OpenID | WebKit | Dojo | RDFa | Quanta Plus | ZenPhoto

LINUX JOURNAL

Since 1994: The Original Magazine of the
JULY 2008 | ISSUE 171

Using Dojo
The Industrial-Strength JavaScript Toolkit

DIGITAL TV TUNER CARDS ON LINUX *REVIEWED*

Get Started with OpenLaszlo

Qt and WebKit
Create Application Opportunities

QUANTA PLUS TUTORIAL
The Perfect Web Development Tool

MOVING TO A SOLID STATE DRIVE
Performance Comparison

WordPress Founder
and Open-Source Crusader
MATT MULLENWEG

\+ DESKTOP LINUX and the Enterprise

www.linuxjournal.com
$5.99US $5.99CAN
07
0 09281 03102 4

AMD Opteron 64

THE QUIET REVOLUTION

32 processor performance in your desktop

HiPerSTATION 8000

System Highlights:

- Optimal multi-threaded application performance utilizing (8) eight native Quad-Core AMD Opteron™ processors
- Up to 256GB of DDR2-667 memory
- Next generation dual x16 PCIe technology offers scalable graphic bandwidth
- 2TB of storage
- Designed to be quiet enough to utilize in an office environment; ⅓ the noise of server version
- Ideal for usage in Computer Aided Designs, Visualization, and Digital Content Creation

HPC SYSTEMS INCORPORATED

www.hpcsystems.com 888-725-3472

AMD Solution Provider PLATINUM

GSA GS-35F-0596R Contract Holder

SBA 8(A) CERTIFIED

```
% ar x "a trouble free network"
ar: a trouble free network does not exist

% "How is Avocent geared to helping me create one?
Unmatched."

% ^What if we decide to do nothing?
Bad substitute

_
```

Even your *nix wants Avocent.

ACS 6000 Series Console Servers

- Simplify management with auto-discovery of target devices
- Optimize availability with multiple access paths
- Increase security through access levels
- Reduce set up time with software-controlled Cisco® pin-outs

At Avocent, we realize that as Unix and Linux® users, you like to do things your own way. You like to use CLI more than a GUI, you like to use Firefox® more than "the other one." Heck, some of you even like to set up a few servers at home to develop your own applications.

That's why our software solution allows you to use your CLI of choice, access your Unix or Linux® boxes from anywhere and run whatever browser you want. We wanted it to be flexible enough for you to do things **your way.**

Avocent®

To learn more, go to **www.avocent.com/acs**

Avocent and the Avocent logo are trademarks or registered trademarks of Avocent Corporation or its affiliates in the U.S. and other countries. All other marks are the property of their respective owners. Copyright © 2008 Avocent Corporation. All rights reserved.

CONTENTS

JULY 2008
Issue 171

FEATURES

50 KEEP ON BLOGGING IN A FREE WORLD
Matt Mullenweg gives us a peek behind the WordPress curtain. We find out what's new in blogging, how WordPress evolves, and his feelings about open source.
Katherine Druckman

54 USING WEBKIT IN YOUR DESKTOP APPLICATION
Blurring the lines between the desktop application and the Web.
Benjamin Meyer

60 DOJO: THE JAVASCRIPT TOOLKIT WITH INDUSTRIAL-STRENGTH MOJO
Dojo isn't just a JavaScript toolkit, it's the industrial-strength JavaScript toolkit.
Matthew Russell

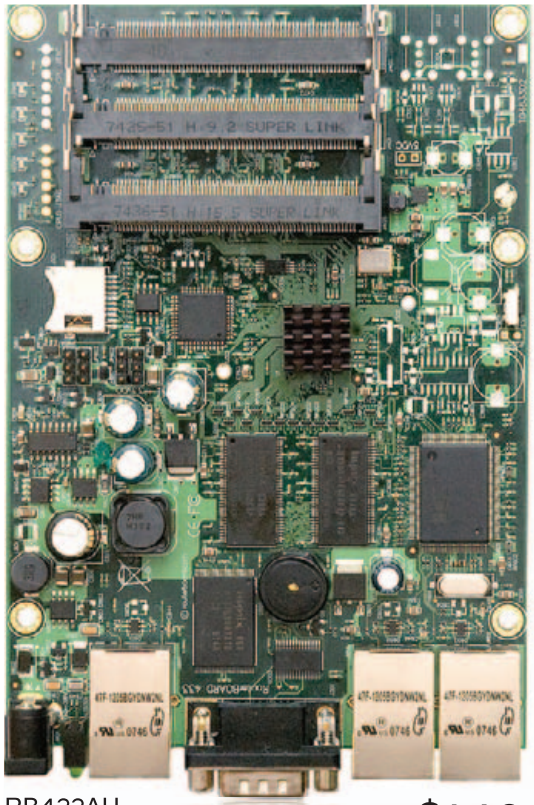
70 INTRODUCING OPENLASZLO 4
OpenLaszlo's goal is to add desktop-like functionality to browser-based applications.
Paul Barry

