolbar at the top has the following tools:
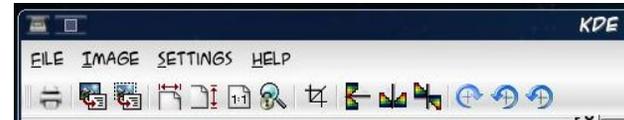
Print, two OCR settings, four zoom settings for the thumbnails, the crop tool, three settings to flip your image, and three settings to rotate your image.
When you put something in the scanner and click preview scan, you will get something like this:
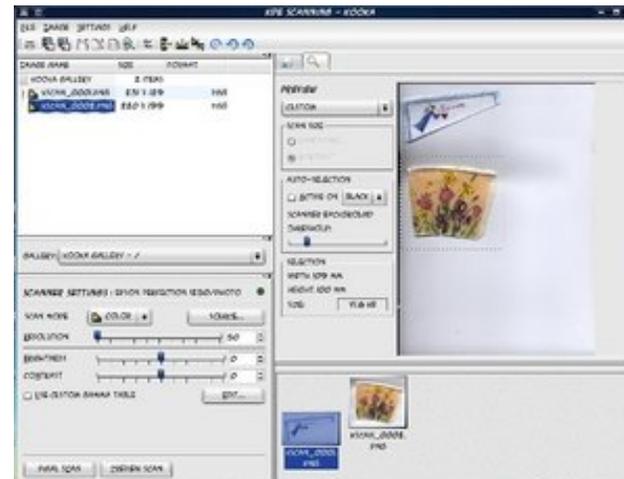
Yep it's that angel again, along with a flower picture (actually it's a paper drinking cup, but I like flowers.) I have already done two final scans, which are saved down in the thumbnail section. They don't have to be done separately. (By that, I mean you don't have to put each item in separately.) In the toolbar, there is a crop tool you can use. Click on that and then "draw a box" around the image you want in your final scan. Click on final scan and you'll get only that image. So I did the angel, then went back and chose the cup. Two things scanned and saved quickly! Works well with multiple pictures as well.
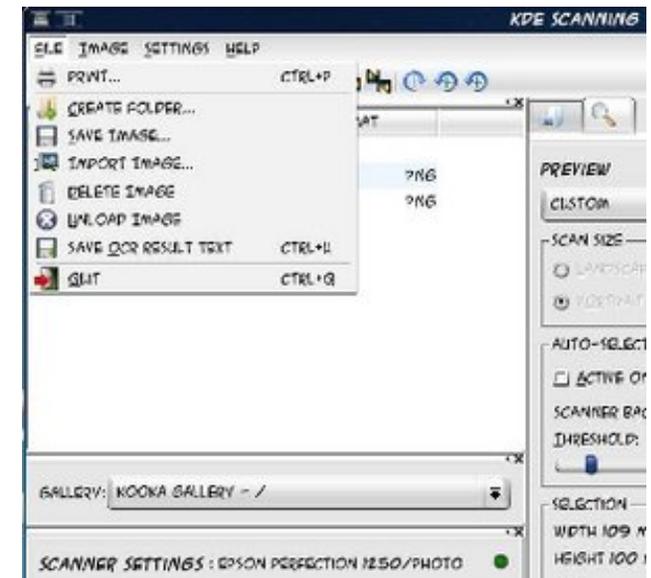
You can also go to the "File" menu and choose "Save Image" - you will get a window asking you to name your file. That one will be saved in your /home folder.

Two features of Kooka that make it different from Xsane is that (1) everything is in the same window and (2) you can rearrange the window the way you like it. I rearranged mine with the settings and preview on the left and the gallery and thumbnails on the right (my last preview was a document.)
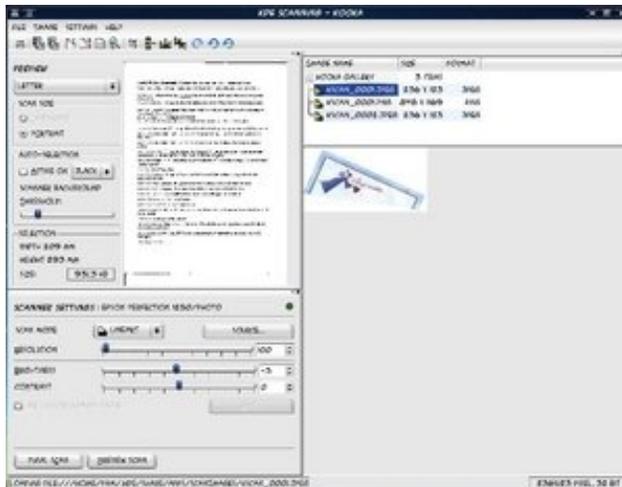
The easiest way to move things is to grab the top border of each section with your mouse and drag it where you want it. You might have to resize it, but it will be moved. You can always close that section by clicking on the x on the right end of the border. If you click on the arrow, it will pop that section loose from the others. To reattach it, just click on the arrow again. If you want things in more than one window, you can always leave one section "popped out."

If you have any trouble, I refer you again to the very capable members of the forum.

Screenshot Showcase

Uploaded by bones113, October 11, 2009, KDE 4.3.2

Double Take & Quick Gimp Tip

Anasers on Page 33

Find at Least Seven Differences Between Cartoons!

Mark's Quick Gimp Tip

If you're like me, you have lots of images piling up in your system. And, if you use The Gimp, you probably like to use certain parts of images and incorporate them into new images and designs. In order to have this "clip art" at hand, you need an image library. An image library is a central place that stores all these images for easy retrieval. On the Windows side, most graphics apps have built-in image libraries. You can drag and drop clip art to the library for later use. The problem is, these libraries are buried deep in the Windows directory. Moving the library (and all those images) when you upgrade or move to another computer system can be a hassle. You have to find the library's exact directory location, and this usually requires a Google search or combing through the app's help area. Plus, you have to place that image folder in the exact same directory loaction on the new Windows system in order for the app to once again find it. Worse still is if you ever need to reinstall the graphics program. The re-installation routine sometimes overwrites the entire library. If you didn't back it up, all that clip art is lost! Luckily, The Gimp in Linux makes it much easier to create an image library. Simply create a folder anywhere in your home directory and store the images there. That's it! (I call mine "Gimp Artwork." It sits on my desktop) When you want to place an image into a photo you're editing, simply open the folder, grab the file thumbnail and drag and drop into the Gimp work area. The Gimp creates the clip art as a new layer. Conversely, you can easily add images and art clips to your library. Just select the area of the image you want and copy it to your clipboard. Then, open the library folder, right click and select "paste clipboard contents." You can then add a file name and select the image format.   I like to use those file formats that preserve a transparent background (.tif, .png). If you want to save as Gimp's native .xcf format (or other file format), copy the area, paste as a new image, and simply save the image to the library folder.   The great thing is, with Gimp running in Linux, you'll always know where your image library is. You'll always be able to take it with you, move it to other Linux systems running Gimp, and easily back it up.

Mark Szorady is a nationally syndicated cartoonist. His work is distributed by georgetoon.com. Email Mark at georgetoon@gmail.com.

Forum Foibles
Match the Nerd

?	Rike	*I did the test and never got the result.*	A. Innocent nerd
17	taelti	*I am getting nerd literature and I am going to totally immerse myself.*	B. Smug nerd
44	Lone Stranger	*I suck. Gotta pull out the computer and linux books.*	C. Infidel nerd
64	coolbreeze	*No point asking me anything.*	D. Undecided nerd
72	Hootiegibbon	*Well I could do with more time to be nerdy.*	E. Eyeore nerd
79	gseaman	*I knew the 'correct' answers to score higher, but I had to tell the truth!*	F. Lost nerd
79	cubbybear	*I could have changed answers for a higher score, but decided not to.*	G. Goody Goody nerd
80	JohnBoy	*I am a failed nerd if posts here are anything to go by.*	H. Cross My Heart nerd
86	T6	*Well, I'm a nerd.*	I. Mum's the Word nerd
89	weirdwolf	*also 67*	J. Wimpy nerd
89	Texstar		K. Hillbilly nerd
96	Neal	*Who? Me?*	L. Boastful nerd
97	AndrzejL	*I dont believe I am that nerdy.*	M. Gee Whiz nerd
97	Sproggy	*Would I be anything else?*	N. Teacher's Pet nerd
100	rudge12000	*Hey I got 100% on the test after some research and only two tries.*	O. Get Up and Go nerd
100	parnote	*I have no reason to be boastful.*	P. Timeless nerd

Multimedia Mean Machine, Part One: Mencoder

by Paul Arnote

You may not realize it, but when it comes to converting multimedia formats, you are sitting on a couple of powder kegs. They have the potential to make your multimedia conversion projects quick and simple — once you understand how to use the tools. These programs go by the names **mencoder** and **ffmpeg**. In this article, we will focus on learning how to use mencoder. We'll take a closer look at ffmpeg in a later article.

Mencoder is in the PCLinuxOS repository. Related to Mplayer, mencoder is the tool to encode multimedia files. In fact, both programs share the same online documentation at http://www.mplayerhq.hu/DOCS/HTML/en/mencoder.html.

Mencoder is so powerful, and has so many options, that we will only be able to explore some of the more common and more frequently used options. If you want more information, I humbly refer you to either the web site mentioned above, or to the extensive man pages for Mplayer, which contain the information for mencoder.

As much as most users dislike the command line, mencoder is a command line utility. Face it — sometimes, the command line is the fastest and easiest way to get something done. Granted, someone probably could create a GUI for mencoder, but it definitely would be difficult, given the wide variety of options available for mencoder. Several attempts have been made. Some such attempts (that are in the PCLinuxOS repository) are: AcidRip, Kmenc15, Kalva, Kmediagrab, and rm-ogv-flv-to-avi, to name a few.

You will find that there is little that mencoder cannot do. If you have the codec installed on your system, or can play the codec back on your system, chances are high that mencoder can convert between those installed formats.

To find out which video codecs mencoder supports on your system, enter:

```
mencoder -ovc help
```

It's easiest to think of the -ovc command line switch as "output video codec." The above command will list out all of the video codecs supported on your system.

Similarly, to find out which audio codecs mencoder supports on your system, enter:

```
mencoder -oac help
```

Much as its video counterpart, the above command is easiest to remember as "output audio codec," and will list out all of the audio codecs supported on your system.

Once you start to see a few command line examples, you should start to see the pattern, along with all the possibilities. For example, to convert an AVI file to FLV, you can use the following command:

```
mencoder -forceidx -of lavf -oac
mp3lame -lameopts abr:br=56 -srate
22050 -ovc lavc -lavcopts
vcodec=flv:vbitrate=250:mbd=2:mv0:trel
l:v4mv:cbp:last_pred=3 -vf
scale=360:240 -o /path/to/output.flv
/path/to/source.avi
```

-forceidx forces the rebuilding of the index of the file.

-of lavf specifies the output format to be encoded with the libavformat muxers (combining audio and video).

-oac mp3lame -lameopts abr:br=56 -srate 22050 specifies for the audio to be formatted as 56 kbps MP3, at a 22050 sample rate.

