

LIQUID LEMUR MAGAZINE

Vol. 1 No. 1

Oct-Dec 2011

Inaugural Edition!

In This Edition -

An Interview with the Lead Dev!

Theming Window Maker Part 1

Adopting Open Source Software

Screenshots and MORE!



Liquid Lemur Linux

Contents

Contents

A Word From The Editor

by busprof

An Interview With DragonPalemoon: Liquid Lemur Lead Dev

by busprof with DragonPalemoon

eLinks: A Text-Only Browsing Alternative

by busprof

Adopting Open Source Software: Reasons and Best Practices

by Max Heifei Li

Theming Window Maker Part 1: Styles

by busprof

Screenshots!

Release Ramblings

by DragonPalemoon

Dockapp Roundup: wmpinboard

by busrprof

Disclaimer

- 3 The information in Liquid Lemur Magazine is provided on an "as-is" basis. All warranties, whether expressed or implied, of any kind, regarding any matter, pertaining to any information, advice or replies are disclaimed and excluded.
- 4 The contents of Liquid Lemur Magazine are intened only for general information and use. As such, the contents do not constitute advice and should not be relied upon in making (or not making) any decision. Any specific advice or replies to questions in any part of the magazine are the opinion of such persons and are not subscribed to by Liquid Lemur Magazine.
- 9 Liquid Lemur Magazine and its associates shall not be liable, at any time, for damages of any kind arising in contract, tort or otherwise, from the use of or inability to use the magazine, or any of its contents, or from any action taken (or not taken) as a result of using the magazine or any of its 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 in the magazine.
- 12 No representations, warranties or guarantees whatsoever are made as to the accuracy, adequacy, reliability, completeness, suitability, or applicability of the information in the magazine to a particular situation. All trademarks are the property of their respective owners.
- 14 A majority of the content of the magazine is comprised of material submitted by users.
- 19 Liquid Lemur Magazine accepts no responsibility for the content, accuracy, conformity to applicable laws of such material.
- 23 Entire Agreement
- 25 These terms constitute the entire agreement between the parties with respect to the subject matter hereof and supercedes and replaces all prior or contemporaneous understandings or agreements, written or oral, regarding the subject matter.

A Word From The Editor

A Word From The Editor

by busprof

Beginnings are hard.

Working on this first issue of Liquid Lemur Magazine over the past couple of months has been relatively easy. Sitting down and putting together this article - which I fully realize will be the first thing most readers will see - has in some ways been the most difficult part of the entire process. What makes it hard is that I want to start on the right foot. I want to make this community-built effort as good as it can be. I want to provide both a good introduction to this first issue and an invitation for future participation by as many Lemur community members as possible. Most of all, I want Liquid Lemur Magazine to reflect positively on the distribution itself and the community of users, contributors and developers that it represents.

Thinking about how to start this introduction to the first edition made me consider new beginnings generally, and DragonPaleymoon's decision to launch Liquid Lemur Linux specifically. We have a great interview with DragonPaleymoon in this issue, by the way, and he talks about the genesis of Liquid Lemur there, but what he did not say, and what has been brought home to me as I considered how to start things off here, is that beginnings require courage.

Consider - It took courage for DragonPaleymoon to release Liquid Lemur publicly. Putting together a customized Linux system for your own use - especially

if you have the skills, abilities and experience of a DragonPaleymoon - is not exceptionally difficult. If you make mistakes, they are private mistakes and there is no one looking on to see them. If you pick an application for inclusion in your system and it does not work out the way you expected, then you just replace it with something that better meets your needs - no one else will ever know that your first choice was poor. Putting something - anything - out for public consumption, however, leaves you open to criticism, to second-guessing, and even to embarrassment. Add to that - if you have any sense of responsibility at all - the knowledge that the mistakes you make in putting out a "product" might actually cause some harm to others (if only in time wasted) and it quickly becomes as much a matter of courage as it is of conviction to make something available for others to use.

Thankfully, DragonPaleymoon had not only the conviction that his vision of a good Linux distribution would be valued by others, but also the courage to make it available to others. I am proud to be associated with Liquid Lemur Linux. I am pleased with this first issue of Liquid Lemur Magazine. I hope that you will enjoy reading it, and that you will have the courage to share your skills, abilities and experience with others by submitting an article, artwork, a screenshot or a story to a future issue. One of the things I have enjoyed most as a Linux user over the years is the community involvement Linux allows. I believe this to be a real strength of the open source approach. Without an involved community of people with the courage to share, there would be no Linux.

Let's all work together to make Liquid Lemur Linux as good as it can be. That only comes with sharing - and sharing takes courage.

With best wishes,

busprof



Liquid Lemur Linux and the **Liquid Lemur Linux** logo are the trademark of DragonPaleymoon.

Liquid Lemur Magazine is a quarterly publication and is published by and primarily for members of the Liquid Lemur Linux community.

The Magazine contains material related to Liquid Lemur Linux.

Volunteers from the Liquid Lemur Linux community serve as members of the Magazine staff.

The following volunteers made this issue of Liquid Lemur Magazine possible:

Editor: Bill Nance (busprof)

Layout: Bill Nance, Ed Snyder (DragonPaleymoon)

Contributors:

busprof

DragonPaleymoon

Max Heifei Li

Liquid Lemur Magazine is released under the Creative Commons Attribution Non-Commercial Share-Alike 3.0 Unported license. Some rights are reserved.

Copyright © 2011. Visit us at <http://www.liquidlemur.org>

An Interview with DragonPalemoon: Liquid Lemur Lead Dev

An Interview with DragonPalemoon: Liquid Lemur Lead Dev by busprof



The camera catches Lead Developer DragonPalemoon answering Forum help requests

LLM: DragonPalemoon - interesting screen name - tell us how that came about, and share a bit about your background.

Well, I am part (one-quarter) Cherokee Indian. My great-grandmother gave me the "Palemoon" name a very long time ago, when I was a kid. She was full-blooded Cherokee and even though my skin tans quick and is darker than most, she still considered me pale...compared to her. She also noticed I was fascinated with the moon. So, at first, the palemoon label was just to tease me. But later, it stuck.

As I became a teenager, she told me it was decided that "Palemoon" would be my Indian name. I argued with her about it, because I didn't think it sounded like an Indian name. I mean, what happened to "Running

Bear" or "Sitting Bull" and the likes? She told me I watched too many movies (*laughs*). She also told me that since I was not full Cherokee, I did not qualify for a "full" Indian name. Later in life, before she passed away, she told me that part was a joke. She told me that my body may not be "full" Cherokee, but my spirit was. That made sense...considering my love of nature, animals, the land, etc.

Now the "Dragon" bit was added later. Growing up, I was always interested in dragons. I don't know why, but I still am. Anyway, as a young adult, I learned that I was born in the Year of the Dragon (according to the Chinese calendar). So, again, it all made sense and seemed to fit.

LLM: And your background?

I've been using computers since 1982. My first computer was a Timex-Sinclair 1000, that I had to assemble. I can remember paying \$300 for a 16k (yes I said K) memory upgrade. I wrote my first program on that thing. It was a Zork type game. I later graduated to the Commodore Vic-20 with the killer cassette tape drive. (I really miss those...not!) I wrote a Risk clone for that and actually sold it to a few people while I was in the military. I soon was the proud owner of a Commodore 64 and that head

banging 1541 floppy drive. I wrote several apps and games for that puppy. I got my first taste of Assembly programming on the C64...it was a bitter taste. I went through several C64's over the years. Including a C64c and the overheating SX-64 (portable). I ran a BBS or two from the C64.

In late 1985, I got my first REAL computer... The Amiga (I won't say Commodore Amiga, because Commodore screwed us Amigans over in the end.). The OS was sleek and fast. The multimedia and



The computers of PaleMoon: Timex-Sinclair 1000 (top left); Commodore vic-20 (top right); Commodore 64 (bottom left); Amiga 3000 (bottom right)

games were awesome (back then). It was also my first Unix based OS. Yes, Amiga OS is (was) Unix based...and Amiga OS 3.1 is the best freakin' OS ever conceived IMHO. I went through just about every model of the Amiga. The A3000 Tower was my favorite. I miss my "Miggy!" It was way ahead of its time back then. We all tried to get Carl to re-create his wonderful OS to run on x86 hardware...instead, he came up with REBOL. Thanks Carl! Although, there are attempts to bring an Amiga like OS to the x86 platform (AROS, icAROS, and Anubis) which I follow closely, and help out when I can. A bit later I ended up on Windows 3.11. Coming to Windows 3.x from Amiga OS 3.1 was like going from a Porsche to a Ford Pinto.

I suffered through the various incarnations of Windows for a bit...then I discovered Linux. And what a discovery it was! My first distro was Caldera, and it was a work of art. I loved it! Even had the bumper sticker! But as luck would have it, I was forced into the MAC world. I've always considered the MAC a sister to the Amiga. Anyway, I took a job that required me to use MACs. Now I could either spend my life at this company, or buy a MAC and do most of the work at home...with family. I chose the latter. I ended up with a dual 2.0 G5. It was a nice machine. The company later went under, but I kept at the MAC until it died. The repair costs were too much for me (about the price of a killer PC). I could have fixed it myself, but Apple wouldn't sell me the parts! So I sold the G5 as is and thought I would just get an Intel MAC when they came out. Wrong! The G5 equivalent Intel MAC was the same price as the old G5! Why? I could get

the same hardware in a PC for a third of the price!
Thanks Apple!

So, I decided it was time to jump ship on anything Apple or Microsoft and went searching for a good Linux distro. I tried several, but ended up on openSUSE. Stuck with it for awhile, but when they started forcing KDE4 down my throat AND teamed up with Microsoft, I decided I had enough. I tried Ubuntu and thought, "Eeewww!" Searching for something MACish I came across Dreamlinux...and the rest is history as they say!

I still tested other distros now and then, but Dreamlinux in some form was my main distro. It was in the form of DreamStep...a distro I created from Dreamlinux and the Window Maker desktop. I also had a Dreamlinux KDE4 MOD in the works. I can hear the roar already...you're wondering why I would create a KDE4 distro if I left openSUSE because of KDE4. Good question. But, KDE4 now is actually usable...back then it was not. They were forcing us to



DreamStep Linux - The spiritual forerunner of Liquid Lemur Linux

test an incomplete desktop by making it the default desktop within the distro...which caused me to leave.

I've been helping out LxH (*editor's note: LxH is 'Linux Hardcore,' a Linux forum now offline*) and DLF (*note: Dream Linux Forum*) behind the scenes, not to mention the distro building and helping in the forums from time to time. So, I guess they felt I was worthy enough to wear the LxH badge with my new Geek hat. LxH is no more...

I eventually became part of the Dreamlinux dev team and almost took over. Things went South quick. So, here we are... I am no longer part of Dreamlinux. I left the team and the distro for good. As a result of that,



HEKA OS: The 'personal' distro that went public

my DreamStep distro was discontinued.