ON THE COVER
- Get Started with OpenLaszlo, p. 70
- Quanta Plus Tutorial—the Perfect Web Development Tool, p. 92
- Moving to a Solid State Drive—Performance Comparison, p. 84
- Using Dojo—the Industrial-Strength JavaScript Toolkit, p. 60
- Digital TV Tuner Cards on Linux—Reviewed, p. 44
- Qt and WebKit Create Application Opportunities, p. 54
- WordPress Founder Matt Mullenweg, p. 50
- Desktop Linux and the Enterprise, p. 88

Photos of Matt Mullenweg by Richard Wheeler
www.richardwheelerphotography.com

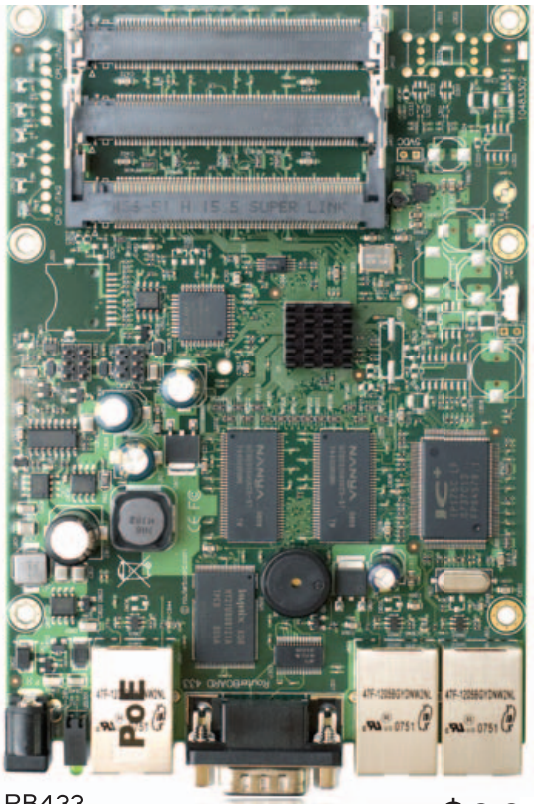
RouterBOARD 433

680MHz MIPS (overclock to 800MHz)



RB433AH $149

300Mhz



RB433 $99

The RB433 is a high speed AP/router. It has a new generation 300Mhz Atheros CPU and 64MB of RAM. It is provided with three 10/100Mbit Ethernet ports and three miniPCI slots.

The RB433AH is a faster, enhanced version of the RB433 with a 680Mhz CPU, 128MB RAM and a microSD card slot for an additional memory card. It supports overclocking to 800Mhz, pushing the price/performance ratio to new standards.

These devices are preinstalled with the MikroTik RouterOS Firewall/QoS/Routing operating system. RB433 comes with a L4 license, and RB433AH with L5