-ovc lavc -lavcopts vcodec=flv:vbitrate=250:mbd=2:mv0:trell:v4mv:cbp:last_pred=3 -vf scale=360:240 tells mencoder to encode the video using the libavcodec codecs, to specifically use the flv video format, with a video bitrate of 250 kbps, using macroblock decision algorithm 2 to produce the best rate distortion, to encode each macroblock of the video with MV=(0,0) and to use the better one, to find the optimal encoding for each 8x8 block, allowing 4 motion vectors per macroblock for better quality, to find the optimal coded block pattern to minimize distortion, the amount of motion predictors from the previous frame, and to scale the video to 360 x 240 pixels.

-o /path/to/output.flv /path/to/source.avi tells mencoder to produce the specified output file from the specified source file.

See what I mean about mencoder having so many options? Don't worry if you don't completely understand all the options. The more you work with multimedia files, the more these things will make sense. In a way, it's like learning a new language. All of the options in the above command are explained in the Mplayer/mencoder man pages, and at the web site mentioned near the beginning of the article.

To convert an MPG file to an AVI file, try this short and simple command:

```
mencoder /path/to/file.mpg -o
/path/to/file.avi -ovc lavc -oac lavc
```

You can expand the above command to include the type of audio codec you would like to use in the conversion process. The command:

```
mencoder /path/to/file.mpg -o
/path/to/file.avi -ovc lavc -oac lavc
-lavcopts acodec=pcm
```

will create the audio portion of the AVI file in uncompressed PCM format. Similarly, the command:

```
mencoder /path/to/file.mpg -o
/path/to/file.avi -ovc lavc -oac lavc
-lavcopts acodec=libmp3lame
abitrate=192
```

will create the audio portion of the AVI file as MP3 audio at 192 kbps bit rate. If you use libmp3lame, you can also use -lameopts to specify additional options to use in the MP3 conversion process. The command:

```
mencoder /path/to/file.mpg -o
/path/to/file.avi -ovc lavc -oac
mp3lame -lameopts vbr=2:q=5
```

will create the audio track of the AVI file as variable rate MP3 file. The "q" parameter can be any value between 0 and 9.

Just as you can specify the audio codec to use in the finished file, you can also do the same thing with the video codec. The command:

```
mencoder /path/to/file.mpg -o
/path/to/file.avi -ovc lavc -oac lavc
-lavcopts
acodec=libmp3lame:abitrate=160
vcodec=xvid
```

to compress the MPG file into an XVID AVI file, with MP3 audio at 160 kbps.

You can, instead of going through lavc (libavcodec), use XVID directly. To create an XVID file without going through lavc, use the following command:

```
mencoder -ovc xvid -oac mp3lame -o
/path/to/output.avi
/path/to/source.avi
```

Since we are addressing XVID directly, via mencoder, we can also spedify other XVID options. For example, to specify a specific bit rate for our XVID AVI file, we can issue the following command:

```
mencoder -ovc xvid -oac mp3lame
-xvidencopts bitrate=700 -o
/path/to/output.avi
/path/to/source.avi
```

The above command will create an XVID encoded AVI file, with the video bit rate set to 700 kbps. Naturally, the higher the bit rate, the better the quality of the video. But the trade-off for using a higher bit rate is that it also results in a larger file size.

You can, however, maintain higher quality while using lower bit rates (and a smaller file size) by using two-pass encoding. With two-pass encoding, the source file is scanned to analyze the content, and that information is written to a file. On the second pass, the video undergoes the actual encoding, using the information gleaned from the analysis of the file on the first pass. To do a two-pass XVID encoding of an AVI file, use the following command:

```
mencoder /path/to/source.avi -oac
mp3lame -lameopts abr:br=128 -ovc xvid
-xvidencopts pass=2:bitrate=550 -o
/path/to/output.avi
```

Mencoder also allows you to transcode video directly from an input source, like a DVD. Use the

following commands to create an 800 kbps XVID encoded AVI file:

```
mencoder dvd:// -oac mp3lame -ovc xvid
-xvidencopts pass=1 -o /dev/null
mencoder dvd:// -oac mp3lame -ovc xvid
-xvidencopts pass=2:bitrate=800 -o
/path/to/output.avi
```

Adjust the video bit rate to your liking. Again, higher bit rates result in larger file sizes, but also with higher quality video. And a video bit rate of 800 kbps will give you fairly good quality video.

So would you like to create an XVID encoded AVI file, from a DVD, that will fit onto a 700 MB CD? Use the following command:

```
mencoder dvd:// -ovc xvid -oac mp3lame
-xvidencopts bitrate=-700000 -o
/path/to/output.avi
```

Setting the bit rate to a negative number (in this case, -700000) allows XVID to set the video and audio bit rates to fit within the size constraint specified.

If you choose to do any of the previous three video translation scenarios, be prepared to go have dinner or something. Either of them will require a fairly high CPU load, and you will not be able to use your computer for much else while the video is being converted. Of course, you can make mencoder behave "nicer" with the rest of your system, so you may be able to at least read your emails or browse the web while the conversion is underway. You can

do that like this:

```
nice -n 19 mencoder dvd:// -ovc xvid
-oac mp3lame -xvidencopts bitrate=-
700000 -o /path/to/output.avi
```

Maybe you have a video that you want to add background music to. Enter this command:

```
mencoder -ovc copy -audiofile
/path/to/the/mp3file.mp3 -oac copy
/path/to/the/source_video.avi -o
/path/to/the/output_video.avi
```

Or, perhaps you have a video file that you would like to resize. As we mentioned earlier, the higher the video bit rate, the larger the file. Well, another thing that contributes to the video file size is the size of the video image. In the following example, we can resize the video to 320 x 240.

```
mencoder /path/to/the/source_video.avi
-ovc lavc -lavcopts
vcodec=mpeg4:mbd=2:trell -oac copy -vf
scale=320:240 -o
/path/to/the/output_video.avi
```

Sometimes we have two (or more) video files we would like to join together. You can issue the following command:

```
mencoder -forceidx -ovc copy -oac copy
-o /path/to/the/output_video.avi
/path/to/the/source_video_part_1.avi
/path/to/the/source_video_part_2.avi
```

You may also wish to only make a clip of a portion of a video. Again, mencoder comes to your aid. Use the command below, replacing the starting point (ss) and ending point (-endPos) in the command below with the values that work for your particular project.

```
mencoder -oac copy -ovc copy -ss 0
-endPos 54
/path/to/the/source_video.avi -o
/path/to/the/output_video.avi
```

Did you receive a portable digital movie player as a gift, but it won't play Ogg/Theora video? Again, mencoder can come to your aid. Issue the command below to convert your OGV file into an AVI file that you can play on your movie player:

```
mencoder ./out-1.ogv -o ./file.avi
-ovc lavc -oac pcm
```

I'm sure you've noticed many videos on YouTube that have background music, as well as many older videos of live performances by your favorite band or bands. You can easily get an MP3 copy of the audio, simply by issuing this command:

```
mplayer -dumpaudio name-of-file.flv
-dumpfile name-of-file.mp3
```

Now I know this situation has happened to everyone reading this article: you sit down to watch something on TV (it always happens when it's something you've really been wanting to see), and you start feeling sleepy. You just know that there is no way you are going to be able to stay awake to watch the rest of the television program. Once again, mencoder can

come to your rescue. By issuing the following command, mencoder should record from your TV tuner card, and continue to record for two hours (sorry, but I don't have a working TV card to test this with):

```
mencoder tv:// -v -tv drive
r=v4l:width
=720:height=576:input=1:device
=/dev/video0:immediate mode
=0:/alsa:adevice=hw.0,0:outfmt=yv12 -o
output.avi -af volume=10:0 -ovc lavc
-lavcopts vcodec=mjpeg:aspect=4/3
-aspect 4:3 -oac pcm -endpos 02:00:00
```

Finally, the last one comes from kolosus, in the PCLinuxOS Forum. Here are the mencoder commands he uses to convert videos to a format compatible for playback on his BlackBerry.

To convert an existing AVI file:

```
mencoder -vf scale=240:180
/home/syed/Koop\ -\ Koop\ Island\
Blues.avi -o ~/Koop.avi -of avi -ovc
lavc -oac mp3lame -lavcopts
vcodec=mpeg4:vbitrate=240 -lameopts
br=64:vol=9
```

To convert a DVD to an AVI file (With english captions, no less!):

```
mencoder -vf scale=240:180 dvd://1
-dvd-device /mnt/cdrom/VIDEO_TS -slang
en -o ~/chinese.avi -of avi -ovc lavc
-oac mp3lame -lavcopts
```

```
vcodec=mpeg4:vbitrate=240 -lameopts
br=64:vol=9
```

Of course, if your device is capable of playing back larger sized videos (e.g., the BlackBerry Storm can play back video at 480 x 360), you can change the scale values to reflect the larger video size.

As you can see, the possibilities with mencoder are virtually endless. And, as you are likely to find out, there are usually multiple ways to achieve your goal. Mencoder truly is a powerful tool for your multimedia arsenal. In a later article, we'll take a look at the other powerful multimedia tool on your system: ffmpeg.

Screenshot Showcase

Uploaded by critter, October 11. 2009, KDE 4.3.2

Gadgets & Gear:
Silvercrest OM1008 Wireless Mouse

by AndrzejL

There is this moment in the life of every single computer user when his old and favorite mouse "dies," and it has to be replaced. Will you go to the computer shop and buy just about anything, or will you hunt for the perfect mouse? Well, it wasn't such an obvious choice to me. I was hoping for the perfect mouse, but my wallet was squeaking for something VERY cheap.

So there I was at the crossroad of choices, and the need for new mouse was trying to speed up my decision. This is not the greatest feeling, I must say, as I hate making a decisions, and I hate it even more when the decision has to be made fast. Not the greatest material for the ER surgeon, I know, but I do OK in my normal, quiet, sometimes even boring, life.

As always on Tuesday, I was doing my weekly shopping in the Lidl store. I was walking around, picking up fruits and vegetables, and all of the sudden it hit me. "They were selling computer mouses in Lidl a while back, and I remember thinking that they looked pretty decent. Maybe they still have them." Well they did. In fact, they had two designs to choose from. The first one was smaller. It looked and felt light. Grayish. Nothing too fancy at first sight.

The other was looking much better to me. Bigger. Slightly heavier. It felt more reliable. I liked the ergonomic shape and the darker color scheme. The boxes that they were in allowed me to grab the mouse, without unpacking it. Solid grip. It felt good.

Brand? Silvercrest. What do I know about Silvercrest? They come from Germany. I also know that I have 2.5 year old wireless headphones from them that still work like a charm. Two pairs of them, in fact. Well, they cant be bad then. I took a closer look at the mouses. They were both wireless working on 2.4 GHz frequency, with programmable buttons. I remember thinking "Will it be really hard to get them to work in PCLinuxOS?" Its really hit and miss with Linux sometimes to get your new toy to work. I looked at the price. €14.99. Not bad. Three years warranty. Even better. Then my Lady said something that convinced me. "We can always bring it back if its not working". That's the truth. Lidl's return policy is very good. If it ain't working as expected, bring it back. You will get either a new item, or your money back. OK. OK. Enough of trying to find the reasons not to buy it. So I took it. My next thought was, "Is it battery hungry?" So I got an 8 pack of AA batteries too, just in case.