Still wanting a Window Maker distro, I created a "personal" distro called HEKA OS. HEKA OS was never intended to be a "public" distro. It was started as "my" distro, with the everyday apps that I used, pre-installed. Some wonderful friends of mine were nice enough to help make HEKA OS public. So, I created the website and made it official. However, the recent events with Dreamlinux had left a bitter taste in my distro-building mouth and I was fast losing interest in HEKA OS. And the complaints, from Window Maker fans, about HEKA OS being DVD size didn't help. Apparently, it is taboo for a Window Maker distro to be DVD size...or so I'm told. Had those people actually

read the "About HEKA OS" page they would have learned why it was DVD size! Anyways, I pulled the plug on HEKA OS.

Not being able to shake the distro-building bug, and longing for a Dreamlinux like desktop again, I created a new distro... Liquid Lemur Linux.

LLM: Some of our readers are younger people who might have an ambition to one day develop a distribution on their own. What is the right way to move in that direction? What skills do they need as programmers? As project managers?

I would say the first step to building your own distro would be to decide what you are trying to accomplish, and then see if there is already a distro that fits your needs. If so, then volunteer to help that distro instead

...I think project management should be a priority...I have a huge dry erase whiteboard...where I track various bits of Lemur development!

of starting a new one. If not, then start your own. You need to figure out what base would be best (i.e. Debian, Redhat, etc.) for your project and what package manager you want to use. Then there is deciding on the desktop environment, default apps, etc.

Sometimes, you may like a certain distro, but think it would be better with a few changes. For this, a simple re-spin of that distro would be better than starting a new one. Most major distros have community based re-spins of some sort.

To do a simple re-spin, or even create a new distro, one does not really need a lot of skills as a programmer. However, if you plan to make a go at a full-blown custom distro, learning bash scripting and python are critical. I came from Dreamlinux, where we used Ruby and LUA a lot. After starting Liquid Lemur, I've had to take up learning python, as some of the custom apps are coded in it. Also, knowing about forums and websites is a huge plus if you plan to go it alone.

Personally, I think project management should be a priority, even for one-man shows (that's one person doing all the work). For me, I have a huge dry erase whiteboard on the wall where I track various bits of Lemur development. If you are a one-man show, then I think managing yourself is more important than managing the distro. Because if you properly manage yourself, the distro will just fall into place. Project/Self management was something I did well in the beginning of Lemur. However, lately I've been distracted with real life events and I think Lemur is feeling the effects.

LLM: You created Liquid Lemur Linux. Why did you do this distro? What are your motivations?

I was a developer on the Dreamlinux distro for a bit. I even created a Window Maker version of Dreamlinux called, DreamStep (Blending DREAMlinux with a nextSTEP look). I really loved that distro, but wanted to see it grow. Things started slowing down for Dreamlinux and many of the developer and staff left. Lack of communication was a huge problem on the Dreamlinux side. The Lead Dev of Dreamlinux was dealing with health issues and was looking to either pull the plug on Dreamlinux or get someone to take over. I agreed to take over, but only if I could do what I felt was best for the project. I was given the green light and started to revive Dreamlinux. Just when I was getting into it, the Lead Dev returned and decided to do things his way and stopped communicating with me. So, I left Dreamlinux for good. After a brief period, I started feeling the need for a Dreamlinux style distro. I couldn't find what I was looking for, so I started Liquid Lemur. Initially, it was to be a Window Maker only distro. However, I (as childish as it may seem) felt the need to prove a point to the Dreamlinux Dev. So, I went with Xfce as the main desktop for Lemur. Once I proved my point, I brought Window Maker into Lemur.

Right now, my main motivation is Window Maker. I eventually want to shift focus to the Window Maker Lemurs. Xfce will still be a part of Lemur, but I want the Window Maker versions to be the main desktop.

LLM: One of the unique things about Liquid

Lemur is that it offers the Window Maker window manager as a default option. Why Window Maker?

Well, I've been using Window Maker for a long time. Coming from the Amiga computer and having messed with NextStep in the past, I was looking for something fast and easy on the resources. Window Maker fit those needs.

After using Window Maker for a bit, I also realized that it allowed me to get my work done a lot faster. I liked that the desktop environment didn't get in the way either. Sure, there is a learning curve for Window Maker...but that's true of any new OS or app. Once you learn just some of the secrets, under the hood, of Window Maker you will be hooked. The problem is, most just try Window Maker for 20 minutes and think..."yuck!" I say use it for at least a month and stick your head under the hood once in awhile. Then decide. :)

LLM: What do you see as the future of Window Maker? I know Carlos Mafra is leading a 'fork' of Window Maker, which he has named 'wmaker-crm.' Is there enough active development to keep Window Maker going in the future?

Actually, the future is looking good. Carlos' fork is currently in the process of being added to Debian...taking the place of the severely outdated 0.92 release. So, very soon, Debian will have the latest Window Maker code from Carlos in its repos. Now that Debian is getting involved I think we will see some significant improvements in Window Maker, and I think it will still be active many years from now.

LLM: What do you see as Liquid Lemur Linux's role with regard to the Window Maker project?

Well, currently, I think Liquid Lemur is the only "actively" developed Window Maker distro. I also believe that Liquid Lemur is the first to offer a 64 bit Window Maker setup. Additionally, we have started a Window Maker Resurrection Project (WMRP), in hopes of keeping all the themes and dock apps alive a bit longer. We have a website (<http://wmrp.liquidlemur.org>) dedicated to it. Anyone is free to contribute to WMRP. The website is still under development and over the next few months you will see it take shape. The goal with WMRP is to preserve as many Window Maker bits as possible, and become a central location for all Window Maker fans to find Window Maker goodies. There is also work ongoing to update some of the older, more useful dock apps. So, I think in the long run, Liquid Lemur will play a huge role in the survival of Window Maker.

LLM: There is an ongoing debate currently regarding the growing size of distribution iso images. Some of the leading distros have gone to DVD-sized images, with an option for a CD-sized 'network install' image. So far, you've managed to keep Liquid Lemur at the CD-sized level. Where do you stand on this debate? Should distros keep to CD-sized images for the benefit of users who lack DVD capability?

This is a good question and has caused much heated debate in the past. My former Window Maker distro (HEKA OS) was DVD size and I caught much flak for

it. Mostly from Window Maker users. The reason it was DVD size was because I included all the everyday apps I used (which is another debate).

Personally, I think in this day and age that DVD size distros are okay. If Liquid Lemur was DVD size, it would make my life easier. That being said though, Liquid Lemur's original goal is to be a "base" distro that users can expand on as they see fit. So, based on that, I feel it should remain CD size. However, it is getting increasingly difficult to keep it CD size, while maintaining the base layout I want.

If Liquid Lemur was to move to a DVD size distro, then I could include the Desktop Edition bits. But, then that opens up a new can of worms because not everyone wants the same browser, media players, etc.

So, as I've posted online, I will keep Lemur at CD size as long as I can. But, at some point we will need to move to DVD.

LLM: The Xfce version is described as Liquid Lemur's 'flagship' version. What attracts you to Xfce? Why not Openbox or Fluxbox?

Dreamlinux was my first real experience with Xfce and I loved it. I like that it can be molded into many different looks, while still being lightweight. Openbox is just too plain for me. I tried to get used to it, but never could. As for Fluxbox, never really tried it so I can't say much. The Xfce Lemur is described as the flagship version because it is the version that started it all and currently the most popular.

LLM: Last question - It's still very early days for Liquid Lemur Linux. What are your long-term goals for the project?

Well, I would like to have a stable rolling release by using a custom Debian repo. Very similar to what is happening with LMDE. I had started on this idea with my old HEKA OS distro, but never completed it. I would like for the Lemur apps to be complete and fully functional. I want the Window Maker Lemurs to transition to the "flagship" versions. I would like to complete the "Roll your own edition" scripts (both in APE MAN and as a website).

One thing I am working on now is to allow various bits to be user selectable during installation. Another feature I have planned is an OEM installer option. We had this on Dreamlinux and I would like to see it on Lemur as well. This will be needed for my planned Liquid Lemur PCs, currently under development. Yep, exclusive news here... Liquid Lemur will soon have custom Lemur PCs...with media and gaming capabilities, all packaged in a very unique case. I will post more details very soon. :)

eLinks: A Text-Only Browsing Alternative

eLinks: A Text-Only Browsing Alternative
by busprof

Today's internet is heavy with graphical content, so using a browser that is specifically designed to eliminate all of the graphics and provide you with just the text may seem like a step backward, if not downright silly. But consider...When you access the internet, what are you after? If your answer is that you need the graphical content, then you should not use a text-only browser. But sometimes, what you're after is not the pretty images but the solid text content. In that case, why bother with a full-featured browser? In fact, using a 'regular' browser – no matter how good – can sometimes be a distraction when you're really after the text content of an article or post.

Another issue is speed. If you are using the latest Core i950 CPU with 40Gajillion MB of RAM and your ISP is providing you with a fire hose-sized connection - fine. You need not read any further. But if you are on average (or older) hardware or if you are using the slow wifi connection at the local coffeeshop, then a text-only browser might be just the help your hardware or connection needs to keep you from buying yet another three-dollar cup of coffee while waiting for that web page to load.

Finally, there is security. A good deal of malware comes embedded in image files nowadays. Yes, that's right – the very things that characterize today's internet are often the carriers of viruses, trojans, and other nasty pieces of work intended to cause your system harm and compromise your data security. Other sources of malware include java scripts – which are also ubiquitous on the internet today – and flash content. Well, you get the idea. Since text-only browsers do not make use of things like java and flash, and since they do not automatically display image files of any type (but more on the image issue later) your level of internet security is also enhanced when you surf text-only.