www.routerboard.com

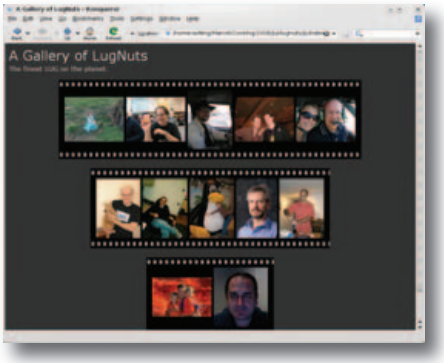
CONTENTS

JULY 2008
Issue 171

COLUMNS

20 REUVEN M. LERNER'S AT THE FORGE
Integrating OpenID

24 MARCEL GAGNÉ'S COOKING WITH LINUX
You Look Marvelous on the Web!



30 DAVE TAYLOR'S WORK THE SHELL
Of Movies, Trivia Games and Twitter

32 MICK BAUER'S PARANOID PENGUIN
Customizing Linux Live CDs, Part III

36 KYLE RANKIN'S HACK AND /
Migrate to a New Hard Drive

96 DOC SEARLS' EOF
A Tale of Two Futures

REVIEW

44 OVER-THE-AIR DIGITAL TV WITH LINUX
Alolita Sharma

IN EVERY ISSUE

10 LETTERS
14 UPFRONT
38 NEW PRODUCTS
40 NEW PROJECTS
81 ADVERTISERS INDEX

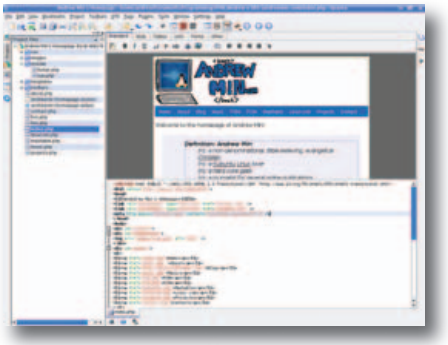
INDEPTH

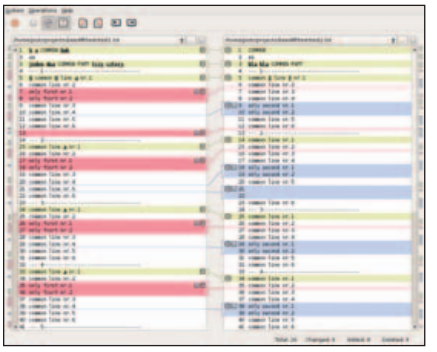
78 SEMANTIC WEB PUBLISHING WITH RDFA
The why and how of using RDFa to add semantics to your site.
Golda Velez

84 MY MOVE TO SOLID STATE
Is a solid state drive worth it? In this article, Kyle Rankin pits a 1.8" 4200rpm drive against an SSD in a series of real-world Linux tests.
Kyle Rankin

88 LINUX AND THE ENTERPRISE DESKTOP: WHERE ARE WE TODAY?
Can Linux finally make inroads onto the enterprise desktop?
James Gray

92 HOW TO USE QUANTA PLUS, THE WEB DEVELOPER TOOL WITH EVERYTHING BUT THE KITCHEN SINK
A tutorial on doing it all with Quanta Plus, the open-source Web IDE.
Andew Min





40 BEEDIFF

Next Month

COOL PROJECTS

Next month, we'll feature some how-to articles that'll get you through the dog days of summer. We'll show you how to fake a UFO landing with Voodoo, how to build your own 16 Terabyte fileserver, how to turn a spare USB stick into a sysadmin's dream with Billix, and how to get X running on a Gumstix embedded device with an E-Ink display.

As always, there's much more. We'll have a review of the Cradlepoint PHS300 (personal hotspot), and we'll show you how to connect your wiimote to a computer and use it as a mouse or wireless controller for any game on your system. And last but not least, we'll take a look at hacking the BUG from Bug Labs.

USPS *LINUX JOURNAL* (ISSN 1075-3583) (USPS 12854) is published monthly by Belltown Media, Inc., 2211 Norfolk, Ste 514, Houston, TX 77098 USA. Periodicals postage paid at Houston, Texas and at additional mailing offices. Cover price is $5.99 US. Subscription rate is $29.50/year in the United States, $39.50 in Canada and Mexico, $69.50 elsewhere. POSTMASTER: Please send address changes to *Linux Journal*, PO Box 980985, Houston, TX 77098. Subscriptions start with the next issue. Canada Post: Publications Mail Agreement #41549519. Canada Returns to be sent to Bleuchip International, P.O. Box 25542, London, ON N6C 6B2

OUTRAGED
by the high cost of Fibre Channel or iSCSI Storage ?

Switch to AoE !

SR2461

Coraid Offers a Complete Line of Clustered Modular Storage Products:

- High Performance EtherDrive® SATA+RAID Storage Appliances with 1 GigE or 10 GigE Connections
- Clustered VirtualStorage™ Appliances *(a Revolutionary Logical Volume Manager)*
- Scalable NAS Gateways *(File Sharing with EtherDrive® Storage)*

Coraid's EtherDrive® Storage with AoE is fast, reliable disk storage that's easy to use. And it's much more affordable than iSCSI or Fibre Channel!