Guess what was the first thing I took out from the bags when we came home? Groceries? Heck no :D! My new toy! I have opened the box. Mouse, Mini USB Transceiver, user manual, drivers CD, and some no name battery. "Wow... Only one battery goes inside? I will have to change them every 2-3 days. Good thing that I got that 8 pack of batteries

after all. But on the other hand, Silvercrest thinks of everything."

First things first. Lets put the battery in. It was as easy as opening the flap at the bottom of the mouse and inserting the battery. Then, I turn the mouse on by flicking the switch at the bottom of the mouse. OK. The mouse started living! The red light (which was good in this case) was on :). I plugged in the USB transceiver, AND MOUSE POINTER ON THE SCREEN MOVED. "It cant be that easy. Where is the catch?" Well there was no catch, and it was THAT EASY. The Silvercrest OM1008 Mouse is truly a plug and play device, in my opinion.

I thought, "How about getting the extra buttons to work?" So I started messing around in the Firefox and Googling stuff. And while I was at it, I pressed one of the side buttons for the heck of it. WOW! What just happened? I was back at the Google site. Hmm did I pressed the Back button? No. Then what happened? By pressing the side buttons on the mouse, I am jumping backwards and forwards through the pages in Firefox! No set up was required. It was working out of the box.

I went further playing in Firefox. Wheel vertical scroll. Check. Wheel horizontal scroll? Check. Quake 3 and Arena Linux Client assigning the side button to ZOOM? CHECK! As easy as under Windows. I rest my case. This mouse IS AWESOME.

I've been using it for three weeks now. I never had a chance to unpack the AA batteries pack because the no-name battery that came with the mouse continues to work.

I decided to test how far I could go with the mouse. It stops working about 5-6 meters from the computer, when I reach the hallway and the wall gets between me and the computer. Five meters? Why would I go 5 meters away from my laptop? Well, I am glad it does not keeps me chained up to the desk ;).

The mouse hasn't gotten scratched – so far. That's another good thing. I have hands of a blacksmith. I mean that I break gentle stuff easily. I broke my

Wireless Optical Mouse OM1008

Overview
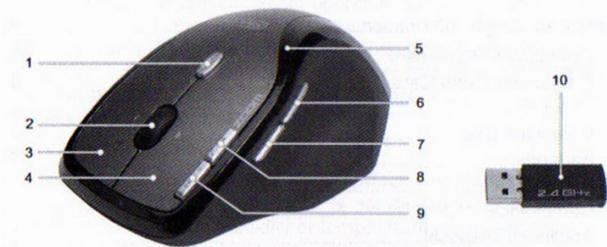
#	Description
1	Toggle 500/1000 cpi (pressed briefly) Toggle standard / media operation (pressed continually for 3 seconds)
2	Tilt wheel (standard) Volume control and sound off (media operation)
3	Right mouse button
4	Left mouse button
5	Battery power control lamp
6	Forward (standard) Next title (media operation)
7	Back (standard) Previous title (media operation)
8	Zoom + (standard) start Media Player (media operation)
9	Zoom – (standard) Play / Pause (media operation)
10	Miniature USB receiver

fathers favorite brass sculpture just by holding it. Want to guess why I don't own a wrist watch?

Well, I know. I get excited so easily. But, if that is not something to be excited about, then I don't know what is. By the way, I have tried using the mouse with the PCLinuxOS Live CD too, and it works out of the box.

Now few technical details:

From a Konsole window, lsusb lists the mouse as:

Bus 002 Device 005: ID 04fc:0538 Sunplus Technology Co., Ltd

The PCLinuxOS Control Center Hardware Browse section shows:

Description: Silvercrest OM1008
Module: usbhid

And, the /etc/X11/xorg.conf section for this device looks like this:

```
Section "InputDevice"
    Identifier "Mouse1"
    Driver "mouse"
    Option "Protocol" "ExplorerPS/2"
    Option "Device" "/dev/mouse"
EndSection
```

I know that by editing the file /etc/X11/Xmodmap I could change the settings for the programmable buttons. But, so far I didn't found a reason for that,

since the mouse does exactly what I wanted, how I wanted, and even more.

Would I buy another product from Silvercrest? I already did. They had 2.1 speakers for €20 last week.
(http://service.targa.co.uk/dokumente/Silvercrest_SP2124.jpg).

AndrzejL

Editor's Note: For North American readers, the Silvercrest OM1008 does not appear to be available outside of Europe. However, Newegg has a very similar wireless optical "super mouse" (actually, called a "gaming mouse"), made by Wolf King, at http://www.newegg.com/Product/Product.aspx?Item=N82E16826620004 for under $30 U.S.

by Gary L. Ratliff Sr. (eronstuc)

Lisp was one of the earliest computer languages invented. It appeared in the late 50's and stood for List Processing. It was used in artificial intelligence and was very powerful for regular programming. In the mid 70's a graduate engineering student enrolled in MIT and worked in the MIT AI lab. His name was Richard Matthew Stallman, and programmers preferred to be called hackers in those days before the word acquired its current evil connotation.

A very early full screen editor was TECO (Text Editor and Corrector which was then in use in the AI Lab. Many of the hackers had written macros for this editor which extended its capabilities. Stallman wrote many and then cataloged them and this set of macros was placed on the computer. Because no program on this system started with the letter E, this group of macros for text editing came to be known as Emacs. The features of the program continued and the size of the program and number of its features grew. The core of the current Emacs editor is written in C and the program features name extensions which are loaded into memory when they are asked for. These extensions are written in a dialect of Lisp called Elisp by old hands or more properly Emacs Lisp.

Getting Your Computer to Understand Elisp

The editor which comes standard with the KDE version of PCLINUXOS 2009-1 is kwrite. I just installed the GNOME 2009-2 version and its default editor is gedit. This was verified while the initial boot from the CD was being installed into the most unused partition of the main system. Gedit and Kwrite are very similar. However, by default you will

not have the emacs editor installed. By going to Synaptic and doing a search for emacs, you will be able to have this editor installed on your system.
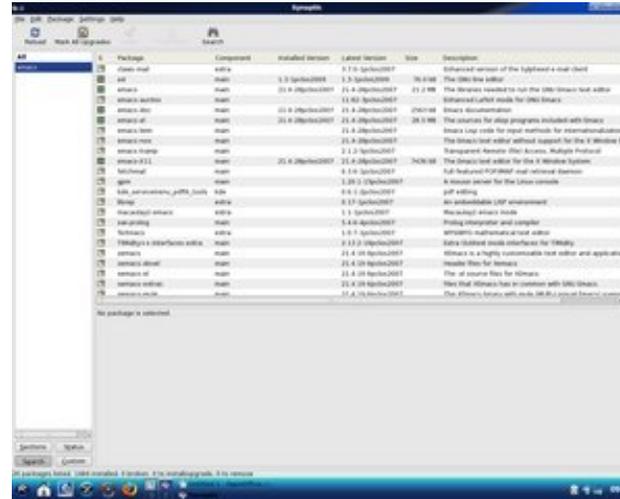
Here we see that the items we desire to install from emacs have been selected and installed by using "apply" now they are marked as having been sucessfully installed into the system.

Here from the screen shot you can see that I also elected to install the documentation for emacs as well as the elisp source code and support for the X11 version of emacs. There is also another version of emacs named Xemacs. For the purposes of this article we will only install the emacs versions. Now you will run the emacs program from: More Applications; Editors; emacs on the menu. Once in the emacs editor click on the Buffers Menu item and select: Scratch. You will notice a message to the effect that this buffer is for notes or Lisp evaluation. The one thing to note about Lisp and its dialects is

that you will learn to type the left and right parentheses keys very often. So after the three lines of text at the beginning of the buffer enter the following lines of elisp code:

```
(+ 1 1)
(+ 2 1)
(- 4 2)
(* 8 4)
(/ 20 5)
(+ 1 2 3 4 5 6 7 8 9)
(* 1 2 3 4 5 6 7 8 9)
```
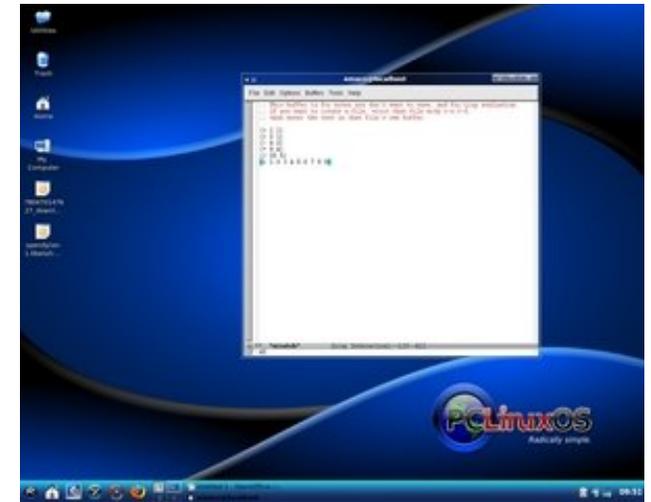

The scratch buffer from Emacs has had some simple forms entered. The item marked by the current location of the cursor has been evaluated by pressing down the Control key and tapping x and then e. Note the answer appears in the small window at the bottom of the scratch buffer.

Now move your cursor to the space to the right of the right parenthesis in the line: "(+ 1 1)." With the cursor resting in this position hold down the Control key and tap the x key and then the e key. At the bottom of the window another small window should appear which has the number 2 in it. So this has answered the math question: "What is 1 + 1?" Also in the discussion on Dylan it was noted that the decision was made to abandon the prefix notation of the Scheme dialect of lisp in favor of the infix notation. However, Elisp like the many other Lisp dialects uses prefix or Reverse Polish Notation. The operator comes before the arguments. Now by simply pressing the down arrow the cursor should be properly located to be at the next problem.

By again holding the Control key and tapping x and e in succession you should note a 3 appear in the window. Dropping down to the next line reveals 2 and then 32 and then 4. These answer the questions: 2+1 =; 4 – 2 =; 8 * 4 =; and 20 / 5 =.

Now if upon reaching the last mentioned problem you see 5 instead of the correct answer 4 it means that you just pressed the down arrow and your cursor is not outside the right parenthesis.

If in a similar manner you pressed the down arrow to the next problem it would be resting on the number 3 and the number 2 would appear in the answer box as that is the number to the immediate left of the 3. By moving to outside the right parenthesis and again request that the function be evaluated you see the answer 49 appear. This is the e i = 1 to 9. And the answer to the last problem will appear as 362880 which is the Π i = 1 to 9 (the summation and successive multiplication of those integers respectively.)