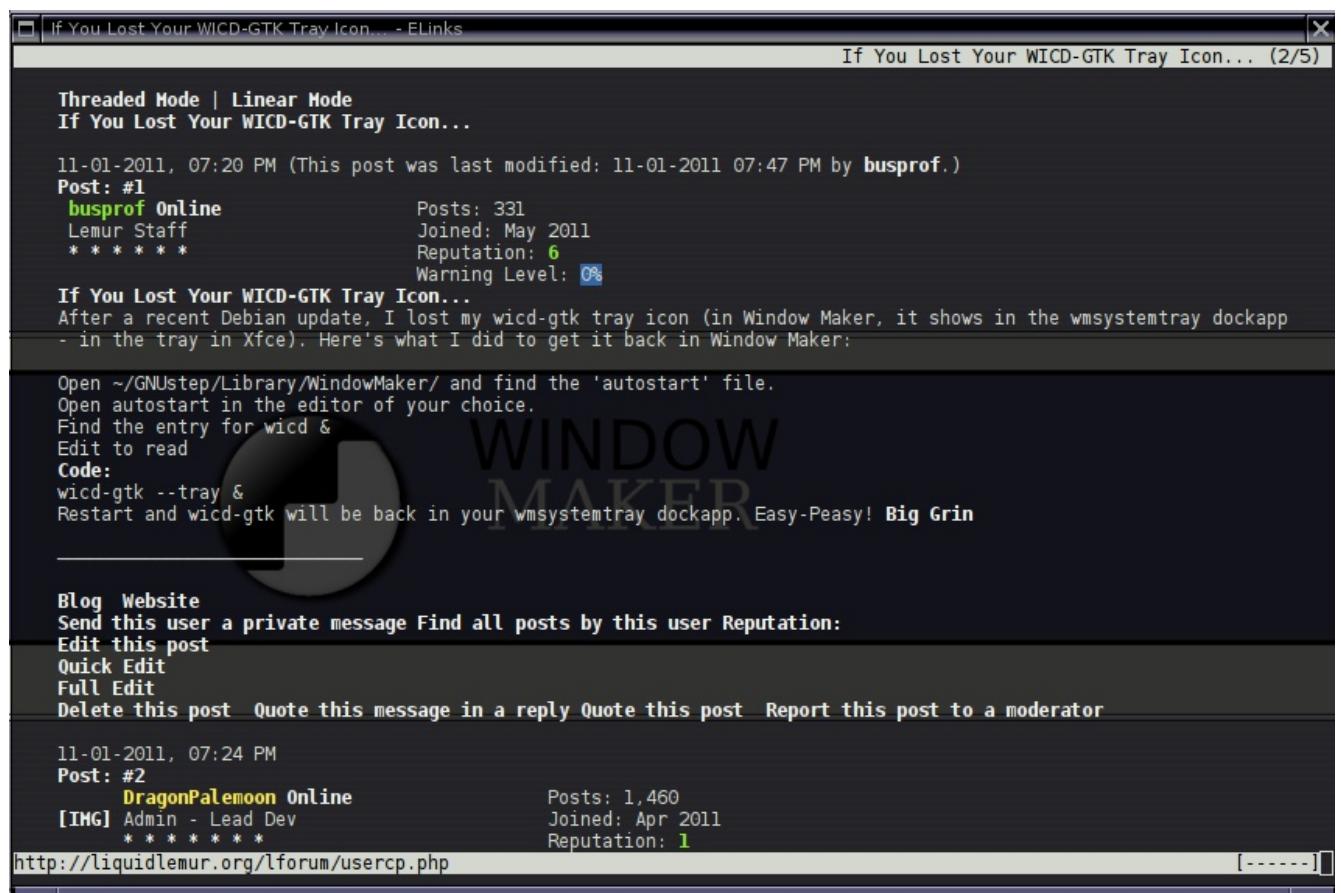
If you would like to try a text-only browser, you have a few choices. The top three text-only browsers seem to be lynx, w3m and eLinks. Having tried all three at one time or another, my favorite continues to be eLinks. eLinks is much more like a 'normal' browser than lynx and w3m. It uses tabs. It can access form fields for data entry. It highlights links in one color and visited links in another. It allows mouse scrolling. It allows for copy-and-paste functionality. In short, it does most things you want and need in a browser without the things you don't need - at least not always

- maybe mostly.

Below is an example of eLinks used to access a post on our own Liquid Lemur Forum:

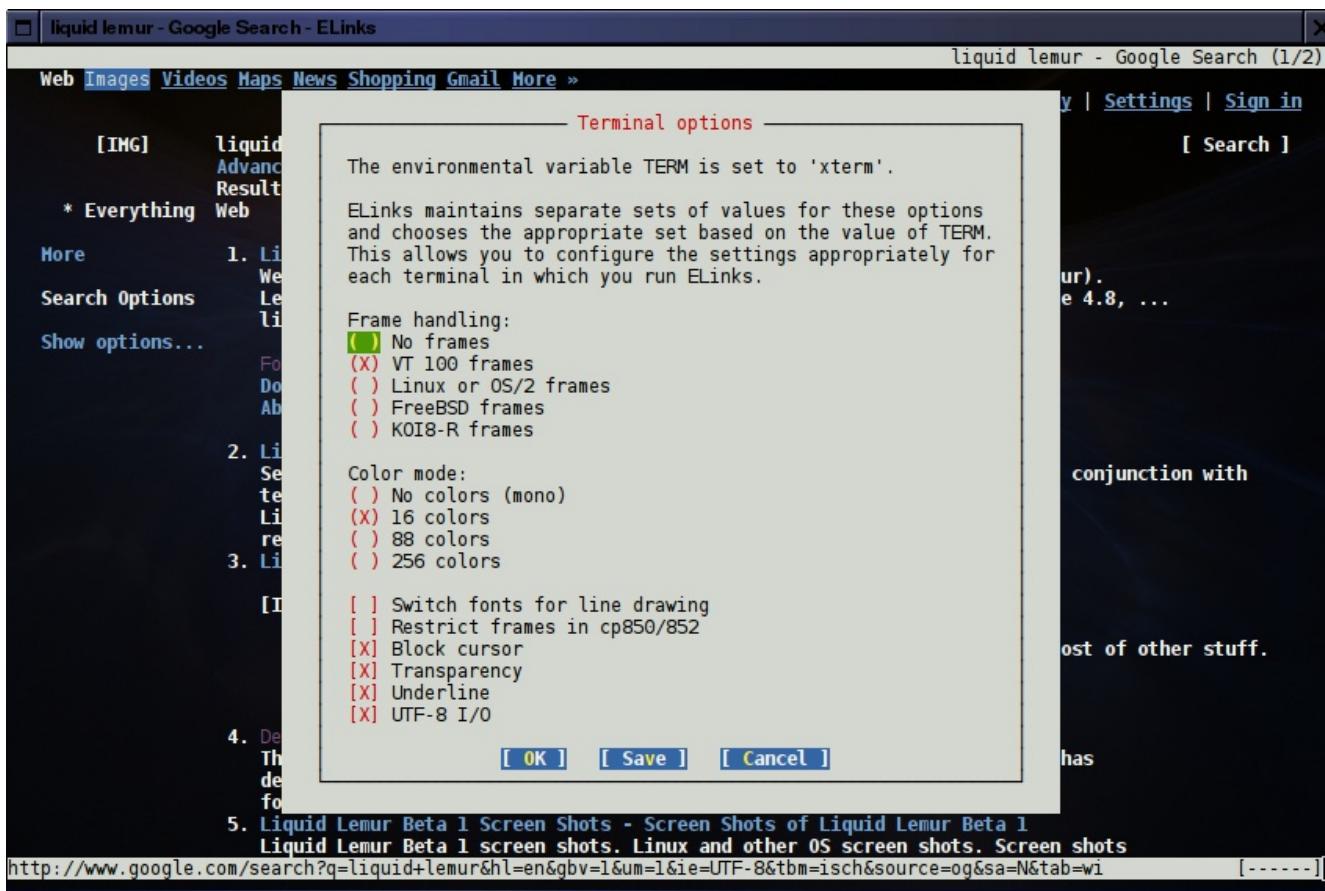
Now, if you try eLinks, there a few things you might want to know to get started.

First, eLinks is available in the Debian repositories used by Liquid Lemur, so getting the application is a simple matter of 'aptitude install elinks.' Once you've installed, open your terminal emulator and simply



enter 'elinks' at the command prompt. You'll be asked to enter a URL by default. Before you do that, though, you might take a moment to do a little setup.

There is a 'Setup' menu accessible by hitting the 'Esc' key and then either clicking or 'arrowing' over to the Setup entry. You'll want to start by playing around with the 'Terminal Options' settings to get the look and feel as comfortable for you as you can. Here are my settings (for now - I like to change things sometimes just because I can):



Also under 'Setup' is the 'options manager.' There are a large number of settings for default behaviors here, and you will want to review these and change anything you think appropriate. For example, you can add faux 'LED' indicators to the bottom right corner of the display that will show when certain issues arise, and you can add a clock to the same screen area.

Now that you've done your initial setup, navigating is pretty intuitive. Type 'g' and you'll get the URL entry field. Enter the URL of your choice (I suggest a site

you are already familiar with from previous visits). You can move around the pages you visit in a variety of ways.

Your mouse scroll wheel works. The PageUp and PageDown keys move you a full 'page' at a time. The left and right arrow keys are the same as 'forward' and 'back' buttons in a 'normal' browser. The up and down arrow keys move you from link to link on the page.

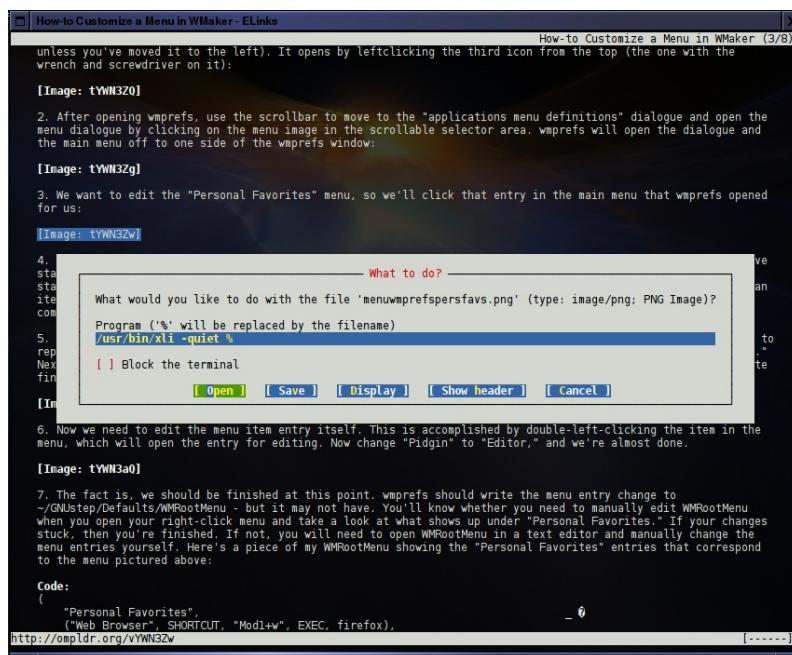
You can open a new tab in elinks by hitting the 't' key on your keyboard. The tabs are displayed at the bottom of the screen - not the top - so keep this in mind. You can close a tab by using the 'c' key. Attempting to close that last open tab in the browser causes a prompt asking if you really want to quit elinks.

The menu can be accessed by the 'Esc' key, or by left-clicking the light-colored bar at the top of the display.

As with a number of browsers, elinks uses cached pages to speed page loading. In elinks, case - maybe a bit too much reliance on cached pages - so if the page looks as though it should be a bit different, reload the page by hitting 'Control-r' (this applies to some forms, such as login forms, which don't always show you as logged in until you refresh the page).

Highlighting text requires that you hold down 'Shift' while using the mouse to highlight the desired text. Once highlighted, you can copy-and-paste as you would in any other browser.