- Coraid products use open AoE (ATA-over-Ethernet) block storage protocol, for high performance without the TCP/IP overhead
- With AoE, your shared storage capacity is infinitely scalable – at a fraction of the cost of iSCSI or Fibre Channel storage
- We provide a 3-year warranty and free firmware upgrades on all our products, as well as support from first-rate engineers trained in our technology

Product Excellence Award Winner LINUXWORLD CONFERENCE & EXPO Best Storage Solution

EtherDrive® Storage has a field-proven track record and is 1000+ large data storage customers strong.

CORAID

The Linux Storage People

To learn more about this and Coraid's other products, go online or call

+1.706.548.7200 (toll free: **877.548.7200**)

www.coraid.com

Get the best value when you World's #1 Web Host!

1&1 is both the world's biggest web host, and the fastest growing. Our global community is over 7 million people strong. We are financially stable, profitable, and look forward to hosting your website for many years to come.

Your complete website solution

Whatever your Internet requirements, from personal websites to dedicated servers and e-commerce sites for businesses, 1&1 has a complete solution to fit your needs. The powerful, intuitive 1&1 Control Panel is web-based, giving you full control of your information from any Internet-connected computer with no software to install. Update your domains, modify all elements of your site, and even upgrade your 1&1 package, all with one password and one login.

FREE domains!

Sign up for one of our Web Hosting packages and receive up to 5 domain names FREE! You don't need an extra registrar!

All-inclusive prices, unbeatable value

Other companies promise affordable web hosting, but then charge extra for setup fees, higher renewal rates, or promotional services. With 1&1, the listed price is the number you'll pay, and you can expect a fully loaded, comprehensive suite of web services.

Advanced Data Centers

1&1 has spent $25 million to provide the ultimate home for your valuable information. Surpassing all standards of safety, power, connectivity, hardware, performance, and security, our Data Centers represent the future of information storage. 1&1's superior technology delivers reliable service for your website.

90-Day Money Back Guarantee

We take pride in our reputation for reliability. If you're not completely happy within 90 days from placing an order with 1&1, you will receive your money back – no questions asked.

MEMBER OF united internet

Call 1.877.go1and1

choose the

.info
Domain only
$0.99
first year!*

Best Value:
Compare for yourself.

	1&1 Home	Go Daddy DELUXE	Hostway GOLD
Included Domains (.com, .net, .org, .info or .biz)	2 Valued at up to $13.98/year!	$1.99/year	$7.95/year
Web Space	120 GB	150 GB	12 GB
Monthly Transfer Volume	1,200 GB	1,500 GB	250 GB
E-mail Accounts	1,200 IMAP or POP3	500 POP3	250 POP3
Mailbox Size	2000 MB	500 MB	75 MB
Website Builder	12 Pages	Additional $8.09/month	✓
Access to Open Source Application Library	✓	✓	–
Starter Software Suite	✓	–	–
Search Engine Submission	✓	–	–
90-Day Money Back Guarantee	✓	–	–
Support	24/7 Toll-free Phone, E-mail	24/7 Phone, E-mail	24/7 Toll-free Phone, E-mail
Price Per Month	~~$4.99~~	$6.64	$13.95

3 months FREE
on the 1&1 Home Package!**
Offer ends 6/30/2008!

"Over the past 3 years, I've had 1&1 as a hosting solution for all my needs. Not only have they been extremely reliable, but fast also. And when considering value, 1&1 offers the best solution for the lowest price hands down. You just can't beat it!"

Nick Jagodzinski, www.blindroutine.com

© 2008 1&1 Internet, Inc. All rights reserved.

Visit 1and1.com for details. Prices based on comparable Linux web hosting package prices, effective 4/24/2008.
* Price valid for first year of .info domain registration. After the first year, regular prices will apply.
** Offer valid for Home Package only, 12 month minimum contract term required. Home Package special expires 6/30/2008. Product and program specifications, availability, and pricing subject to change without notice. All other trademarks are the property of their respective owners.

Visit us now **1and1.com**

1&1

LINUX JOURNAL™
Since 1994: The Original Magazine of the Linux Community

Digital Edition Now Available!