So it appears that we know how to write simple math functions. In Lisp these are known as forms. So how would we enter simple text? Or how would we tackle the infamous "Hello World!" program? For this next set let us close and then restart emacs. Now enter the following forms for evaluation:

(message "Hello World!")
(message "My name is %s" "Gary")
(message "I am %d years old" 66)
(message "%s had a little lamb and took him to school when she was only %d years old" "Mary" 7)
(message "The first letter letter in my name is: %c " 71)

So in C one would say: "printf("Hello World!\n); while in Dylan it would be written: format-out("Hello World!\n"); Here we see the simpler elisp method. Also notice that the format specifications are the same in Elisp as in C however, notice no , separator is used in the Elisp notation. Also the character is an integer and as such takes a numeric argument in Elisp while in C this argument could have been written as 'G' for a single character. Doing this in Elisp will invoke the debugger. Now the text of the debugger will stay on the screen so this would be a good time to quit emacs and restart the editor.

Now we shall explore how some comparisons are made using Elisp. Enter the following forms into a fresh start of emacs:

(if (< 5 4) (message "yes"))
(if (< 5 4) (message "yes") (message "no"))
(if (> 5 4) (message "yes") (message "no"))
(if (= 5 5) (message "yes") (message "no"))
(not (if (= 5 5) (message "yes") (message "no")))

Now use the Control x e to evaluate these forms and you should see these answers appear in the window at the bottom of the scratch buffer: nil, no, yes, yes, nil. In C these would be the familiar if then and if then else expressions. Now if a value is false then nil is used. (Later we will learn that it also will be used for and empty list.) While if it is true then the symbol t will be used. In the first form as there is no else so the answer is false and the system reports this by showing nil. The second form presents the same question and there is a message to be displayed if the answer is false. So here the answer no is used.

In the third form we change the condition so that it is true and the answer yes if given. Now in Lisp there is no != as there is in C. So to obtain that result the logic of the expression is reversed by surrounding it with the not symbol as shown. And for this the answer is nil.

Now "didn't you mention that LISP stood for List Processing?" you ask. So our next examples show some lists:
(list ())
(list '(a ()))
(list '(a b c))
(list '("a" "b" "c"))

Now when we ask for these to be evaluated we see these answers: (nil) ((a nil)) ((a b c)) and (("a" "b" "c")). Here we see examples of the empty list being noted by nil. Also because these are lists we see this clearly indicated by the surrounding parentheses.

This is just an introduction to the many features of Elisp. I am certain that you might like to explore this

fascinating language further. The creator of Emacs: RMS along with many others wrote the definitive reference for the current and many past versions of Emacs. This is available in many forms and the pdf format version of this is 1017 pages long! This assumes that the reader is currently familiar with the operations of the Emacs editor. There is a similar reference manual available for the Emacs editor itself.

The full reference manual notes that people who are just now learning the language might find the text of: An Introduction to Programming in Emacs Lisp Third Edition more useful for them. Both of these may easily be found by entering Emacs Lisp in the Google search box on the Firefox browser.

A form from Xah's Emacs Lisp Tutorial highlighted and ready to copy using the browsers Edit menu.

Another easy introduction to Elisp is available from

xahlee.org and is listed in the google search engine as: Xah's Emacs Lisp Tutorial.

One final tip as you are reading these tutorials you may wish to try out the lisp forms. Have the text of the article you are reading in one screen and that of the Emacs editor in another. Highlight the text you wish to use and the Copy it then go to the emacs editor on the scratch buffer and paste it into the buffer. Now you may execute the code as mentioned in this article.

Here the form appears in the scratch buffer of Emacs ready to be evaluated. This is pasted into the scratch buffer using the Edit menu and the Paste command.

It's easier than e=mc²
It's elemental
It's light years ahead
It's a wise choice
It's Radically Simple
It's ...

Wiki Wicket: Migrating From Windows To PCLinuxOS

With the recent release of Windows 7, this month's Wiki Wicket is dealing with how to migrate from Windows to PCLinuxOS. Aimed at new users making the switch, even old Linux hands can glean some well-needed refreshers from the material below. This information will be especially useful for those users who have chosen to search for an alternative to paying a significant price for yet another version of Microsoft Windows, or who may be looking for something more secure. — Paul Arnote, PCLinuxOS Magazine Chief Editor

Introduction

This section outlines the similarities and differences between Windows and PCLinuxOS. It's based on material from the previous docs.pclinuxos.com materials, in some cases quite directly so.

The User Interface

Windows gives you its own desktop environment, which (at a given version - XP, Vista etc.) behaves the same and gives the same customization options on whichever Windows machine you use. In PCLinuxOS and other Linux distributions, there is a choice of a number of different desktops (although there is usually a "default" - e.g. Gnome in Ubuntu, KDE in PCLinuxOS, etc.) If you have an older computer, you can pick a light-weight desktop environment. On more current machines you can pick an environment that works more like the Mac, or one that has

really fancy graphics. The default desktop environment for PCLinuxOS is KDE; another option is GNOME. The different environments typically come with a whole suite of software, including window managers, office productivity applications, educational programs, file managers, browsers, CD burners, and so on.

Once you pick a desktop environment, you can then alter it to look like anything you please (called "themeing"), just as in Windows XP and WindowBlinds. You can alter the entire look-and-feel, change the wallpaper, the way the title-bar looks, icon-sets, colors, the mouse-cursor looks, change the look of textfields and buttons and such. In KDE this is done via point-and-click from the KDE Control Center. Additional look-and-feel components can also be downloaded from the web via these control panels.

Here is the main "Appearance and Themes" screen in the KDE control center; note the categories of

appearance etc. that you can alter.

You can also alter the way the desktop behaves; on the left of the KDE control center, below the "Appearance" categories, are some "Behavior" items.

Applications Equivalences

There are a lot of applications which serve the same purpose as those you're used to on Windows. Consult the list of Applications equivalences for more details.

Security and Viruses

Any serious Windows user knows that they need to run firewall and anti-virus software on their system if it is on the network. While Windows XP (post-SP2) has a built-in firewall, previous versions required the user to look for third-party programs to perform that function. Linux comes with everything you need for security, and it's free! And Linux, being a different operating system, isn't susceptible to Windows viruses and spyware! There are a few Linux specific viruses but the ones that exist can't usually do a whole lot of damage. Spyware? Not an issue with
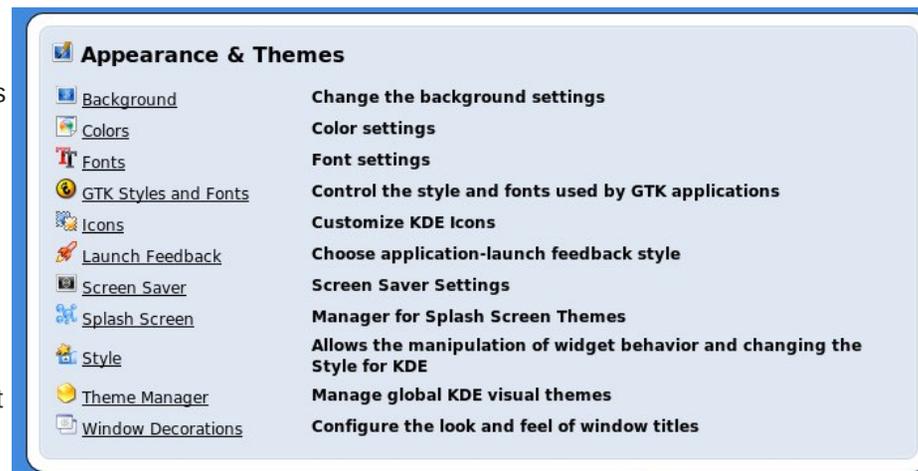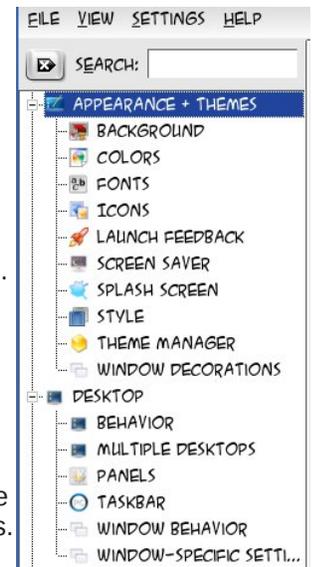
Linux either. PCLinuxOS also comes preinstalled with a personal firewall, just like Windows XP.

What's Running?

Windows provides the "task manager" to list processes on your Windows computer. This is usually run by right clicking on the task bar and selecting from the resulting menu. In the default

KDE environment of PCLinuxOS you get the KDE "sysguard" which provides similar information: process ID, user running it, percent of system being used by it, etc.:

There's no equivalent of the Windows "right click on the taskbar"; find ksysguard in the KDE PC menu or launch it using the Ctrl+esc keyboard shortcut.

Defragmentation

You may have noticed that your Windows system has become slower with age. Some of this is probably due to the additional software that gets installed over time (do you know what all those tasks are which are listed under your user ID in the task manager ? You should!). But some of it may be due to fragmentation of your hard disk - files are stored not in contiguous blocks of the hard disk but in "fragments" - sometimes hundreds per file in extreme cases - as the operating system has to find empty disk area to put bits of a new file into. There is a disk defragmenting tool in Windows but too few people know where it is, and too few know how to get the most out of it (e.g. how to ensure that the page file is not fragmented). Net result: slow performance on memory-hungry applications like video viewing, which gets worse until you do a comprehensive defragmentation.

Now the good news: you don't have to defragment Linux hard disks! Linux filesystems handle that themselves because they avoid fragmenting files in the first place.

That said, when your hard disk is filled up to 99% the system's performance will still suffer, just as with Windows. Make sure you have enough room for the operating system to handle itself, and you'll never have any issues with defragmentation.

International Community PCLinuxOS Sites

NETHERLANDS

TURKEY

DENMARK

CZECHOSLOVAKIA

ITALY

POLAND

Linux Ads Hit The Airwaves

by Paul Arnote (parnote)

Back in September of this year, something phenomenal happened. Linux finally got some advertising airtime on the radio waves. And not just at 3 A.M. when no one but night owls and insomniacs are listening. Oh, no! These ads ran during the weekly Kim Komando radio show -- at least, in one of the radio markets that airs her broadcast.

For those of you who are unfamiliar with Ms. Komando, she is an unrelenting supporter of Microsoft products. To her, Microsoft is not the evil empire that many view it as; rather, she views Microsoft as the harbinger of all that is great in the world of computers.

So there, among the ads for registry cleaners, anti-virus software, and all the other necessary add-ons that Windows users must purchase just to get any resemblence of stability and security, was a different message. A message that computer users DO have a choice. A message that, to many, sounds too good to be true: a FREE operating system, free of viruses, malware, spyware, and all the other computing headaches that afflict the most widely used OS on the planet. And, an OS where the vast majority of the software is also FREE.

Surely, it just can't be true! Where's the catch?! Did someone mention something about swamp land? Nobody these days just gives away anything that is ... well, worth anything.