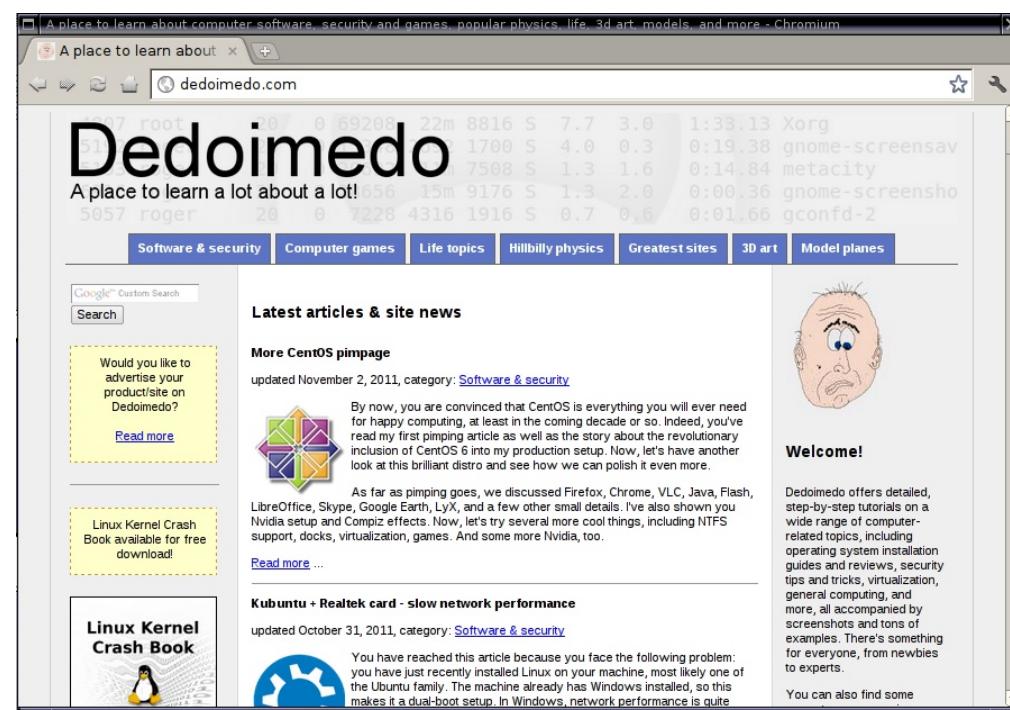
Now, about those graphics - elinks shows links to images, but not the image itself. If you want to see an image, clicking the link will open a dialogue that asks 'What to do?' If you want to open the image in your default image viewer, just click 'open.' Other options include 'save,' 'display,' and 'cancel.' The 'display' option shows the actual compiled code that makes up the image (not very useful).



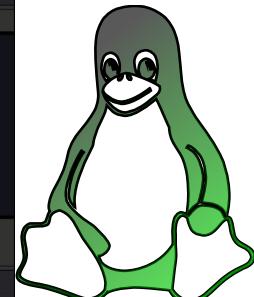
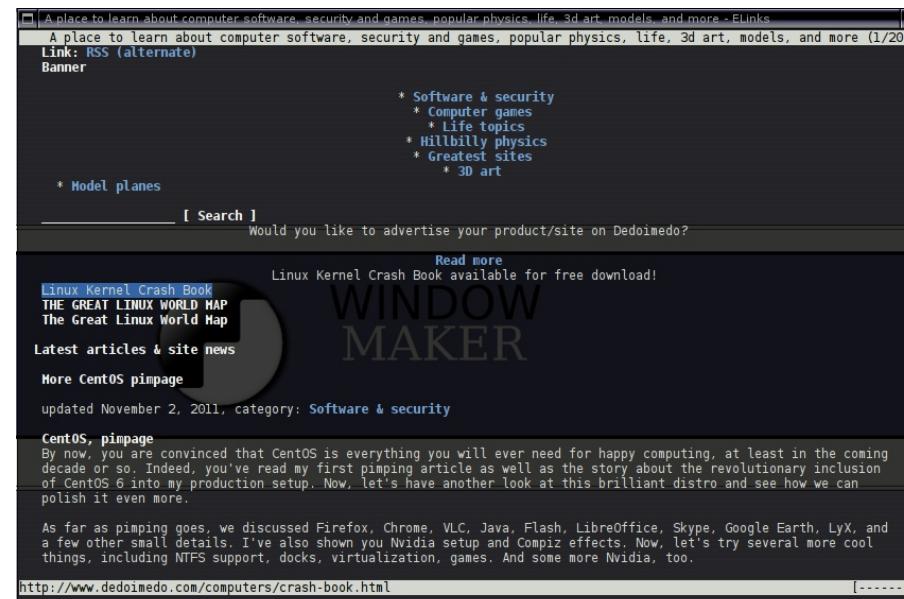
The 'header' option shows information on the image, the server, and some other things that are interesting but, again, not all that useful in most cases. You'll probably use 'display' and 'save' most often. Just as with images, links to downloadable files are shown. elinks handles file downloads very well, and it's at least as good in this area as any 'normal' browser. Clicking a download link will give a menu where you can enter the appropriate path and filename. When you are finished with your elinks session, hit 'q' to quit.

If you need more help, the homepage for the elinks project is <http://elinks.cz>. There is a good deal of information on configuration, and a nice collection of scripts to extend functionality.

Give elinks - or one of the other text-only browser mentioned in this article - a try. You'll be glad you did!



Dedoimedo.com in Chromium (above) and in elinks (below)



Adopting Open Source Software: Reasons and Best Practices

Adopting Open Source Software: Reasons and Best Practices by Max Haifei Li (hli@uu.edu)

As an associate professor of computer science at Union University, I have used many software packages throughout my teaching career. Some of them are proprietary software like Microsoft Office 2010 and Macromedia Dreamweaver. Some of them are open source packages like XAMPP and Notepad++. All software packages that I have used for my teaching are free to me because I don't have to pay for them out of my own wallet. However, I enjoy open source software more than proprietary software for various reasons.

Freedom and Convenience

The first reason for adopting open source software is freedom and convenience. One frustration that I have with Microsoft Office 2010 is that I cannot give it freely to my students due to licensing restrictions. Some students may have Microsoft Office, but it is usually an old version like Office 2007 or Office 2003. When these students use old versions of Microsoft Office to do their homework, they often get lower grades because of compatibility issues and I have to specifically ask them to use the correct version of Microsoft Office. Some students do not have Microsoft Office and I need to ask them to buy it through a discount website. Office 2010 costs about 95 Dollars for students, and some students simply do not have money to buy it. Another frustration with Microsoft

Office is the inconsistency of the interface between the Windows version and the Mac version. We use the Windows version of Microsoft Office in our classroom and I have to tell Mac users (there are a lot of Mac users among college students!) that they simply cannot use their Macs for my class. A lot of students are disappointed with that and they feel that they are treated as second-class citizens. As for open source software, I can just tell students to download from websites and they are ready to go.

"...Microsoft bribes my department and my university so I can get (their products) for free."

Low Cost

The second reason to choose open source software is low cost. Even though the basic open source software is free, refined open source software with professional support is usually not free. However, the price is much more affordable when compared with proprietary software. For example, Union University has adopted MoodleRooms, the commercial version of the open source Moodle Learning Management System software, for its course management system. Since it is based on open source software, it is much cheaper

than Blackboard that we are currently used. From the training sessions that I have recently gone through, I am thrilled to find new features that I really want but Blackboard does not support. (Editor's Note: Blackboard is a learning management system, as is Moodle. LMS's provide tools that teachers use to provide online content and access to students.)

Community Support

The third reason for adopting open source software is the community support that popular open source software enjoys. I use PHP and MySQL in teaching my database management course. Whenever I have a particular issue with Open Source Software, a simple Google search gives me the best clue. Sometimes, I need to spend more time digging deep into the search results, but most issues can be solved in a short period of time. Proprietary software vendors control their software and they generally don't want to let customers know the shortcomings of their software because of the fear of bad publicity. Apple is an example. Apple hides the facts that Macs can also get viruses and does not provide much-needed help to users that are actually affected by Mac viruses.

Best Practices when Choosing Open Source Software

The first best practice advice is to be both open and conservative. "Open" means that you can explore different choices. The choice should be based on your unique situation. "Conservative" means that you need

to be careful in choosing which open source software to use for your project. Because of the open nature of Open Source Software, there is no penalty if the developers or maintainers of the project decide to abandon the project. As a result, you may get stuck with open source software that is no longer supported by a developer or maintainer. If you are good at programming and not afraid of DIY (Do It Yourself), you can fix the problem easily because the source code is available. For most people that don't have time or necessary skill set to study the source code, the best strategy is to choose the most popular ones that are likely to stay alive and in active development for a long time.

The second best practice advice is to seek opinions from friends and online communities. The rise and fall of open source software depends on the popularity rating. Generally speaking, the quality of open source software is directly related to its popularity. If you don't know much about which open source software to use, ask your contacts about their opinions on which open source software to choose. Opinions from online communities vary dramatically and you should be careful in deciding which source to trust. Your friends and colleagues can give you honest answers to most of your questions.

Conclusion

Surprisingly, I do not spend a lot of time using LibreOffice, a popular alternative to Microsoft Office. The first reason is that Microsoft "bribes" my

department and my university so that I can get it for free. The second reason is that I teach Microsoft Office and I have to learn a lot of details about Microsoft Office. I like Microsoft Office but I think it has become too expensive, and I also think the cost will continue to increase. Ask yourself: when was the last time you saw a proprietary software vendor reduce the price of its product?

"When was the last time you saw a proprietary software vendor reduce the price of its product?"

Let me briefly talk about open source software that I am using. Firefox is my favorite browser. Even though Google Chrome is surging in its popularity, I am not a big fan of Google. I would say that I am not a big fan of any big business, including Microsoft, Apple and Google. Notepad++ is my favorite editor for many computer languages including PHP and SQL. I prefer Eclipse over NetBeans for software development because Oracle (another "evil" in my opinion) controls NetBeans and I have concerns about NetBeans' future. I use 7Zip for compressing and uncompressing because I don't want to pay for WinZip.