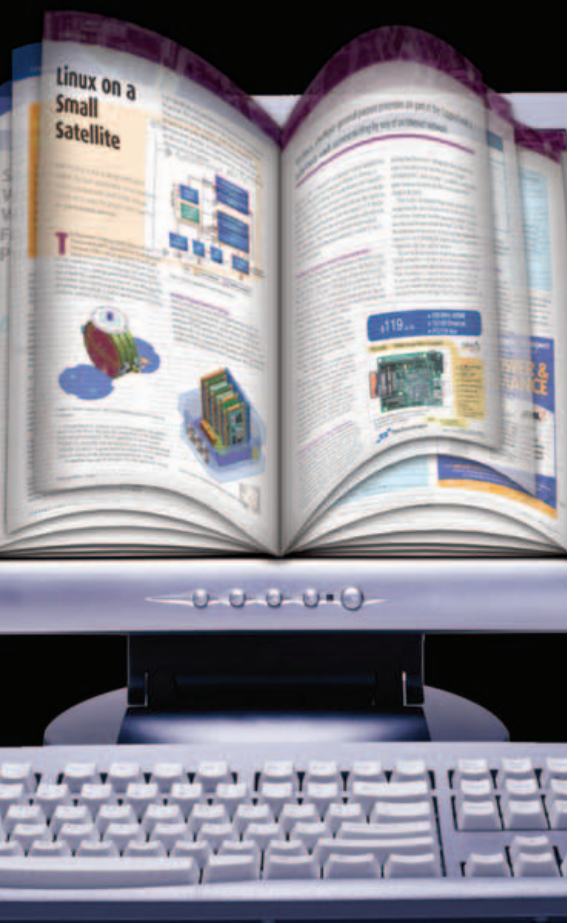
Read it first
Get the latest issue before it hits the newsstand

Keyword searchable
Find a topic or name in seconds

Paperless archives
Download to your computer for convenient offline reading

Same great magazine
Read each issue in high-quality PDF

Try a Sample Issue!
www.linuxjournal.com/digital



LINUX JOURNAL

Executive Editor	Jill Franklin jill@linuxjournal.com
Senior Editor	Doc Searls doc@linuxjournal.com
Art Director	Garrick Antikajian garrick@linuxjournal.com
Products Editor	James Gray newproducts@linuxjournal.com
Editor Emeritus	Don Marti dmarti@linuxjournal.com
Technical Editor	Michael Baxter mab@cruzio.com
Senior Columnist	Reuven Lerner reuven@lerner.co.il
Chef Français	Marcel Gagné mggagne@salmar.com
Security Editor	Mick Bauer mick@visi.com

Contributing Editors
David A. Bandel • Ibrahim Haddad • Robert Love • Zack Brown • Dave Phillips • Marco Fioretti
Ludovic Marcotte • Paul Barry • Paul McKenney • Dave Taylor • Dirk Elmendorf

Proofreader Geri Gale

Publisher	Carlie Fairchild publisher@linuxjournal.com
General Manager	Rebecca Cassity rebecca@linuxjournal.com
Director of Sales	Laura Whiteman laura@linuxjournal.com
Regional Sales Manager	Joseph Krack joseph@linuxjournal.com
Regional Sales Manager	Bruce Stevens bruce@linuxjournal.com
Circulation Director	Mark Irgang mark@linuxjournal.com
System Administrator	Mitch Frazier sysadm@linuxjournal.com
Webmistress	Katherine Druckman webmistress@linuxjournal.com
Accountant	Candy Beauchamp acct@linuxjournal.com

Linux Journal **is published by, and is a registered trade name of, Belltown Media, Inc.**
PO Box 980985, Houston, TX 77098 USA

Reader Advisory Panel
Brad Abram Baillio • Nick Baronian • Hari Boukis • Caleb S. Cullen • Steve Case
Kalyana Krishna Chadalavada • Keir Davis • Adam M. Dutko • Michael Eager • Nick Faltys • Ken Firestone
Dennis Franklin Frey • Victor Gregorio • Kristian Erik • Hermansen • Philip Jacob • Jay Kruizenga
David A. Lane • Steve Marquez • Dave McAllister • Craig Oda • Rob Orsini • Jeffrey D. Parent
Wayne D. Powel • Shawn Powers • Mike Roberts • Draciron Smith • Chris D. Stark • Patrick Swartz