Ken Starks, author of the "Blog Of Helios," a Linux advocacy blog (http://linuxlock.blogspot.com/2009/09/tux-takes-to-airwve.html), came up with the ads and funded the airing of the radio spots. He started his effort back in 2006. Then, he was approached by "George" (not his real name), a professional voice talent who offered to do the voice-over for the commercials -- pro bono. Without

charge. For free. For the same price as Linux.

Even more astounding, and in the spirit of FOSS, George has given over the rights to the recorded tracks to Starks. And Starks has made the raw audio tracks freely available under the Creative Commons Attribute-ShareAlike 3.0 license -- with no attribution necessary! There is a 30 second spot, and a longer, 60 second spot. According to the "Blog Of Helios," you are free to cut, splice, and lay tracks under them as you wish and see fit. In fact, links to download the files are at the end of this article.

The Results

The ads did, indeed, run for two weeks on KLBJ AM 590 in Austin, Texas. Original plans were for the ads to run for a month, but budget restraints would not allow that to occur. Over the two week time span that the radio ads ran, Starks claims that the ads played 100 times. The 30 second ads ran during the week, and the 60 second spots ran exclusively during the Kim Komando show -- once an hour for the three hour duration of her radio broadcast. But, according to a contact at KLBJ, the ads did indeed run during the last two weeks of September, but only a mix of the 30 and 60 second spots during the weekend programming block, including during the Kim Komando show, and nowhere near the 100 airings that Starks claims.

Perhaps one of the biggest surprises was that there was not an increase in web traffic, as had been anticipated. Rather, people chose to call and talk. Most who visited the web site simply jumped to the "contact us" page to get the phone number so they could call on the telephone.

All total, 179 phone calls, 63 emails, and 4 personal visits resulted from the ads. Starks breaks the inquiries down into three categories: "cautious and curious, interested but skeptical, and those who should never be allowed to touch a computer without professional on-site guidance." He

goes deeper into the results on his blog (http://linuxlock.blogspot.com/2009/10/grand-experiment-linux-ads-on-radio.html) in the follow-up article, which is actually quite humorous -- especially when describing the latter group of respondants to the ad.

Starks calls his "grand experiment" a failure, at least monetarily. He expects the radio ads to net him less than one-half of what it cost him to run the ads. He chalks it up to various factors: some bad decisions in cutting the 30 second ad, spacing and timing of the ads through the week, etc.

Nonetheless, it is a start -- there is always a starting point. Perhaps this is just the first salvo fired. Just maybe, more Linux advocates will pool their resources around the country -- and around the world -- and get more air time for these Linux ads, and to help more people find out about Linux. Afterall, hope does spring eternal.

30 Second Ad: OGG   MP3

60 Second Ad: OGG   MP3

Final 60 Second Ad: MP3

Say What?!

Linus Torvalds was present at the Japan Linux Symposium on October 22, 2009 – the very same day that Microsoft released Windows 7. During a break in the symposium festivities, some attendees decided to have a little fun at the expense of Microsoft, and they captured this posed photo of Linus Torvalds in front of a local shop's Windows 7 display. You have to just love the sarcasm on Linus Torvalds face. The photo, and the tongue-in-cheek caption, appear below, as they were released.

Microsoft tried to torpedo the success of the Japan Linux Symposium by launching their Windows 7 product that same day. They even had setup a big promotion booth across the street from the conference center.

During a break, we decided to make some fun of Microsoft and dragged Linus over there. When we arrived there, Linus was sold immediately on the product as you can see in the picture. At least that's what the sales guy thought. He obviously had no idea who he was dealing with. But in the end Linus surprisingly did not buy a copy. Wise man!

Photo by Chris Schlaeger.
Reprinted with permission.

by Peter Kelly (critter)

In last months tutorial, I presumed that anybody reading it had no experience whatsoever of using the command line. If you worked through that, then you should be ready for a more in depth look. There is nothing too taxing in here, but you may find more text per sub-heading. If you find an area where you come to a brick wall, just walk around it and carry on. Maybe come back to revisit it, or maybe wait for that forehead slapping moment "Dohh!" or even "Eureka!" when enlightenment arrives.

If you followed along with last months installment, you will now have a little experience of typing commands on a command line using the application konsole. So what? You could have done any of those things without having to do all that silly typing.

The application konsole is known as a terminal emulator. It allows you use the command line, without losing sight of your warm and cozy KDE GUI. But what happens if the X System, that is the windowing system that KDE runs on, crashed? Or some configuration file that the system depends upon got corrupted, and when you booted up, you were greeted only by some weird message and an almost blank screen?

A leap in the dark

Press and hold `Ctrl + Alt` and press `F2.`

```
PCLinuxOS release  2009 (PCLinuxOS) for i586
kernel 2.6.26.8.tex3 on an i686 /tty2
home login:
```

Now **that** is a terminal.

Don't panic! Your precious GUI is still around.

`Ctrl + Alt + F7` gets you back.

Actually, you could have pressed any of the function keys from F1 to F6 to get a raw, text only terminal.

I am currently logged into my KDE session as user jane, so if I now drop into a text terminal as before

`Ctrl + Alt + F2`

I am prompted to log in.

```
home login:
jane (enter)
password
```

Yes, in Linux I can be in two places at the same time. I am logged in as jane in my KDE session, and I have now logged in again as jane in this terminal. I now have access to all janes files and can edit them, delete them, move them, rename them and create new ones. I could have logged in as any user that I knew the password of and had access to all of their files. If I had logged in as root then I could have had access to all files on the system and have inadvertently caused chaos. For that reason you should avoid logging in as root at all costs, there are other ways to do things. There are times when it is necessary to log in as root but it is very rare and should only be done if you are absolutely sure about what you are doing.

A change of direction

Type `cd ~` to make sure that you are in your home directory, and then create a new file by using the command.

`touch newfile (enter)` which creates a new, empty file called "newfile"

`echo "this file was created in terminal 2 on " > newfile`

puts some text into the file.

Remember that the > symbol catches the output from the command and puts it into the file (replacing what was there originally so be careful when using it).

Type `date >> newfile`

Using the symbol twice >> catches the command output and appends it to the file.

`exit` logs me out and

`Ctrl + Alt + F7`

puts me back into KDE. Or, more correctly, into the terminal that is running the X System and the KDE environment.

Let's have a look at the contents of the file that we just created.

`cat newfile`

```
[jane@home ~]$ cat newfile
This file was created in terminal 2 on
Sat Sep 19 12:11:24 CDT 2009
[jane@home ~]$ █
```

What we are doing here is known as 'redirection,' and is a very important concept for working on the command line. Most Linux commands are 'stream oriented.' This means that data flows into and out of the command rather like a sausage machine – meat in one end, turn the handle and get sausages from the other end. The data is processed by the command as it flows through the command.

Let me try to explain what happens when you sit down and start to type a command at a terminal running a shell program such as bash. As you press a key (or a combination of keys, like Shift + a), the shell program stores the value of that key press in a special area of memory known as a buffer and prints a copy of it to the screen (usually the screen anyway – see later). It then waits for another key-press to add that also to the buffer. When you press the enter key, it signals the end of that batch of input, and the entire contents of the 'keyboard buffer' are sent to be interpreted. This where bash works its' magic.

Bash takes all of the key presses that you have typed from the keyboard buffer, lays them out on the table into groups that you separated with spaces, looks for any group of characters that it recognizes as a command which it can execute, looks for certain special characters that have a special 'shell' meaning and then decides what to do with the rest of the groups on the table based upon what it has just found.

Usually, this just means that when you enter a command line such as

`ls /home/jane`

bash has two things on the table, `ls` and `/home/jane.`

Now bash recognizes ls as a command and so it looks for input information (how to use the command), any options that may modify the default way that this command performs its' function, and for what to do with the results. (This a very simplified overview but is sufficient for our present needs).

Previously we defined the command format to be

`{cmd} {options} {argument}`

Now we know a little more about bash we can expand this to

`{cmd} {options} {input argument} {output argument}`

`{cmd}` is the name of the command to execute.

`{options}` such as -a or -al are ways of modifying the output or result of the command.

`{input argument}` is anything that you want to send to the command to work with.

`{output argument}` is where you want the results of the command to go.

Now that bash has found a command `ls`, it looks for

a group of key presses on the table that qualifies as a suitable {input argument} for the command and finds /home/jane. If nothing qualifies, then the programmer who wrote the command will hopefully have provided for a 'no input' default condition. ls with no input defaults to the value 'wherever I am now.' There are no options to tell the command to modify its' output, therefore the output will be the default for the ls command – a simple listing.

With no `{output argument}`, most commands default to 'print it to the screen'.

So this command prints a simple listing of the directory `/home/jane` to the screen.

These default values for where the input comes from, and to where the output is directed, can be changed by redirection. This is where you tell bash to temporarily change its habits, and to take instructions from the command line.

The shells default input and output are known as `stdin` and `stdout` (the standard input device – usually the keyboard, and the standard output device, which is normally the screen). (There is a third data stream known as stderr 'standard error,' but let's learn to walk first, eh?)

You can however, redirect data from other sources, or to other destinations such as files. In the previous examples, we have redirected the output to a file, instead of displaying it on the screen by using the > operator. To redirect the input from somewhere other than stdin, we use the < symbol. Try this.

`ls /etc > newfile2` lists the contents of the

directory /etc to the file newfile2.

`sort -r < newfile2` sorts the contents of the file in reverse order.

Here, the ls command takes its input from **`stdin`** (/etc) which was typed into the keyboard buffer and the output is redirected to the file newfile2 instead of being printed to the screen. In the next line, the sort command uses contents of the file newfile2 as its input and, as we haven't specified otherwise, sends the output to **`stdout`**, the screen.

There is a better way to do this using a mechanism which you have seen before. It is called a pipe, and looks like this |.

`ls /etc | sort -r` gives the same result as the two lines above and cuts out the middleman i.e newfile2. The output from the command ls /etc is pushed through a pipe | into the command **`sort -r`**

So what's the difference between **`ls > sort`** and **`ls | sort`**?

This is often a source of confusion. **`ls > sort`** takes the output from the command ls and redirects it to the file 'sort', which it creates if necessary, rather than to the screen (**`stdout`**). Probably not what was intended.

`ls | sort` takes the output and pipes it through the command sort, which in turn sends its' output to the screen (stdout), as this output has not been redirected. In this manner, fairly complex commands can be built up.

Create a small file with a random list of names using some of the bash editing features described last month.

```
touch contacts
echo john > contacts
Use the up-arrow to bring back the
previous line then alt + b and the
delete key to edit the line.
echo amy > contacts   (don't forget to
use the > (append) operator here.)
echo gustav > contacts
echo bob > contacts
echo glenn > contacts
echo simon > contacts
echo george > contacts
Look at the file contents
type
cat contacts
```

```
john
amy
gustav
bob
glen
simon
george
```

```
cat contacts | sort | tr [a-z] [A-z] >
contacts2
```

What did that do?

```
cat contacts2
```

```
AMY
BOB
GEORGE
GLEN
GUSTAV
JOHN
SIMON
```

In the above compound command, the contents of the file contacts is piped to the sort command which, with no options supplied to modify the output, sorts the contents alphabetically from a to z, which is the commands default action. This in turn, is fed to the tr (translate) command which converts any character in the range [a-z] to its uppercase equivalent [A-Z]. Finally, the results from the translation are written to a file called contacts2, which will be created if necessary, or overwritten if it already exists. Don't worry if you don't understand how these new commands work. I just want you to get an idea of how we can 'flow' data from files, through commands and filters, and then write that data to a file or to the screen.