Theming Window Maker Part 1: Styles

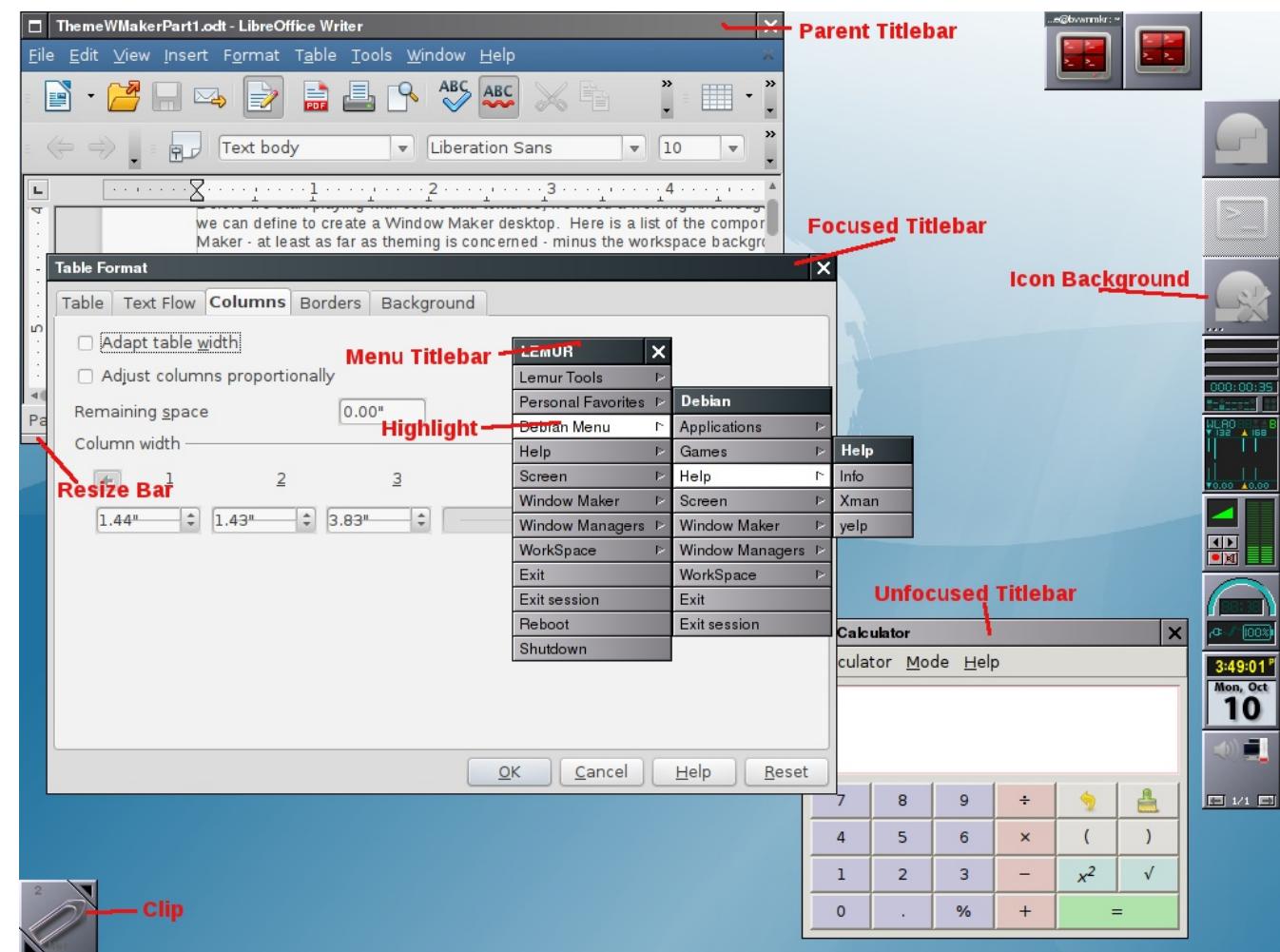
Theming Window Maker Part 1: Styles by busprof

Window Maker is very easy to theme. The foundation for all Window Maker themes is the 'style' file, and the most basic type of 'style' file defines all desktop components except the workspace background image.

Window Maker themes that lack a background image are referred to as "styles." In this article, we're going to accomplish two goals with regard to styles. First, we're going to carefully examine the components of a style. Second, we're going to hand-build a style from scratch - and without resorting to a GUI tool, although a couple of these are available by default in Liquid Lemur Window Maker. It's my belief that learning the components of a style file, then actually using that knowledge to build a style, will enhance your ability to use the GUI tools (if you even want to use them after practicing this) and will increase your understanding of the Window Maker environment generally.

Components of a Style

A style defines the basic components of the Window Maker desktop, so we need to know how Window Maker identifies these components. For theming, we're most interested in two key attributes of these components. The key attributes we need to define in a style or theme are color and "texture." Texture in this context means using multiple colors in various types of gradients - we're not limited to solid colors only.



In this first article on theming, we're not going to use image files for our elements, and we'll stick with a solid color for each element. We will cover texture and the use of images in future articles. (We'll also be interested in fonts – but more on that later).

Before we start playing with colors and textures, we

need a working knowledge of the various components we can use to create a Window Maker desktop. Above is a screenshot of a typical Window Maker desktop with the key components labeled.

In the highlight box on the following page is a list of the components recognized by Window Maker minus the workspace background itself.

Specifying Colors for Components

Colors are specified either as color names in the standard X color name format or by color numbers in the standard rgb (red-green-blue) format. You can use any color name shown by the showrgb program (like black, white or gray) or any color value in the #rrggbba format, where rr, gg and bb is the intensity of the color component (such as #ff0000 for pure red or #000080 for medium blue). Note that color names in the #rrggbba format must be enclosed with double quotes.

In Liquid Lemur, there are a large number of rgb color names pre-defined, and it is interesting to open a terminal, enter showrgb, and look over the choices you have. For example, there are four different "tomato" shades defined (tomato1, tomato2, tomato3 and tomato4 – obviously!). So the showrgb program (installed by default in Lemur) can be a valuable resource for theming and styling.

For solid colors, the attribute is defined by
`<AttributeName> = (solid, <Color>)`

For example, if I wanted icon backgrounds to be light blue, I would use

IconBack = (solid, LightBlue)

I chose the string `LightBlue` to define the color because the color list generated by `showrgb` provided it as the correct color name for my system. (In reality, most of these color names have become quite standard over time – so if you're worried about portability, don't – most Linux distros use identical color names now.)

I could also use a color picking program such as the one available at <http://www.colorpicker.com> to generate an rgb number. I might then use the rgb number as follows:

IconBack = (solid, "#8FB8F2")

Note that I have enclosed the rgb number with quotation marks. This is necessary because the color number contains a non-alphanumeric character (the # symbol) and therefore requires the quotation marks for proper interpretation by Window Maker. Any time a string contains a non-alphanumeric character in a style file (characters such as '#', ''.' etc) you need to enclose that string with quotation marks.

How to use showrgb to find colors:

In a terminal, entering `showrgb` will generate a list of all colors recognized by your system. If you are interested in a particular color or group of colors (for example - shades of blue) then enter '`showrgb | grep blue`'. This will generate a list of all the colors containing the string 'blue.' Note that your system's colornames are sometimes capitalized, so you'll want to also enter '`showrgb | grep Blue`' to ensure that you are seeing all possible blue shades available.

Our First Style File

With the information we have covered so far, we can put together a basic style, save it into our `~/GNUstep/Library/Windowaker/Styles` directory, and use it on our system.

Step 1:

Start by opening your text editor and creating a file named `~/GNUstep/Library/WindowMaker/Test1.style`. Copy or type the following “template” into the file:

```
{
IconBack = ();
FTitleBack = ();
FTitleColor = ;
PTitleBack = ();
PTitleColor = ;
UTitleBack = ();
UTitleColor = ;
MenuTitleBack = ();
MenuTitleColor = ;
MenuTextBack = ();
MenuTextColor = ;
HighlightColor = ;
HighlightTextColor = ;
MenuDisabledColor = ;
MenuTitleFont = "-*-lucida-medium-r-normal-*12*-*-*-*iso8859-1";
MenuTextFont = "-*-lucida-medium-r-normal-*12*-*-*-*iso8859-1";
MenuStyle = ;
ClipTitleColor = ;
CClipTitleColor = ;
```

```
ClipTitleFont = "-*-helvetica-medium-r-normal-*10-*-*-*-*-*";
IconTitleFont = "-*-helvetica-medium-r-*-*8-*-*-*-*-*";
ResizebarBack = ();
Displayfont = "-*-helvetica-medium-r-*-*10-*-*-*-*-*";
TitleJustify = ;
WindowTitleFont = "-*-lucida-medium-r-normal-*12-*-*-*-*-*";
}
```

The template simply provides you with all of the components I described earlier, formatted for use in a Window Maker style or theme. Before we go further, let me point out a few things about the template. First, notice that the file begins with an opening `{` sign, and ends with a closing `}` sign. These brackets must be present at the start and end of the file or the style will not load. Next, notice that all of the component names are followed by an `=` (preceded and followed by a space) and a concluding `;` sign. These three elements must be present for each component or your style will not load.

Next, notice that some of the elements are followed by a pair of parentheses `()`. These correspond with elements that can carry both color and texture definitions. For these items, we can specify texture commands that will produce various color gradients. We will enter color and texture information between the parentheses. For now, we're just going to use solid colors – we'll add some gradient textures later.

Finally, notice that I have “pre-loaded” the font information. Window Maker themes and styles allow for specifying fonts using the X Logical Font Description format. This is a very complex (some would even say unnecessarily complex) syntax, and I'm not going into it in this article. You can see the syntax here:

http://en.wikipedia.org/wiki/X_logical_font_description or you can use a program like `xfontsel` to find and select fonts. `Xfontsel` will provide all of the syntax defining a particular font for you. For now, I've simply provided a couple of examples.

When selecting fonts, remember that a font that is not installed on your system (or someone else's system, if you're sharing your styles and themes) may not display properly – the system will choose an alternate font for you if the one you specify is not installed.

Step 2:

Now we're ready to start building our style. For this exercise, we'll use various shades of blue and gray – but again, run `showrgb` and take a look at the dozens of pre-defined color names available, or go to `colorpicker.com` and use your imagination! Enter (or copy and paste) the following values for the components in your template:

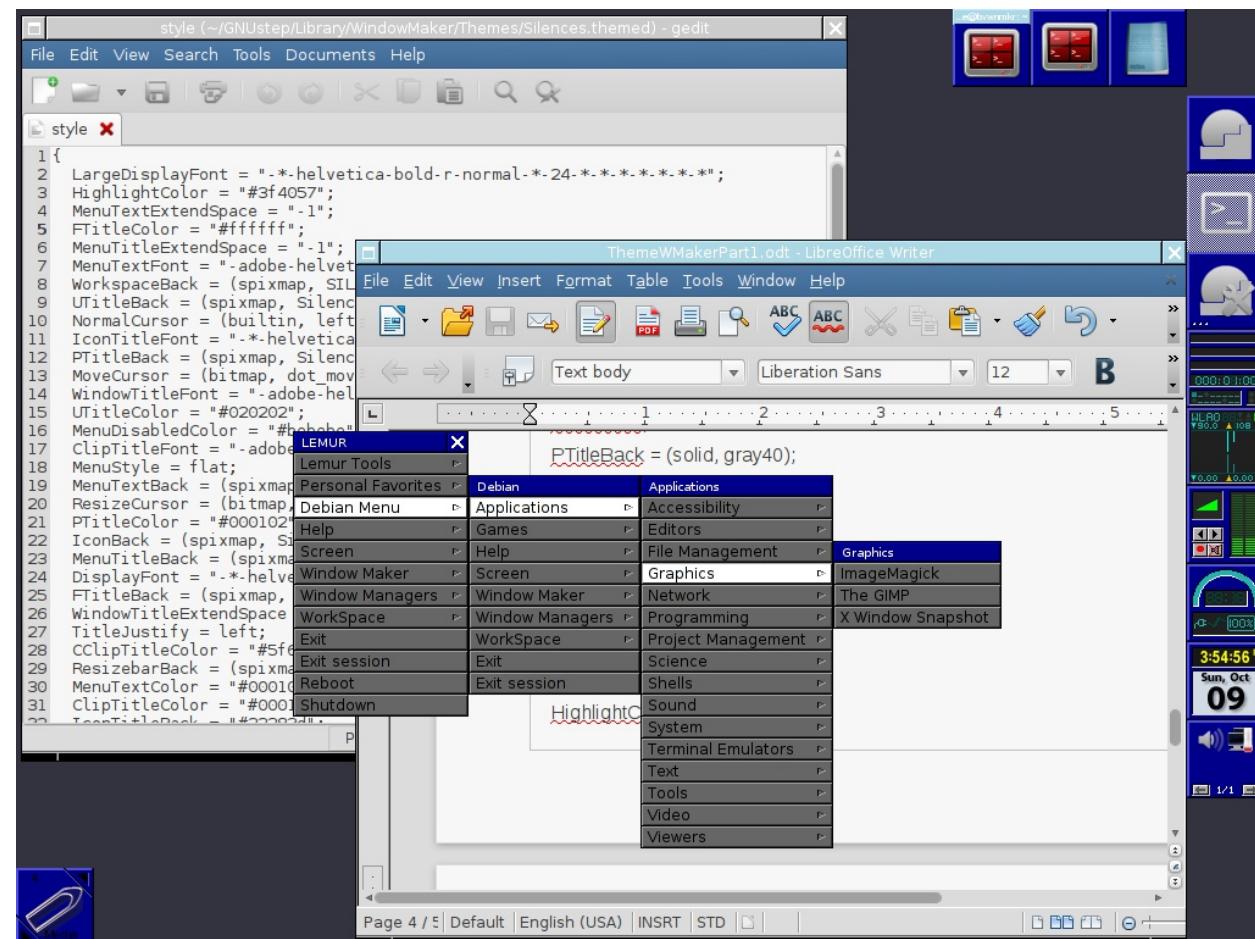
```
{
IconBack = (solid, DarkBlue);
FTitleBack = (solid, LightBlue);
FTitleColor = white;
```

```

PTitleBack = (solid, gray40);
PTitleColor = white;
UTitleBack = (solid, gray80);
UTitleColor = white;
MenuTitleBack = (solid, DarkBlue);
MenuTitleColor = white;
MenuTextBack = (solid, gray40);
MenuTextColor = black;
HighlightColor = white;
HighlightTextColor = black;
MenuDisabledColor = gray80;
MenuTitleFont = "-*lucida-medium-r-normal-*10-
*-*-*-*iso8859-1";
MenuTextFont = "-*lucida-medium-r-normal-*12-
*-*-*-*";
MenuStyle = normal;
ClipTitleColor = black;
CClipTitleColor = black;
ClipTitleFont = "-*helvetica-medium-r-normal-*-
10-*-*-*-*";
IconTitleFont = "-*helvetica-medium-r-*-*8-*-*-*-
*-*-*";
ResizebarBack = (solid, black);
Displayfont = "-*helvetica-medium-r-*-*10-*-*-*-*-
*-*-*";
TitleJustify = center;
WindowTitleFont = "-*lucida-medium-r-normal-*12-
*-*-*-*-*";
}

After saving this into your
~/GNUstep/Library/WindowMaker/Styles directory, you
can select your new style from the menu under
Workspace—Appearance—Styles. Selecting the