Editorial Advisory Board
Daniel Frye, Director, IBM Linux Technology Center
Jon "maddog" Hall, President, Linux International
Lawrence Lessig, Professor of Law, Stanford University
Ransom Love, Director of Strategic Relationships, Family and Church History Department,
Church of Jesus Christ of Latter-day Saints
Sam Ockman
Bruce Perens
Bdale Garbee, Linux CTO, HP
Danese Cooper, Open Source Diva, Intel Corporation

Advertising
E-MAIL: ads@linuxjournal.com
URL: www.linuxjournal.com/advertising
PHONE: +1 713-344-1956 ext. 2

Subscriptions
E-MAIL: subs@linuxjournal.com
URL: www.linuxjournal.com/subscribe
PHONE: +1 713-589-3503
FAX: +1 713-589-2677
TOLL-FREE: 1-888-66-LINUX
MAIL: PO Box 980985, Houston, TX 77098 USA
Please allow 4–6 weeks for processing address changes and orders
PRINTED IN USA

LINUX is a registered trademark of Linus Torvalds.

servers DIRECT

MORE PRODUCTS, BETTER SERVICE, GUARANTEED.

YOUR HIGH PERFORMANCE COMPUTING SOLUTION HAS ARRIVED

The ServersDirect system with the Intel® Xeon® Processor helps you simplify computing operations, accelerate performance and accomplish more in less time.

GO STRAIGHT TO THE SOURCE!

1.877.727.7887 | www.ServersDirect.com

SDR-I1200-T00

* 1U Rackmount Server with 350W Power Supply
* Intel® 3000 Chipset
* Dual/Quad-Core Intel® Xeon® processor 3000 sequence
* Support up to 8GB unbuffered ECC/ non-ECC DDR2 677/533 SDRAM
* 2x 3.5" Internal Drive Bay
* Dual Gigabit Ethernet connections

SDR-C1300-T04

* 1U Rackmount Server with 520W Power Supply
* Intel® 5000V Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 16GB DDR2 667 & 533 FB-DIMM
* 4 x 3.5" Hot-swap SATA Drive Bays
* Dual-port Gigabit Ethernet Controller

SDR-S1312-S04

* 1U Rackmount Server with 650W Red High-efficiency Power Supply
* Intel® 5000P Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 32GB DDR2 667 & 533 FB-DIMM
* 4 x 3.5" Hot-swap SATA Drive Bays
* Dual-port Gigabit Ethernet Controller

SDR-C2300-T06

* 2U Rackmount Server with 600W Power Supply
* Intel® 5000V Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 16GB DDR2 667 & 533 FB-DIMM
* 6 x 3.5" Hot-swap SATA Drive Bays
* Dual-port Gigabit Ethernet Controller

SDR-S2301-T08

* 2U Rackmount Server with 700W High-Efficiency Redundant Power Supply
* Intel® 5000P Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 64GB DDR2 667 & 533 FB-DIMM
* 8 x 3.5" Hot-swap Drive Bays
* Dual-port Gigabit Ethernet Controller

SDR-C3300-T08

* 3U Rackmount Server with 600W Power Supply
* Intel® 5000P Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 32GB DDR2 667 & 533 FB-DIMM
* 8 x 3.5" Hot-swap SAS Drive Trays with SES2
* Dual-port Gigabit Ethernet Controller

SDR-S3300-U16

* 3U Rackmount Server with 800W Redundant Power Supply
* Intel® 5000P Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 64GB DDR2 667 & 533 FB-DIMM
* Dual-Channel U320 SCSI with 16 (8+8) Hot-swap Drive
* Dual-port Gigabit Ethernet Controller

SDR-S4302-U08

* 4U Rackmountable / Tower w/ 800W High Efficiency Power Supply
* Intel® 5000P Chipset
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 64GB DDR2 667 & 533 FB-DIMM
* Dual-Channel U320 SCSI with 8 (4+4) Hot-swap Drive
* Dual-port Gigabit Ethernet Controller

SDR-I6301-T06

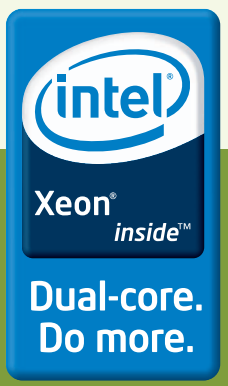
* 6U Rackmount with 550W Power Supply
* Intel® Chipset 5000V
* Dual Intel® 64-bit Xeon® Quad-Core or Dual-Core
* Support up to 16GB DDR2 667 & 533 FB-DIMM
* 6 x 3.5" Hot-swap SATA Drive Bays
* Dual-port Gigabit Ethernet Controller

SERVERS DIRECT CAN HELP YOU CONFIGURE YOUR NEXT HIGH PERFORMANCE SERVER SYSTEM - CALL US TODAY!