Editing on the command line

When we made the file contacts, we did it line by line, which is obviously unsatisfactory for all but the simplest files. What is really needed is a text editor. We could call up a graphical one that we are familiar with, such as kwrite, but not when we are in a text terminal, as we were when we typed **`Ctrl + Alt + F2.`** If, for example, the X windowing system

won't start, then you may well find yourself in just that position.

The basic editor which you will find in almost every distribution is vi, which we will visit later, as it may well be that one day it is all you have at your disposal. It is however, very powerful editor, though difficult and not very intuitive to use for new users.

Fortunately, PCLinuxOS comes with a very nice, simple edtor for command line work. Meet nano!

Typing `nano` on the command line opens the editor with a blank page. If you specify a file-name after the command `nano`, then it will open that file, if it exists. If no file of that name exists, then no file is created at this stage but you will be prompted to save your work with this file-name when you exit. Only then is the file created.

Typing `nano contacts2` opens the editor with our sorted file, and the cursor is on the first character of the first line.

The screen is divided into four areas:

* The top line of the screen is known as the header bar. This shows the version of nano and the name of the current file being edited. If you didn't specify a file name, then this will read 'new buffer.' If the file has been modified since the last save, then 'Modified' will be shown on the right hand side of the header bar.

* The bottom two lines show a list of command shortcuts. Nano commands are defined by either the control key being held down while the shortcut key is pressed, or by the shortcut key being preceded by the escape key. The caret symbol ^ represents the control key, so for example, `Ctrl + x` exits the program. The escape key is represented by M. These few commands are usually enough for most purposes. `Esc - a` marks text, and `Esc - m` enables/disables limited mouse support. If you want more then there is more, `Ctrl + g` will shw you a brief introduction.

* The third line from the bottom, just above the list of the commands, is the status line which shows that nano read in 7 lines from the file contacts 2.

* The rest of the screen is the editing area.

With the file contacts2 loaded, use the arrow keys to position the cursor on the letter L of the name GLENN. Press `Esc - a` to start marking text. Press the right arrow key 3 times to mark the 3 letters LEN press `Ctrl + k.` This removes the three letters and places them in the cut buffer, a temporary storage area. Press `Ctrl + u`, and this inserts the contents of the cut buffer at the current cursor position restoring the name GLENN. Use the arrow keys to position the cursor at the end of the file, and press `Ctrl + u` again to add the new contact LEN. This is cut & paste, nano style.

`Ctrl + o` prompts you to write out the file with name contacts 2. Pressing `enter` saves the changes and puts you back in the editor. If you change the file name to save the file as, you will be prompted to confirm this, and be returned to the editor with the new file.

Add a few more names then press `Ctrl + x.` Answer y and press enter to leave the editor saving your changes.

This type of simple editor is ideal for beginners to edit Linux configuration files, as it produces only text with no fancy formatting that might be misinterpreted. If you want to write a novel use something else.

Sitting in the bosses chair

For some things you do need to have the special privileges of the root user and the safest way to do this is to use the command `su.` This command

allows you to 'switch user' identity to that of any user you know the password of.

`su john` will prompt you for the password of john. If there is a user account for john, and if the password is successfully entered, then the shell will allow you full access to all of johns files and directories. This is why you should keep your password safe. `su` is a very powerful command.

Typing `su` without a user name will assume that you want to have root access to all files and directories, and will prompt you for the root password.

When you have root privileges you are able to make your system completely unusable!

Let's do some root stuff.

```
[jane@home ~]$ su
Password:
[root@home jane]#
```

Notice that the prompt symbol has changed from $ to #?

Maybe you didn't, but I'm pretty sure that you noticed the prompt is now bright red. This is not always the case, but the developers of PCLinuxOS believe that you really should be aware that you are now in a position to do some real damage and have modified the prompt to reflect that. Notice also that the prompt shows I am working now as root not jane but I am still in janes' home directory. Be aware that the

commands `cd` and `cd ~` will now take you to the directory /root, and not to janes home directory, /home/jane!

One thing that root can do that mere mortals cannot is to add and delete users on the system. To add a new user named john to the system, the command `useradd john` creates the user account, and sets up the user environment by copying the files that the system administrator or the distribution developers have placed in the /etc/skel directory.

It does not add the user to any groups other than the users default group. This can be done here with the -G option, followed by a list of groups, or later with the command `usermod.` We'll cover groups later when we get to file permissions.

You should follow the account creation with

`passwd john`

to create an initial user password for the new john account and then pass this password to the user who, once he has logged in with it, may change it using the same command.

`userdel john`

deletes the user. If you specify the -r option here then the users home directory and any files it contains will be deleted.

There is also a command called `adduser`, which is similar to `useradd.`

Having done our work, we should renounce our special root privileges with `Ctrl + d` or the `exit` command.

```
[root@home jane]# exit
exit
[jane@home ~]$
```

Now check that the account has been successfully created.

```
[root@home jane]# exit
exit
[jane@home ~]$ su john
Password:
[john@home jane]$ cd ~
[john@home ~]$ pwd
/home/john
[john@home ~]$ exit
exit
[jane@home ~]$
```

`su john` logs me in to johns account, but the prompt tells me that I am still in janes' home directory

`cd ~` as I am now logged in as john, this takes me to johns home directory, which I verify with the command

`pwd`

The `exit` command logs me out of johns account and puts me back into janes account, and also back

into whatever directory jane was in when she issued the **su** command.

We'll return to the root terminal later when we have a few more commands to use.

Customizing our environment

After using the command line for a while, you will find that many times you type in the same commands and options over and over. Surely somebody can think of a better way?.

They did. It is called an **alias**, and is a way of giving a command that you regularly use its' own name. You already have some aliases in PCLinuxOS. Type the command alias to show them.

```
[jane@home ~]$ alias
alias cd..='cd ..'
alias cp='cp -i'
alias d='ls'
alias df='df -h -x supermount'
alias du='du -h'
alias grep='grep --color'
alias kde='xinit /usr/bin/startkde'
alias l='ls'
alias la='ls -a'
alias ll='ls -l'
alias ls='ls -F --show-control-chars --color=auto'
alias lsd='ls -d */'
alias mc='. /usr/share/mc/bin/mc-wrapper.sh'
alias md='mkdir'
alias mv='mv -i'
alias p='cd -'
alias rd='rmdir'
alias rm='rm -i'
alias s='cd ..'
[jane@home ~]$
```

Look at one about halfway down the list: **alias ll='ls -l'**

If you type

ll

on the command line bash will interpret this as **ls -l**, and execute it accordingly.

Let's make our own new alias. Suppose that I often want a hard copy of a directory listing with the contents sorted by file size and with these sizes in a format that is easily understood.

To print out files on the printer in Linux we use the command

lpr

The command **lpr myfile1** will send the contents of the file myfile1 to the default printer without the need for any redirection by the user but it is also common practice to pipe the input to lpr from another command.

I want to create an alias that will print out my listing easily and I would like to use the name lspr, but I don't want to conflict with any existing system command. So enter ls and then press **tab** to show a list of all commands that start with the characters ls.

```
[jane@home mydir1]$ ls
ls        lshal      lskatproc  lsof       lspcmcia   lsusb
lsattr    lsinitrd   lsmod      lspci      lspgpot
lsd       lskat      lsnetdrake lspcidrake lss16toppm
[jane@home mydir1]$ ls
```

From this, I can see that there is no command named **lspr** that I have access to and so I am safe to choose this as the name of my alias.

Press **Ctrl + c** to cancel the command.

To create the alias, I use the command

alias lspr="ls -lhSr | lpr"

This tells bash "whenever I type the key combination **lspr** execute the command **ls -lhSr | lpr.**"

This creates a long (option l) directory listing in human readable form (option h), Sorted by file size (option S) in reverse order (option r) and pipes the output to the printer.

Make sure that your printer is switched on and connected, then type

lspr enter.

This way, I don't have to remember how to format the command, just the alias **lspr.**

Unfortunately, as soon as you end this session of bash by logging out or by closing the konsole window, this new alias is lost. To make it permanent, we need to edit one of those hidden files, the ones whose names begin with a period, in your home directory, **.bashrc.**

This is the bash resource configuration file and is read every time a new instance of bash is invoked.

```
  GNU nano 2.0.9              File: .bashrc

# .bashrc

# User specific aliases and functions

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi




                    [ Read 8 lines ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

`nano ~/.bashrc`

will open the `.bashrc` file that is in your home (~) directory, ready to be edited.

Press the down arrow until you reach the end of the file and then add the alias and press enter.

While we are here, copy and paste the following

`export PS1='\[\033[01;32m\]\u@\h > \W \$\[\033[37m\] '`

Make sure that you include the final quote mark ('), then press enter.

Always press enter at the end of a system configuration file to make sure that it ends with a new blank line.

It should now look like the image at the bottom of this column.

Press **Ctrl + x** and answer y to the prompt then press enter to save the modified file. Close the konsole window to end the bash session, and then restart it. This is necessary to enable the new instance of bash to read the modified configuration file.

If all went well, you should be able to type lspr to get your printout, and you should have a nice green prompt to identify you as "not root." If you don't like green, then you can change it by altering the `01;32` part of this line.

```
# .bashrc

# User specific aliases and functions

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi
alias lspr="ls -lhSr | lpr"
export PS1='\[\033[01;32m\]\u@\h > \W \$\[\033[37m\] '
```

`export PS1='\[\033[01;32m\]\u@\h > \W \$\[\033[37m\] '`

Change 32 to a value between 30 and 37 to change the basic color.

Where have 01, we may put several different values:

00 for normal colors
01 for bright colors
04 for underlined text
05 for blinking text
07 for reverse video text

These can be combined e.g. `01;04;05` for bright, underlined, blinking text.

Adding a value between 40 and 47 changes the background color e.g. `1;34;47`

To try out the colors on the command line use

`echo -e '\033[01;37;44mPCLinuxOS - Radically Simple\033[0m'`

and substitute `1;37;44` for the above values separated by a semicolon ;. The -e option added to the echo command tells it to interpret certain sequences of characters, known as escape sequences, rather than just blindly printing them on the screen, which is why we don't see all that gobbledygook on the screen.

