Choices

mlowman

November 2007

1 Happiness

Linux often has a lot of ways to do something. In Windows, there is a strict line between the programs packeged with the OS and those not: For text editing, you have notepad. It's somewhat limited, but you have to expend effort to get free alternatives. (Not much, though...). Competition makes things better, and Windows only needs to package one text editor and everyone will use it. You can guarantee that a fresh install will have only Notepad.
Not so in Linux. The free alternatives ARE the ones packaged with the OS, and there are lots. Everyone has a shot at making an app that becomes part of the OS package. Therefore, there is nearly always more than one way to do something. If someone wants a small change (or a huge one) in the waay an app works, they write their own from scratch or fork the project. Here I'll make an argument for making certain choices; remember, you can make your own. I recomend trying things. But these are some of the choices, and the ones I like.

2 Text Editors

There are many kinds, but I only use command line myself. Easier, faster, and doesn't involve using the mouse.

2.1 Vim

In the beginning, there was vi. Designed for an earlier age, today's user will find a slightly too high learning curve: traditional cursor movement with

hjkl, not a lot of feedback to describe the different modes... In fact, not a lot of feedback at all. Remember, screens were TINY. Use real estate efficiently. But then along came vim, vi improved. Syntax highlighting for everything under the sun (and you can make your own if you design a new language or something), batch editing (applying the same change to rows at a time), command integration (add the output of a command with the touch of a few keys), modifiable themes, easy macro extension (I write this in vim, and I have a macro to compile this tex document at a touch and then view it), tabbed windows (remember, this is in CLI!)... The list goes on and on. Once you know the first few commands, it's really easy to use. You don't have to touch all that other stuff, but if you want to it's there. And there's graphics, too... gvim. But I don't use that.

2.2 EMACS

Can't speak for it, haven't really used it. It is huge, compared to vim. It has an amazing amount of stuff built in, though: a complete LISP scripting interface, tetris, and a lot more. Plus some of the stuff in vim, too. But vim is easier to use, and EMACS requires a lot of customization for full usage, while vim works out-of-the-box. By the way, there are traditionally huge arguments between EMACS proponents and Vi adherents. So that's what that's all about.

2.3 Nano

A free clone of pico. Pretty much as simple as you get. All the commands are listed by default at the bottom of the screen. But it can do so much less than vim, the only real excuse is to use it for editing files to set up a download of vim if vim isn't installed by default (a travesty!), or for a very small system (nano does have a distince size advantage).

2.4 ed

For the really old-school. Not really a choice for a modern system. Unless you like pain, that is.

3 Desktopish Stuff

First, components of an X desktop setup. You've got a window manager, which deals with window borders, moving windows, tiling, etc. There are lots of these. And you've got a desktop environment. Not necessary, it is a set of programs that do things. They run the gamut from graphical system conf utils (say, a volume control, a keyboard repeat rate changer, or a printer configurator) to utilities (a text editor, a file manager, etc.) and desktop programs (like a program to setup a background, a small bar to attach applets to, virtual desktops, or a root menu). Some of the last part can be in a window manager, too. There are only three desktop environments: Gnome, KDE, and XFCE. One other thing: each has a set of libraries for gui elements/graphics. This keeps everything consistent.

3.1 Desktop Environments

3.1.1 KDE

This came first. It looks a lot like Windows by default (in terms of desktop icons, the main bar, the "start" menu. One of the pre-installed themes is called Redmond: it's a traditional Windows 95/98 look. It takes a lot of system resources in my experience, although some dispute that.

3.1.2 Gnome

The KDE libs, Qt, were originally non-free. As in, they cost nothing, but were not open source. The GNU Project took swift action, and bagan development of Gnome. Nowadays Qt is free, but Gnome's still around. It looks a bit more like a Mac by default, and has lower resource consumption than KDE.

3.1.3 XFCE

My favorite, the newcomer on the block. A better UI design than any of the others, in my opinion. It's only recently become a viable alternative to the others. Destop icons (which I don't like or use) are new. Of course, I think that the minimize windows to icons is rather cool. Very low resource usage, but good functionality, and good interface.

3.2 Window Managers

Every one of the above had an underlying WM. Of course, most use one developed by the same project. But they can use any. And there are lots of others.

3.2.1 Fluxbox

Probably my favorite of the separate WMs, this is a fork of Blackbox. Very minimalistic: the only interaction is really the right click menu, with all programs, accessible from anywhere on the desktop. Obviously low resource. Easy, and changing the menu around is a simple matter of editing a conf file. In fact, all configuration is simple.

3.2.2 Enlightenment

Version 17, that is. Very different from 16. I don't know if it's beta or not, but it's as stable as a full release. Basically, this is a WM with more eye candy than a full desktop environment. Very fast, a proper theme looks positively beautiful, low resources. Until it's released as stable, this will require more work to install than any other: compile it yourself, add alternative sources to your download lists, or repackage it yourself.