```

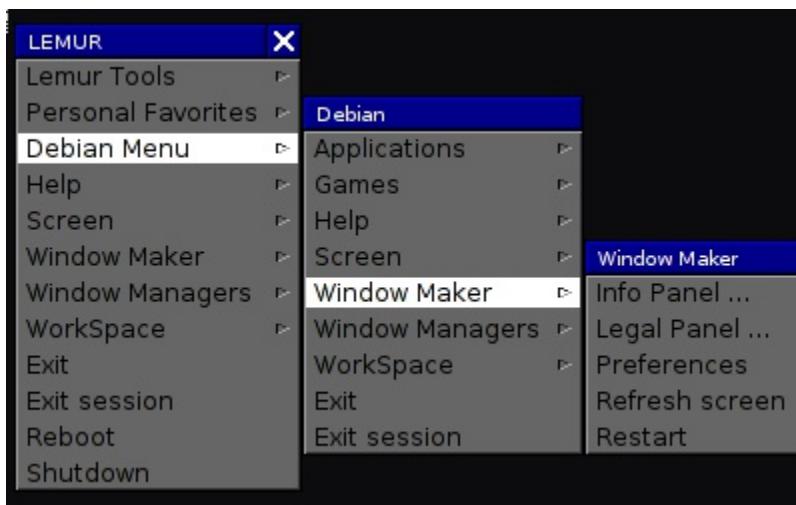


Test1 style should give you a desktop that looks something like the screenshot shown here.

So, we've got our dark blue icon backgrounds, dark blue menu titlebars, light blue focused window titlebars, and light gray unfocused window titlebars. Our menu items have a medium gray background with a "normal" (beveled) separation between each item. The menu titlebar text is white, and the menu item text

is black. Finally, we have a black resize-bar at the bottom of each window.

Now let's try a brief experiment. Open the Test1.style file, and change the entry for MenuStyle from "normal" to "flat." Save the Test1.style file, then reload your style from the Workspace---Appearance menu. Right click your desktop and take a look at the menu. It should now have a smooth menu field instead of a beveled field, like the one shown here.



Would I keep this style and use it? Probably not. I would want to play around a bit with the color of the focused window titlebar, and I would also want a different color for the window titlebar text for both focused and unfocused windows. The menu titlebar text looks too small to me – it should be at least the same size as the menu item text (in most cases, anyway) and I might want to play with the highlight color and highlighted text color to give more contrast and appeal. But now that you know how easy it is to define a style in Window Maker, you can see that all of these changes are easy to accomplish.

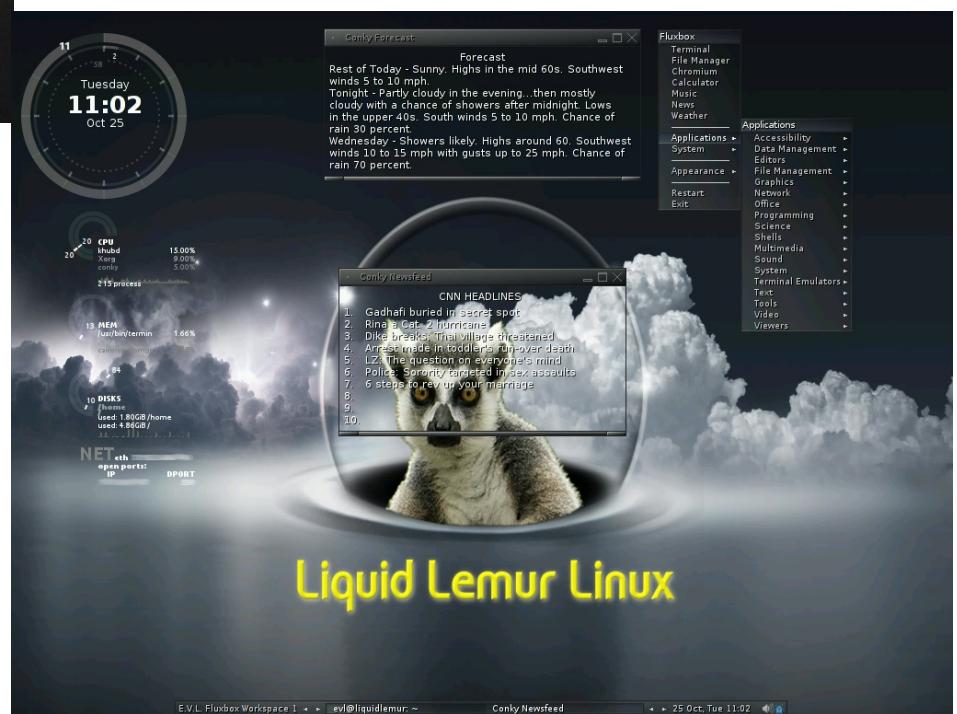
So...What are you waiting for? Get busy styling your Window Maker desktop!

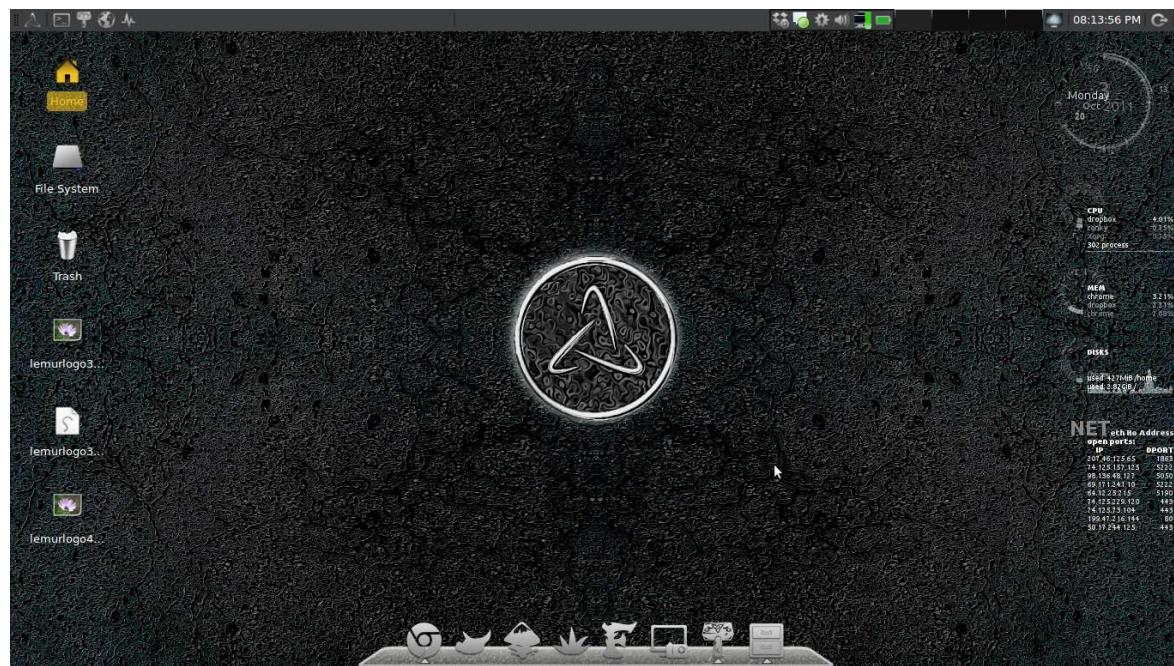
Next time we'll add to the level of complexity by including color gradients (textures) and adding a workspace background, which will move us from the 'style' category into the full-blown 'theme' category.

Screenshots!