```
jane@home > ~ $ echo -e '\033[01;37;44mPCLinuxOS - Radically Simple\033[0m'
PCLinuxOS - Radically Simple
jane@home > ~ $ []
```

Experiment with different color combinations. Maybe login or **su** to johns account and change his prompt to blue. As long as you stay away from the root account you can do no real harm. These are, after all, only dummy accounts.

Do you feel more at home now?

PCLinuxOnline

PCLinuxOS News and Information

L DE

twitter

Want to keep up on the latest that's going on with PCLinuxOS?

Follow PCLinuxOS on Twitter!

http://twitter.com/iluvpclinuxos

Screenshot Showcase

Uploaded by Pirate, October 22, 2009, KDE 4.3.2

ms_meme's nook

They like to chat about compiling

    that they do at night

They chew the fat about a code

    if they've got it right

Inconsequential things

    gals don't really care to know

Become essential things

    that nerds find so apropos

But that's a guy

    they're all the same

It's just a game

    they call it Nerd Talk Nerd Talk

They brag about the ups and downs

    of all their commands

The who the how the why

    the perfection that Tex demands

The stronger sex

    we need them for all our dough

But though we joke

    we wouldn't trade you for a Window

So guys stay

    and gab away

But hear us say

    that after Nerd Talk talk to us

signed
the forum nerdettes

Manually Create A Live USB Flash Stick

by JohnBoy

Note from the Editor: Recently, my wife asked me to install PCLinuxOS 2009.2 on her Acer Aspire One netbook to replace the crawling Windows XP that was originally installed on it, and to set up a dual boot with Windows XP on her Dell Vostro 1000 notebook. Since I was going to install on the Acer Aspire One first, I decided to make a Live USB. However, repeated attempts at using the Make Live USB utility to create a bootable Live USB, failed. Next, I tried to use Unetbootin. I tried the PCLinuxOS version, but it met with the same results as the Make Live USB utility. I tried the Windows version (from a computer with Windows XP on it), and still had no success. Finally, I tried the method below. At first, I was intimidated by the length of the process. But, as I went through it, step-by-step, I discovered that it was not only easy, but it also successfully created a bootable Live USB – the very first time through. – Paul Arnote, PCLinuxOS Magazine Chief Editor

Pre-requisites:

Working from an installed PCLOS ..... either updated to present or a 2009.1 install.
Flash stick minimum of 1GB but preferably larger

Aim:
To produce a flash stick that will run PCLOS 2009.1 (or later) from any PC which is capable of booting from USB

Synopsis:

1. Create an ext3 partition on a flash drive of 1GB size or greater. Give permission to everyone to R/W

2. Copy the OS files to the partition from an ISO or CD of 2009
3. Create a boot folder on the partition, and copy the grub folder from a running PCLOS 2009 into the boot folder
4. Edit the menu.lst file on the flash disk
5. Install Grub to the MBR of the flash drive

That's it!

Below is my suggestion of one method of achieving this. It is not the only method, nor is it exhaustive, but it should get you through the process if you are in any doubt.

Essentially this is the same method as described here http://pclosmag.com/html/Issues/200804/page01.html, with a few minor changes to the boot stanza and the use of the partition Label to boot, and also includes persistence.

•••••••••••

Process:

Step 1.

Use PCLOS Control Center - Local Disks - Manage Disk Partitions. Delete all existing partitions on the stick.

Create an ext3 partition of minimum 1 GB, but preferably more if persistence is needed.

If it is a 4GB flash disk, then make the partition the full size of the flash drive.

Set the permissions of the partition to allow everyone to read and write it.

Give the partition a Label ... let's say, 2009.

Format the partition.

Select Done from the bottom of the page.

Step 2.

Safely remove the drive and then reinsert it after a few seconds.

An icon should appear on the Desktop called 2009.

It should now mount to /media/2009 when clicked.

Right click on the Desktop icon, select Properties and from the Mounting tab, and untick "Access time Updates". Click OK.

Next, using a 2009 Live CD open it in Konqueror ...

or

If you have just the ISO of 2009, right click on it ... Actions - ISO9660 View. This will open the ISO in Konqueror. Click on the folder ISO9660 to display its contents.

Then, click on the 2009 Desktop icon so that it opens in Konqueror or other file manager.

Now from the Konq view of the CD or ISO, copy the isolinux directory and the livecd.sqfs file to the 2009 flash disk partition.

Step 3.

Next create a directory called boot on the 2009 flash disk.

Navigate to /boot in the running OS and copy the grub directory to the boot directory of the 2009 flash disk.

Your flash disk should now have

isolinux   folder
boot   folder   {and inside this folder should be
                        the grub folder}
livecd.sqfs file

Step 4.

Open the /boot/grub/menu.lst file on the flash disk for editing. Delete the existing boot stanzas, leaving the headings etc intact. Place a # in front of the gfxmenu line. (If you want a graphical boot, it can be done later.) Add the following boot stanzas:

```
Code:

title 2009 -- With Persistence
kernel (hd0,0)/isolinux/vmlinuz
vga=788 livecd=livecd fromusb
root=Label=2009 changes_dev=LABEL=2009
acpi=on fstab=rw,noauto
initrd (hd0,0)/isolinux/initrd.gz

title 2009 -- No Persistence
kernel (hd0,0)/isolinux/vmlinuz
vga=788 livecd=livecd fromusb
root=Label=2009  acpi=on
fstab=rw,noauto
initrd (hd0,0)/isolinux/initrd.gz
```

Now, for the terminal stuff.

Step 5.

Open a terminal and su to root. Type grub «enter». After some little time you get a grub prompt

grub>

To be sure we work on the correct drive and not interfere with any HDDs in the system, we ask Grub to find all instances of menu.lst file:

grub> find /boot/grub/menu.lst

You will get a response something like

(hd0,0)
(hd0,5)
(hd2,0)

grub>

This shows I have a menu.lst file in partitions 0 & 5 of hd0 (first HDD) and on the first partition of my third disk. So I reckon that the third disk is my flash drive but I want to confirm it, so:

grub> geometry (hd2)

This gives me:

drive 0x82: C/H/S = 500/255/63, The number of sectors = 8040448, /dev/sdc

Partition num: 0, Filesystem type is ext2fs, partition type 0x83

(Note: There may be other lines like the one above if there are other partitions on the device .... such as Partition num: 1, Filesystem type is ext2fs, partition type 0x83)

I can see from this that the third disk is /dev/sdc, and I can confirm that to be my flash drive by hovering the mouse over the icon on the desktop, which will give me the device node of the drive. In my case, it tells me it is /dev/sdc1. So now I am sure I can go ahead and install grub to the MBR of that drive.

grub> root (hd2,0) «enter»   -> this specifies the location of the files that Grub is to use when booting

grub» setup (hd2)   «enter»   -> this installs Grub to the MBR of (hd2) which is /dev/sdc

When this completes and reports success, all is finished.

To change your flash stick to 2009.2 or .3 or later all that is required is to replace the isolinux folder and the livecd.sqfs file with the newer version.

Note: Without the **fromusb** code in the boot line the stick does not shut down properly. I do not know if its presence will affect correct booting in other situations. I have only a limited amount of hardware to try it on. I would appreciate it if others could check this out on their hardware. If the stick has a problem booting in any situation please first try removing the "fromusb" code from the boot line and then reboot.

Finally there is additional information below for those wishing to have more than one choice of OS on their flash drive. This explains how to configure the files to allow different versions of PCLOS to be bootable from the same flash stick.

For those of us who are using USB flash sticks to carry our favourite PCLOS releases around to boot whatever PC we come across, I thought to add a small change here which might make things easier.

The above method called for the various releases to be in their own folder and to change the location of the release files when you wished to boot a different OS.

Instead of that method, which is awkward, try this ....

Let us imagine that we have the OS files for

* Minime 2009.1
* PCLOS 2009.2
* ZenMini 2009.1

all in their own folders on the USB stick.

Rename each of the pairs of files from isolinux & livecd.sqfs to something reflecting their release. So in the above example they might become

kdemini1 & kdemini1.sqfs
pclos2 & pclos2.sqfs
zenmini1 & zenmini1.sqfs

Now the files can be moved to the root of the stick and they won't interfere with each other as they all have different names, and are named in pairs representing their release.

Because the files are named differently we will need a different boot stanza for each release we have on the stick. So the menu.lst file might look something like this:

```
Code:

title  Minime 2009
kernel (hd0,0)/kdemini/vmlinuz
livecd=kdemini fromusb acpi=on
fstab=rw,noauto vga=791
initrd (hd0,0)/kdemini/initrd.gz
```

```
title   Zen Mini Gnome 2009
kernel (hd0,0)/zenmini/vmlinuz
livecd=zenmini fromusb acpi=on
fstab=rw,noauto vga=791
initrd (hd0,0)/zenmini/initrd.gz