3.2.3 twm

The first (or near enough, I can never remember) WM. As simplistic as you can get: a sickly green background with windows outlined by no more than some white and black lines. Usually, a few xterms and xeyes are run upon start. The only reason I include this is because it is the failover for gdm: if the user's chosen WM can't load for whatever reason, then this will. Basically because you're guaranteed to have it on your system. Need I say it is incredibly small and uses not much more than naught?

3.2.4 Beryl or Compiz or Compiz Fusion

A WM that uses Compositing extensions to X. Not all graphics cards necessarily support it. Certainly they will require good drivers, like the non-free ones for nvidia and ati. Once you get your X set up right, you can use this. Compiz came first, Beryl forked from Compiz because Compiz used

Gconf, then they kissed and made up. Compiz Fusion is the currently developed one. Anyway, the effects are stunning. Think Vista+++, OS X++. Full real transparency, wobbly windows, fire effects, awesome water ripple effects... Very cool. The only problem being the configuration and that it crashes my laptop about every two days or so if I use it. But still quite amazing.

3.2.5 xfwm4

XFCE4's wm.

3.2.6 Others

A lot. Some are experimental, some aren't. Some take the side of only mouse interaction, some use only keyboard. Some are hard to use at first, others are point-n-click simplicity. Try some out if you want. And don't EVER criticize something you haven't used. I mean it.

4 Distros

These are specific to Linux. Basically, every distro includes your set of basic apps. The stuff that GNU made. Then each one has a different way of organizing configuration files, a few distro-specific utilities, and a different way of downloading packages. Of course, there are only a few package systems out there. Oh, there's probably a custom installer, too.

4.1 Debian

One of the oldest distros still actively developed and heavily used, it's pretty average all around. Not a bad first distro by any means. It comes in three flavors: stable, testing, and unstable. Stable is guaranteed to work. Everything's tested a thousand times over. Of course, that also means that not much gets added. Definitely not the latest and greatest, by any means. Testing is what you should use. Average of the two extremes of stable and unstable. It can be annoying to find the isos on the debian site, though: they're very well hidden. Unstable is always code-named sid. It has the newest stuff, but things tend to break. There's no guarantee that stuff won't

break in testing either, but it generally doesn't happen. Lots of distros are based upon Debian. Such as...

4.1.1 Ubuntu

I think it's about the most popular distro right now, but it's a newcomer to the field. With an Ubuntu install, no technical know-how is required. Your devices will "just work." Linux in general will "just work." But this is the coward's way out. Fast and easy, yes. But you won't get any real understanding of how your system works. And don't even think of putting this on a server. Not what it was designed for, guys...I do like the release names. And they guarantee release support for quite some time.

4.1.2 KNOPPIX

The king of livecds. Runs KDE (as well as fluxbox by kernel option at boot), runs it well. A most useful tool in your arsenal, LiveCDs (and live usb) are to today's systems what rescue floppies were to the past. Don't leave home without one. And make sure you have one in your home, too. And the isos to make more if you need to.

DSL KNOPPIX in turn inspired D–n Small Linux, which is extremely tiny. Made to fit on a flash drive in the bad old days of tiny drives, it will never go over 64 MB. And it's amazing the sheer variety of stuff that fits! I don't like it too much, but it's fun to try out. Try to get it on a USB without burning the CD or using a VMware image. It's a good exercise, and when you succeed you'll understand more about the boot process.

4.2 Gentoo

Definitely my favorite, it's what we use here in the lab. A first-time install may be off-putting to the novice, but they have a tutorial (on the site) that will walk you all the way through it. I think that there's a graphical intaller now, too. Their minimal LiveCD is very nice, and I haven't tried the other. A semi-unique package system is in place, called portage: it builds everything from source, and is a much-improved version of the FreeBSD ports system. Gentoo is also completely customizable since you can set compile-time options. But it's better than that...You have to try it. Only

disadvantage is time; everything is compiled from source, so an install of KDE or GNOME can take ten hours or more on a slow system. But not ten hours you have to sit there, so do it overnight.

4.3 Slackware

Older than Debian, and also still active. It shows its age, but therefore has more support for older systems. Not many distros still have support for floppy install. We used this in the lab before Debian (and before I ever got here). I know that at that time there wasn't a package manager, but I can't tell you the current situation.

4.4 Red Hat

Commercial Linux. Red Hat has a free system, but sells support contracts. They are used by big-name companies, since support is extremely necessary to the big business. And for good reason. They had the idea of two different distros: Fedora Core is community-developed, while Red Hat Enterprise Linux costs money. Fedora Core is a testing area for the newest and best, and if it proves to be stable they move it over to RHEL.

4.5 SUSE

Made by Novell, whose former main product was Netware. But that's not used too much any more. Same model as Red Hat, with commercial and community-developed.

5 Other UNIX

There are a bunch of other UNIX versions out there, most from earlier eras. Some are still current, too. For instance, Solaris is common (not to mention used in our lab), *BSD are known for stability/security/compatibility/embeddability, depending upon which one you're talking about, and there's also Tru64, AIX, and lots of others I can't remember. OS X also counts. Some (like FreeBSD) even have binary compatibility with Linux, but general setup varies, and of course the kernel is different.