Above: DragonPalemoon on Window Maker

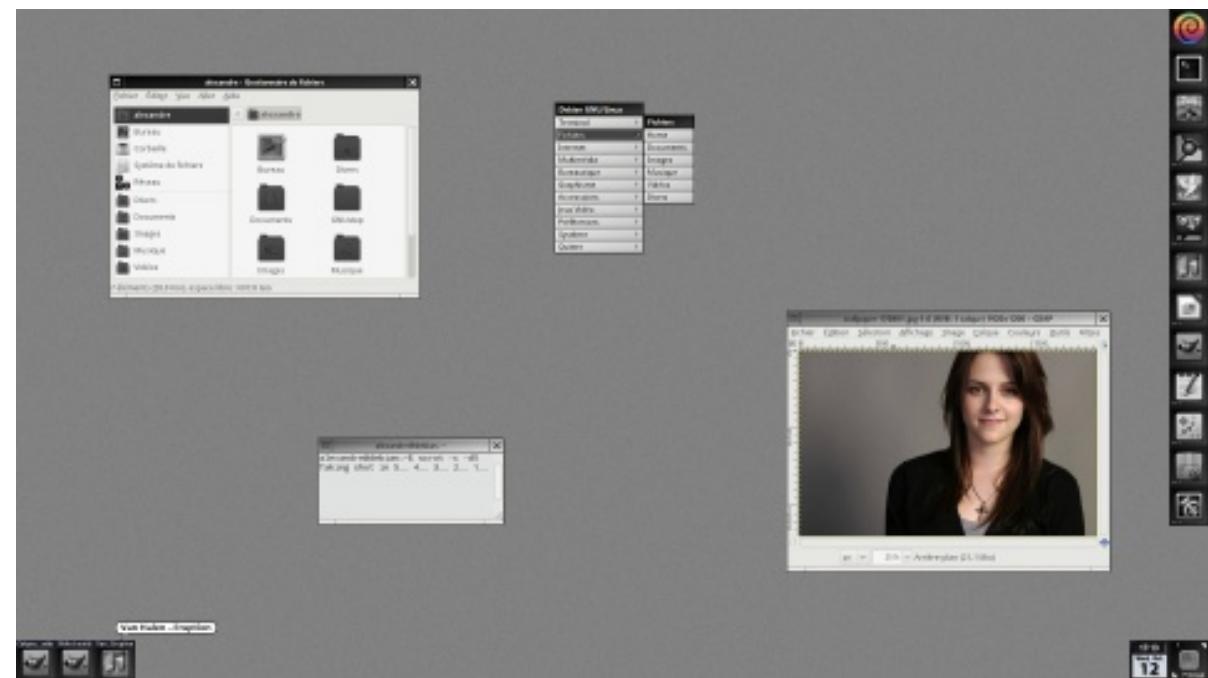


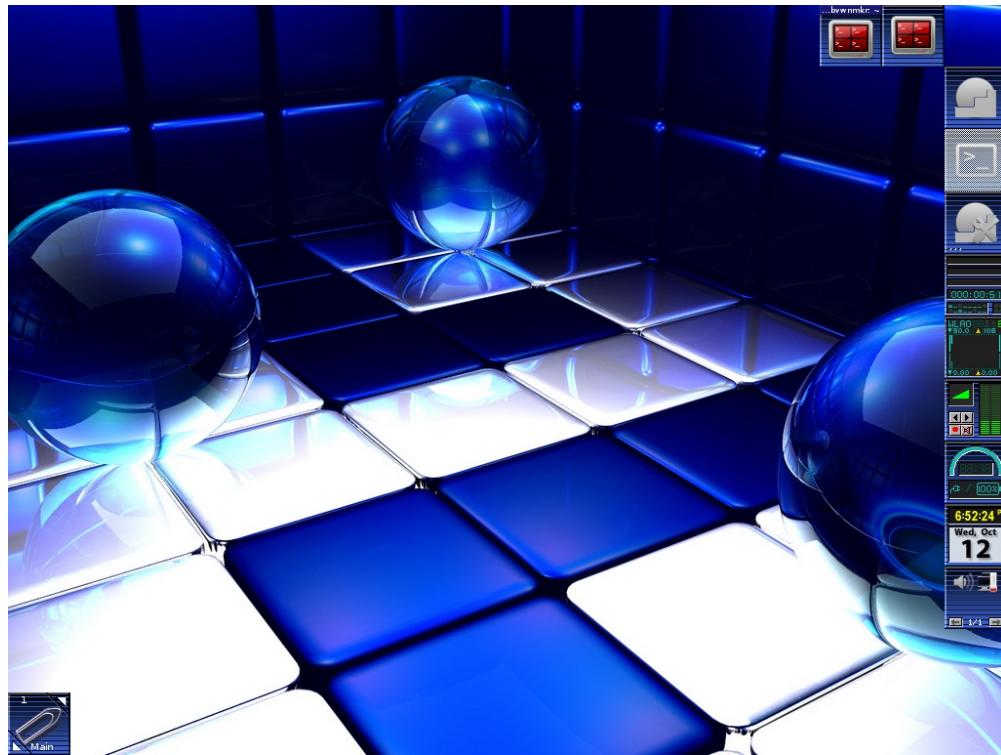


Above: dgf64 on Xfce

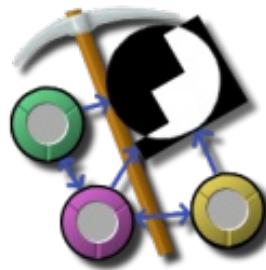


Below: darthwound on Window Maker

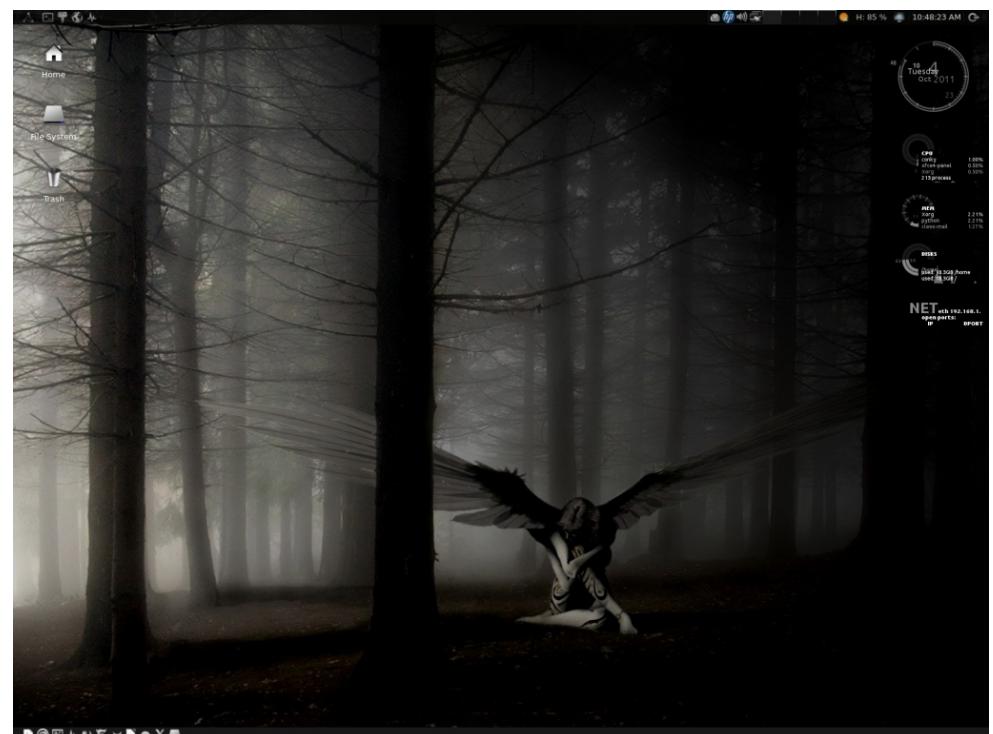


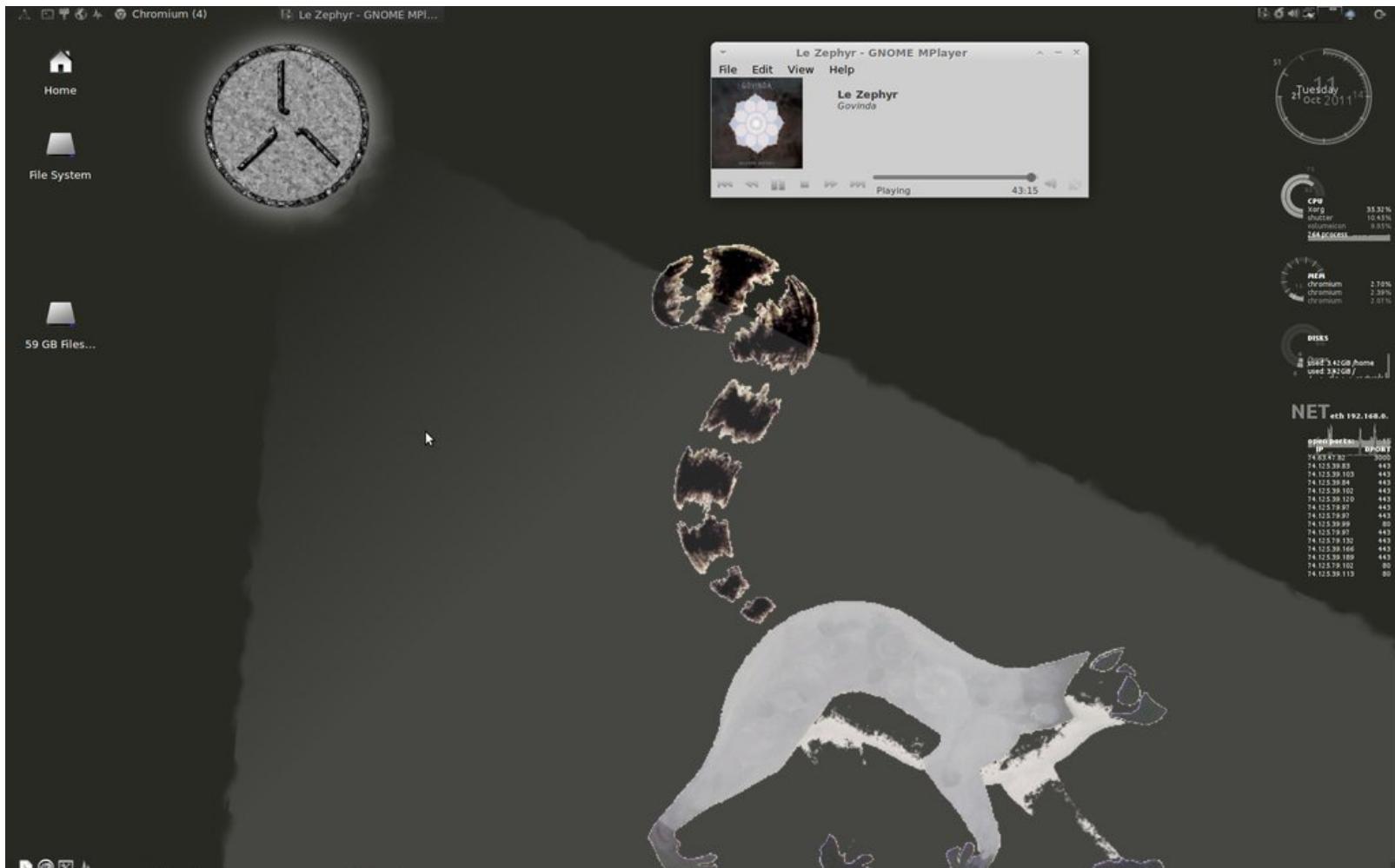


Above: busprof on Window Maker



Below: DragonPalemoon on Xfce





Release Ramblings

Release Ramblings

by DragonPalemoon, Lead Developer

After what seemed like an eternity, my PC is back up and running. So, development of the Lemur version 1.0 release is in full swing. I've got some catching up to do as a result of the downtime. But, rest assured, I still plan to have Liquid Lemur 1.0 ready by the end of the year.