title  PCLOS 2009.2
kernel (hd0,0)/pclos2/vmlinuz
livecd=pclos2 fromusb acpi=on
fstab=rw,noauto vga=791
initrd (hd0,0)/pclos2/initrd.gz
```

You can of course keep adding releases as long as you have the space for them on your disk. If your files are on other than the first partition just change the (hd0,0) to reflect that ... to hd(0,1) if they are on the second partition etc.

NB   If you wish to use an older version of PCLOS -- I tried Minime2008 -- I advise not renaming the files. The change of name by the livecd= boot code was not implemented then it seems.

Have fun!

Wallpaper Sites

Have You Seen This?

There are always questions in the forum about the wallpapers that are used (everyone's screenshots are so awesome!!!!) I'm sure there are tons of sites you can visit, but these seem to come up again and again......

http://www.digitalblasphemy.com/dbhome.shtml
Digital Blasphemy has all different categories of wallpapers, from quiet to really "out-there".

http://www.vladstudio.com/home/
VladStudio has some really different stuff.

http://www.desktopnexus.com/
Desktop Nexus has another assortment

http://getawallpaper.com/

http://www.3datadesign.com/gallery/eng/index.html

http://www.deviantart.com/

http://interfacelift.com/

http://mashable.com/2008/03/31/13-awesome-resources-for-hdr-wallpapers/
Some of the sites in the list are already in this article

http://linfx.com/gallery/index.php
This is the PCLinuxOS site that many of our users post their wallpapers on. There are also some logo files, Compiz skydome pics, etc. It's a great site.

http://www.crosscards.com/
This is a Christian site – but the wallpapers are very pretty. Many of them have calendars on them and some quote Bible passages.

http://www.graffitiwallpaper.com/index.php
Archie shared this one earlier and there are some nice ones there as well.

http://photography.nationalgeographic.com/photography/photo-of-the-day
National Geographic posts a "Photo of the Day" which can be downloaded.... They have been doing it for several years, so they have archives which can be searched for just the right pic!

http://www.socwall.com/browse/index.php
Social Wallpaper has many free, high quality images for use as wallpaper.

http://www.picturecorrect.com/freewallpaper.htm
While not the most extensive collection of wallpaper, they have a collection of very high quality images for free use as wallpaper.

http://www.linuxwallpapers.org/linux-wallpapers.htm
LinuxWallpapers has a large collection of wallpapers, especially geared for use by users of Linux.

http://www.caedes.net/Zephir.cgi?lib=Caedes::Gallery&gallery=abstract
Caedes Desktop Wallpaper has a very large collection of free wallpapers for your computer. Especially nice is their collection of abstract wallpapers (linked to here). But when there, be sure to browse their entire (massive) collection.

http://www.webdesignerdepot.com/2008/11/40-wallpapers-loaded-with-color/
The Web Deisgner's Depot has a nice collection of original wallpapers, free for your download.

http://stockwallpapers.blogspot.com/
High Definition Stock Wallpapers offers some really nice wallpapers for your desktop.

http://www.freeewallpapers.com/index.html

http://www.hongkiat.com/blog/category/freebies-online/free-wallpapers/
If full color spectrum rainbows or auroras are your thing, Hongkiat has a wide assortment of free wallpapers. They also have categories for wildlife, Halloween, Christmas, Marvel Comic Book heroes, and dual monitor setups, just to name a few.

http://antwrp.gsfc.nasa.gov/apod/archivepix.html
If astronomical pictures (the subject, not necessarily the size) are something that interests you, you will like this site. You may have to do some minor editing, but you can get some stunning wallpapers from here.

http://hubblesite.org/gallery/
The Hubble Telescope site has a gallery of pictures and a section in the gallery with wallpapers that can be downloaded in many different sizes. (I was even advised of the best size for my desktop!)

http://www.startrekdesktopwallpaper.com/startreknewwallpaper.shtml
If you are a Trekker, you can make the journey through your computer universe in style, with a wide variety of wallpapers featuring your favorite characters or ships from the Star Trek universe.

http://www.wallpapersuggest.com/
This site literally has a collection of wallpapers that goes into the thousands. A very wide range of interests are featured.

Don't forget http://www.kde-look.org/ , http://www.gnome-look.org/ and http://www.xfce-look.org/

Certainly, there are many, many more wallpaper sites available. But this should get you started (with rocket propulsion) towards finding outstanding wallpaper for your PCLinuxOS desktop. If you haven't visited some of these, have fun, and enjoy hours of delightful wallpaper searching!!!

Also, if you have a favorite wallpaper site that does not appear in the list above, please send it to the PCLinuxOS Magazine's Chief Editor, Paul Arnote. The easiest way is to send a private message on the PCLinuxOS Fourm to parnote. We'll include a link to your favorite wallpaper site in a future issue of the PCLinuxOS Magazine.

This list was compiled by Meemaw and Paul Arnote.

Flashback: Simple Backups Using Grsync

Original article by Iain Jackson (iainrj)
PCLinuxOS Magazine, August 2007
Updated by Paul Arnote (parnote)

This article is a simple tutorial on using Grsync as a backup utility. There have been a lot of threads on the main forum from users asking for advice on backup and I have found Grsync to be the simplest and quickest method around.

Grysnc is a GUI-front-end for the command line application rsync. Rsync has many powerful applications, not least for synchronizing data to web servers. In fact rsync scripts are used by Texstar for synchronising the repositories as new applications are added.

This short tutorial will show you how you can use Grsync to back up your /home directory to either another hard drive, partition or removable drive.

Your /home directory is where all your e-mails and bookmarks are stored in hidden folders. To view them, open your /home directory or partition in Konqueror and click on View » Show hidden files. Depending upon which applications you are using, you might see a .mozilla folder containing your Firefox bookmarks and extensions, a .thunderbird folder containing your e-mails, and a .kde folder containing your desktop settings and settings from KDE applications such as kmail.

First, you need to install Grsync from Synaptic. Once installed, you will see the application under KMenu » Archiving » Backup. It might be a good idea to create a desktop icon for easy access. To do so,

right-click the icon in the k-menu and then choose Add Item to Desktop.

Second, if you want to backup to a removable medium such as a USB flash drive or external USB drive, insert it now. When PCLOS detects the drive, click on Open in New Window to mount the drive.
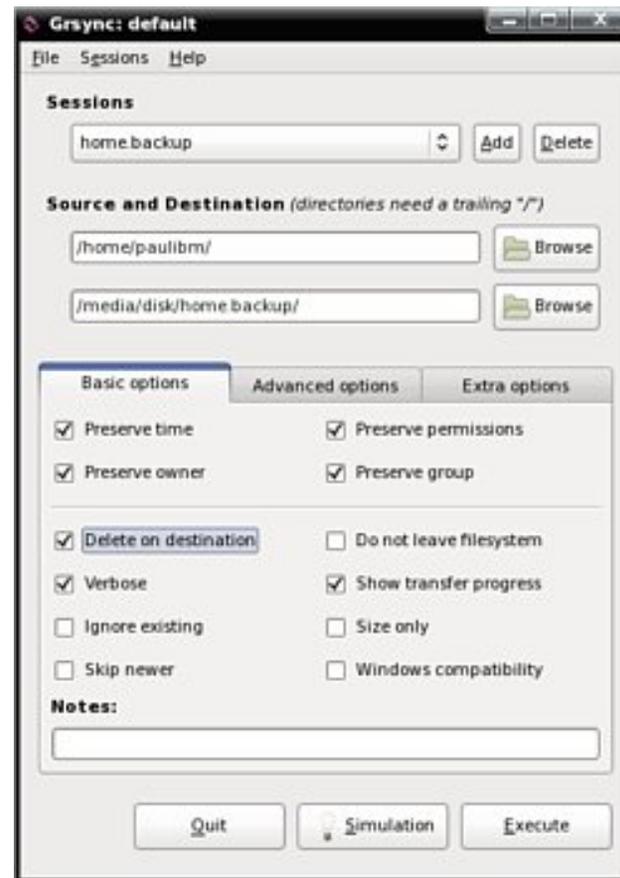
It is a good idea to create a new directory on your removable disk to store the backup files. In your new window, create a new folder. As an example, home.backup is the directory name I use.

Use the panel icon to go back to Grsync once your media is mounted.

Third, now it's time to create a backup session. When you launch Grsync you will see the screen depicted on the left.

Click on Add in the Sessions section, and enter a session name. This is just so that once configured, you can quickly run the same backup routine by choosing the session. Type something like "Home Backup" and then click OK.

You now need to add your source and destination directories. There are 2 blank boxes in the middle of the Grsync window. The top box is the source - this is where your files currently are. The bottom box is the destination - where you want your files backed up to.

Click on Browse next to the source box. This should take you straight to your home directory. Just click on Open to select it.

Click on Browse next to the destination box. Your removable drive should have appeared as a link on the left hand side of the file dialog for easy access. Double-click it, then double-click on your backup folder before choosing Open.

Fourth, it's time to change the basic options. It's probably a good idea to select Preserve Owner, Groups and Permissions. This way, if you need to reinstate from the backup, you won't get into any ownership problems.

Select Delete on Destination to ensure that you don't end up with any old files in your backup that don't exist in your home directory.

Fifth, we can also set some advanced options. Switch to the "Advanced Options" tab. Check the Copy symlinks as symlinks box. This means that if there are any links to files, only the link, rather than the file itself, is copied over.

Additional options are useful if you want to add other rsync or console commands. The one I use is the exclude command. The example above excludes the .thumbnails directory from the backup.

To find out all available rsync commands, type man rsync into a terminal. You can get an abbreviated, shorter list by simply typing rsync (by itself) at the command line.

Sixth, do a "dry run." Click on Simulation and you'll get a quick run through of what is going to be copied over.
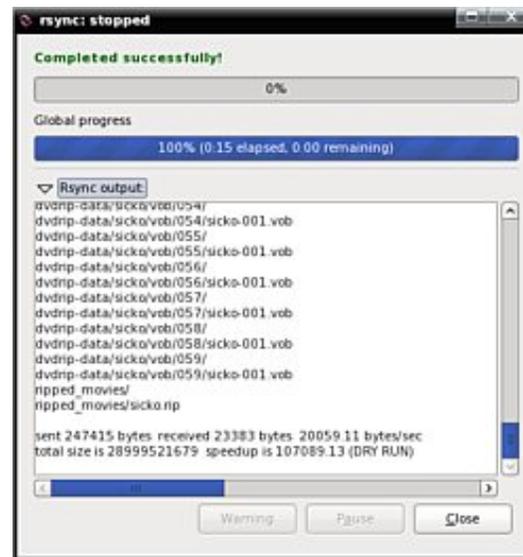
Seventh, when you are satisfied all the correct files are being either deleted or copied, click on Execute. The first time you run the sync it may take a while as it has to transfer all your data. The exact time will depend upon the amount of data you are copying and the speed of your computer. It might take 5-10 minutes to transfer your /home the first time out, longer if you have a lot of large files such as audio or movie files.

Subsequent syncs will be much quicker, perhaps 30 seconds, as it only sends new or updated files for backup.

Grsync makes using rsync easy but doesn't give you the full functionality. If you want to have a go at creating scheduled backups, you can write rsync scripts and use cron or Kcron to schedule the tasks. Here's a good website that takes you through that process: http://www.mikerubel.org/computers/rsync_snapshots/#Rsync. They even have a couple of sample rsync scripts that you can use with cron or Kcron.

For me, Grsync works a treat. About once a week, I switch on my external hard drive, and run my 2 Grsync sessions - one for /home and one for my data partition. The whole process takes about five minutes and I am safe in the knowledge if I need a clean reinstall of PCLOS I have all my application settings, e-mails, bookmarks and - more importantly - my music and photos backed up.

Screenshot Showcase

Uploaded by Linuxera, October 22, 2009, KDE 3.5.10

Disclaimer

1. All the contents of the **NEW PCLinuxOS Magazine** are only for general information and/or use. Such contents do not constitute advice and should not be relied upon in making (or refraining from making) any decision. Any specific advice or replies to queries in any part of the magazine is/are the person opinion of such experts/consultants/persons and are not subscribed to by the **NEW PCLinuxOS Magazine**.

2. The information in the **NEW PCLinuxOS Magazine** is provided on an "AS IS" basis, and all warranties, expressed or implied of any kind, regarding any matter pertaining to any information, advice or replies are disclaimed and excluded.

3. The **NEW PCLinuxOS Magazine** and its associates shall not be liable, at any time, for damages (including, but not limited to, without limitation, damages of any kind) arising in contract, rot or otherwise, from the use of or inability to use the magazine, or any of its contents, or from any action taken (or refrained from being taken) as a result of using the magazine or any such contents or for any failure of performance, error, omission, interruption, deletion, defect, delay in operation or transmission, computer virus, communications line failure, theft or destruction or unauthorized access to, alteration of, or use of information contained on the magazine.

4. No representations, warranties or guarantees whatsoever are made as to the accuracy, adequacy, reliability, completeness, suitability, or applicability of the information to a particular situation.

5. Certain links on the magazine lead to resources located on servers maintained by third parties over whom the **NEW PCLinuxOS Magazine** has no control or connection, business or otherwise. These sites are external to the **NEW PCLinuxOS Magazine** and by visiting these, you are doing so of your own accord and assume all responsibility and liability for such action.

Material Submitted by Users

A majority of sections in the magazine contain materials submitted by users. The **NEW PCLinuxOS Magazine** accepts no responsibility for the content, accuracy, conformity to applicable laws of such material.

Entire Agreement

These terms constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes and replaces all prior or contemporaneous understandings or agreements, written or oral, regarding such subject matter.