Our flexible on-line products configurator allows you to source a custom solution, or call and our product experts are standing by to help you assemble systems that require a little extra. Servers Direct - your direct source for scalable, cost effective server solutions.

1.877.727.7887 | www.ServersDirect.com

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, Pentium, and Pentium III Xeon are trademarks of Intel Corporation or it's subsidiaries in the United States and other countries.



letters

Puppy Love

I just finished reading the article on Puppy Linux [*LJ*, April 2008]. I'm glad to see you introduce this distribution to your readers. I discovered PL about a year and a half ago. Every year, my wife and I travel for about six months, usually in our RV. During the 2007 travel period, I used PL exclusively to use the Internet safely. I found no reason to look at any other distribution. I wholeheartedly recommend it to anyone who wants the flexibility and security of using an operating system on a Flash drive. My version of PL includes Firefox, OpenOffice.org and The GIMP.

I am not sure it was clear from the article, but because the PL OS is loaded anew into the computer memory at each bootup and runs from that memory, any possible corruption of the OS by an on-line attack probably would last only until the computer is turned off. Next boot, fresh OS.

--

Louis Benton

Forgotten gThumb

In the March 2008 issue of *LJ*, in the "Desktop Must-Haves" article, author Dan Sawyer seemed to have forgotten about gThumb as a photo importer and organizer for the GNOME desktop environment, which also allows for photo import (using the PTP protocol), supports slideshows, as well as providing a limited array of image manipulation tasks (balance, contrast, transformation, crop, red-eye removal and so on). It is pretty much standard with many GNOME installations, and yet he didn't mention it, favoring the rather "controversial" F-Spot (due to Mono and its status regarding things such as Windows.Forms and so forth).

Generally, I like native applications better (due to the look and feel), but I do agree with Mr Sawyer regarding all the applications he reviews in this article, with the only exception being gThumb, which I think deserved to be mentioned.

--

Gian Paolo Mureddu

Dan Sawyer replies: *Quite honestly, Linux is a* big *software universe, and I'd not run into gThumb before I got your letter (it did not, alas, come standard with any of my GNOME installations). I haven't had time to do a proper assessment yet, but it looks very promising. Thanks for the recommendation!*

As for the controversiality of Mono, I make it a point to stay as far away as possible from the infighting between various licensing and project camps. Although I certainly have opinions on which toolkits work best consistently, when it comes down to it, I care about the functionality. If that functionality is coming from a Mono codebase or a (until recently) Java codebase over/against another, less controversial toolkit, and it saves to data formats that are easily translatable and/or universally readable, then I have no quarrel with it.

Thank you for the letter. I'm pleased you liked the article!

On Security in General

This letter is not related directly to *LJ*, but as a magazine involved with Internet security issues, I think some of the following reflections could be considered by the readers and the magazine editors who can include some article(s) and discussion(s) on this in the near future.

I am a professor at a university. I do research and I teach. I've used the Internet since my old student days, when we FTPed, Telneted, fingered and so forth. Those were free days, free as in speech, free as in open source, open as it was the Internet. But, then came the "worms", and we closed the doors. Later, we encrypted everything we sent, and built "walls of fire" and "military zones". Now, we filter everything that comes into or out of our nets—sometimes on security grounds, sometimes to reduce traffic jams, and sometimes because of copyright infringements.

In the past few years, the troubles created by these "policies" have been greatly affecting our work. Big institutions have created rules to close their doors without regarding who might be affected. Sometimes we cannot even send e-mail to some colleague because our domain (which can be as general as .xy!!) is on a blacklist.

The most ridiculous extreme occurred last week. I advise students in different institutions, and we interchange information, data and archives. At one of these institutions, the SSH port was moved to a number greater than 1024, at the other, all ports above 1024 were closed, even for client connections. These measures were taken without notifying the users. The result was wasting time trying to discover why what we always have done (until recently) does not work anymore, wasting time in adapting to the new situation, and wasting time having unfruitful discussions with the system managers.

The freedom to filter packets today is amazingly big, and the Internet gradually is becoming a mess of entangled knots instead of a fluid traffic Net.