So, where are we and what's the game plan going forward? Well, we are still good but not as efficient as we once were. See, normally I have at least three PCs to develop and test Lemur on. Now I am down to one. This is not too bad, really. It doesn't affect much as far as "development" goes, but mainly slows down the building process. Now, I can only build one ISO at a time. In reality this won't hurt us that much. I just hate being a one-PC developer. :P This will all go back to normal after the first of the year, when the Lemur budget gets back on track. Speaking of budget, I can't thank the community enough for stepping in and helping out Lemur! Thanks to you, Lemur is safe.

Going forward, the plan is simple. Clean up the ISOs, finish the Lemur Apps and docs, and decide on a 1.0 theme / layout. (However, thanks to busprof and dgf64, deciding on a default theme is getting increasingly difficult!) Many in the community have suggested that there be a "lemur" in the default theme, that much I can promise you will happen. We will also have the new logo from dgf64 in there. The default Icons will be a slightly modified AwOken icon set.

Everything else is still up in the air at the moment.

Sticking to the original Lemur design goals, Liquid Lemur will remain a CD-sized distro. I will be trimming as much fat as I can to keep it that way. Part of the problem is that Debian feels the need to "depend" on packages that really aren't needed for a particular app. For example, Cairo Dock installed from the developer's debs takes up about 20mb. The same version installed from Debian repos takes up about 120mb! Why? Because there are some plugins in Cairo Dock that utilize other apps and Debian feels the need to include every one of those apps! While I can understand the logic behind such action, I don't agree with it. Especially when some of those plugins can use multiple apps, letting the user decide which app to associate with that plugin. So, I will have to manually do some cleanup and remove all the extra / not needed fluff. I did a test build of Lemur this morning and noticed it pulled in some Gnome 3 libs!? So, apparently Gnome 3 is working its way into Testing (Wheezy) now. This should make future builds of Lemur interestingly fun.

On a side note, I've been working to re-write the Dreamlinux tools to work with a Debian base and hope to use them to build Lemur in the future. Although it will not solve the "Debian-depends-on-everything" issue, it will make building easier (especially for those wanting to make Lemur re-spins). I know some of you have suggested dropping Debian as a base for Lemur, and sometimes I wonder if I should.

There has been some flip-flopping of kernels in the past couple of releases. Initially, a few had trouble with the Liquorix kernels. So, I dropped them and just went with Debian kernels. Then Debian deprecated the plain 686 kernels and I was forced to include the 486 kernel for those on older hardware. Then I started running out of room and had to drop the 686-pae kernel from the 32 bit Lemurs and include only the 486 kernel. I wrote a script to test for a PAE capable CPU after install, but it had a typo. Some complained Lemur was slower with the 486 kernel. And it will be if you have a newer CPU (486 kernels only support ONE core.). So we had some confusion for a bit, to say the least. I was left with trying to decide on how to deal with this. One thought was to just drop the 486 kernel, but there are still a few that needed it. So, another thought was a separate ISO for the 486 builds. But, I just don't have the time for another ISO. The third option is to fix the PAE script and leave things as they are. For now that is what I am doing. The 486 kernels will remain in the 32-bit ISOs and after install you will be offered the 686-pae kernels. This may change at anytime...just to warn you.

The Lemur installer is getting a few modifications. Other than being updated to a newer version, it will now ask you if you want to install the Desktop Edition, multimedia codecs, and other bits during the install. I'm also working on an OEM installer (Editor's Note: OEM stands for "original equipment manufacturer"). This will be needed for the upcoming Lemur PCs I plan to design and build. It can also be used by others wanting to build custom PCs with Lemur preinstalled.

For those not familiar with OEM installers, it's basically the regular installer, but the user bits don't come up until AFTER the initial installation. So, as a PC builder you run the installer and then when the user powers on the PC for the first time, they will be asked to fill in the user details (name, password, etc.). The OEM installer probably won't be ready until the final 1.0 release of Lemur.

The main goal for now is to get the Lemur Apps and scripts working correctly and clean the distro up a bit. I also want to get the default layout, themes, artwork, etc. figured out. One other area needing attention is the Xfce dock / second panel option. Some of you prefer a dock and some prefer the second panel. So, in an effort to please both sides, I've created a script that will let you decided which you want. This will be in both the Live session and after installation. Once you decide, it will not ask you again. I have added an option to the menu and the control panel, should you change your mind later. :) The script still needs a bit of work though.

One final note - I am working on a custom DVD-sized Window Maker Lemur. This will basically have everything you might need in an OS, including the kitchen sink. I may do one for the Xfce Lemurs, depending on how things go. I will only need to do this once, since you will be able to upgrade / update it via APE MAN.

So, things are progressing again and version 1.0 should be out by the end of the year! :)

Lemur Releases: A Brief History in Screenshots



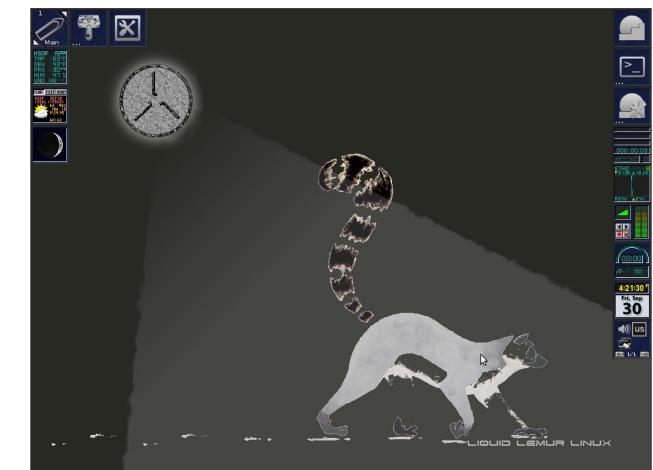
Alpha 1 Xfce



Beta 3 Window Maker



Beta 1 Xfce



Release Candidate 1 Window Maker

This is the current version, as listed recently at DistroWatch (<http://distrowatch.com/table.php?distribution=liquidlemur>)

Dockapp Roundup: wmpinboard

Dockapp Roundup: using wmpinboard by busprof

wmpinboard was created by developer Marco Goetze, although no development activity has been recorded on this application for at least five years. Despite its age and the lack of ongoing development, the application continues to work very well on an up-to-date Window Maker desktop (like, say, the Liquid Lemur version!).



The man entry for wmpinboard describes this neat little dockapp as follows:

"wmpinboard is a Window Maker dock applet resembling a miniature pinboard. It's intended to somewhat relieve heavily littered desktops by allowing you to place reminders on a graphical on-screen pinboard rather than producing a mess of real notes all around your keyboard (thus being environmentally A Good Thing, too ;-)"

...and goes to say that-

"Besides text, you can add small monochrome sketches to your notes or simply encircle or underline words as a means of emphasis, and alarms can be set to explicitly remind you of things. Above all, wmpinboard is animated in redundant ways to make it

look even more attractive, and themeability provides for a way of adapting its appearance to that of the rest of your desktop."

Getting Started

Lemur users can activate wmpinboard from the Debian menu (look under "Applications-->Tools") or from the command line by entering the command - wmpinboard.

When you first open wmpinboard, you have a blank "corkboard" graphic overlaid on a 64x64-pixel Window Maker tile. At the top of the board is the "TODO" button. Left clicking this button generates a "note," which can then be edited with up to 59 characters. To save the note, you can simply wait for the note to be minimized to the board (default is 60 seconds) or press your Esc key to return to editing mode and minimize the note. Left-clicking any note restores it to full view, and places the note in editing mode again.



Organizing Your Notes

The author of the dockapp recommends using different colors ("note paper" colors) to help organize

your notes. The default seems to be green, but there are several colors available including a couple of different shades of blue and bright red. I could envision using green for average, everyday work notes, perhaps blue for non-emergency family notes, and red for screaming, panic-inducing, "if-you-don't-get-this-done-the-world-ends-tomorrow!" notes.



Another means of organizing notes is to move them to different areas on the board. When you move a note, it gets "wrinkled" from "handling" - just like a real note might get worn from being moved from place to place on a real desk.

To start a new note, simply start the process as you did earlier by clicking the "TODO" button. By default, the new note will be in a different color. If you want to change colors, you can use the Shift-Tab keyboard shortcut, or use the Options menu.



The Options Menu

Right clicking the triangular tab in the bottom right-hand corner of the note brings up a little "options" menu. The menu buttons are small graphics that give a hint as to each button's function. Unfortunately, there is no "ballon help" feature providing a verbal description, so you will need to learn what each of the eight buttons provides (I usually do this by trial and error, but feel free to read the man page in your terminal - man wmpinboard).

The options menu is laid out as follows:

```
+---+---+---+
| a | c | e | g |
+---+---+---+
| b | d | f | h |
+---+---+---+
```



Here are the descriptions for each of the menu options adapted from the application's man page:

- (a)** Left-clicking on this button opens and closes the alarm panel, which allows you to configure alarm settings for the note being edited. When the alarm panel is visible, the alarm is activated. To turn it off, press the button again and make the panel disappear.
- (b)** This button cycles through all colors available for notes (20 colors).
- (c)** This button closes the panel and returns to edit mode, with the sketch-drawing feature enabled.
- (d)** Closes the panel and returns to edit mode, with the

sketch-erasing feature enabled.

- (e)** This button removes all entered text on the current note and places the text cursor on the very first character. It also closes the panel, thus returning to edit mode.
- (f)** Pressing this button completely removes a drawn sketch on the current note and returns to edit mode.
- (g)** This option removes (erases) the entire note from the board and returns to pinboard view.
- (h)** This button merely closes the panel (and thus puts you back in edit mode). The same can be achieved by simply right-clicking in this view.

Other Features

As you may have noted, wmpinboard has an alarm feature that allows the user to set daily alarms or date-specific alarms (alarms that happen only once on a specific date at a specific time). The default mode for alarms is date-specific - which prevents users from accidentally creating an alarm that will go off every day without intent. If you wish to try the alarm, I recommend that you first read the man page - it is not difficult, but there are a number of steps involved and, due to space limitations, I'll leave it up to you to read up on this feature on your own.

I found it necessary to install the "beep" package to get the alarm working on my systems (aptitude-install beep). I also had to open alsamixer (in a terminal - alsamixer) and unmute the "beep" control. Your mileage may, of course, vary - so try the method recommended in the wmpinboard man page before

you go to any further trouble.

Also covered in the man page is the use of alternate fonts. The default font seems just fine to me, but you can specify a different font by adding the --font=FONT option to your startup string (either in the menu, or by editing the dockapp "settings" from your desktop once you've launched the application). You will need to consult the man page for further details on this, as well, since there are some restrictions on the type and size of font you can use (mainly due to space limitations in a 64x64-pixel area).

Conclusion

wmpinboard is a useful tool for those whose "real-world" desks become cluttered with notes scribbled on pieces of paper - whether of the sticky variety or otherwise. The application allows you to keep up to twenty notes at any given time, which should be more than enough for most people - especially with some periodic culling of the note "herd." The alarm function is especially nice, but use it carefully and READ THE...(well, you know)...before you start setting up alarms!

Liquid Lemur Window Maker comes with wmpinboard pre-installed - but not activated by default. Non-Lemurs running a Debian installation can obtain wmpinboard from the Squeeze (stable), Wheezy (Testing) or Sid (Unstable) repositories (aptitude install wmpinboard in your terminal emulator of choice).