We need standards—standards for security policies. We need to convince security managers that the best security measure is just to unplug from the Net, or maybe better, to switch off the computer! But this trivial solution, as usual, has no interest to anybody, even to them. I can

(hardly) do research without the Internet, but they will lose their jobs without it.

Security policies should be discussed with the end users who are, at last, the reason we have the Internet.

--
Guillermo Giménez de Castro

Parallels and VMware Fusion

Dave Taylor's article on Parallels and VMware Fusion was a welcome sight ["Running Ubuntu as a Virtual OS in Mac OS X" in the May 2008 issue of *LJ*]. I run Ubuntu 7.04 Server in Fusion on my MacBook, and it works great as a portable server environment. I also can rely on the Ubuntu software repository and get all the advantages of the Open Source world without cluttering up my Mac OS X install. Hopefully, the Linux in Fusion user base will grow over time, and VMware will implement more of the power-user features into its product. I would love to see a headless option that doesn't involve force-quitting the Fusion UI.

Are there any plans for more detailed articles in the future? Fusion in particular has some options (like port forwarding) that can be enabled only through config file editing.

--
Adam Backstrom

And, More on Dave Taylor's Mac Article

I really enjoyed this article. However, I did notice three things that I don't really agree with.

First and foremost to me is the statement in the first paragraph that Mac OS X is a Linux distro. This is wrong. Mac OS X is based on Darwin, which is a BSD variant. BSD is not Linux and vice versa. They are totally separate codebases, although there has been some cross-pollination.

Second, calling X11 "a tightly integrated version of the popular Linux windowing system" is a bit off-base. X11 is a UNIX windowing system, which originally was developed at MIT long before Linux ever was envisioned. The paragraph is not really wrong, it's just a bit misleading—at least as I read it.

Third, in the fifth paragraph, the author states, "Free operating systems (that is, anything but Microsoft Windows)...." There are many nonfree OS systems for Intel machines. Examples include OS/2 (okay, it is now dead), DR-DOS (also dead), Pick (not dead, but has a rather small market share—integrated OS/RDMS system) and Sun's Solaris (the commercial one). On non-Intel machines, most OSes are not at all free, such as z/VM, z/VSE, z/TPF, z/OS on IBM's "mainframe" System z, AIX on IBM's System p and i5/OS on IBM's System i. You may have noticed that I know a bit about IBM machines. I've worked on them, although not for IBM, since the mid-1970s.

--
John McKown

SILICON MECHANICS
visit us at www.siliconmechanics.com
or call us toll free at 866-352-1173

For more information about the iS707 iSCSI/NAS storage appliance visit www.siliconmechanics.com/707.

Expert included.

Kevin is a software installation engineer at Silicon Mechanics. He builds the tools that ensure your servers have an automated, scalable installation of BIOS, firmware, and software. Lately he's been working with the new Silicon Mechanics Storform iS707 iSCSI/NAS storage appliance. Kevin is impressed with the iS707's capabilities, including file sharing, storage consolidation, backup, recovery, and replication. He likes the 24 high-capacity SATA disk drives which can store up to 24TB, and the performance of its industry-leading, hardware-based RAID.

Kevin knows that this storage appliance offers outstanding performance for the price. But Kevin is a software guy, so what really has his attention is the powerful, intuitive, web-based management interface, powered by Open-E. The Storform iS707 boasts management, tuning, consolidation, and data protection features typically reserved for higher-priced products.

When you partner with Silicon Mechanics, you get more than a feature-rich storage solution — you get an expert like Kevin.

Silicon Mechanics and the Silicon Mechanics logo are registered trademarks of Silicon Mechanics, Inc.

Dave Taylor replies: *Oh jeez, sometimes I don't know how these gremlins get into the computer and mess up my perfectly written articles. I mean, really, I might have accidentally said that in my original piece as submitted, but it's clearly wrong and I know it! Mac's Darwin roots are NEXTSTEP, which itself was based on Mach 2.5 and 4.3BSD. Heck, I contributed to 4.3BSD! As you point out, X11 comes from the MIT Athena Project, and was released years before Linux was even a dream. Mea culpa on both of 'em.*

You gotta cut me some slack on the comment about other nonfree operating systems for the Intel architecture, however. I was trying to be a bit wry and sarcastic in my commentary. Of course, there are many commercial operating systems that, outside of illegal P2P copies, are licensed and tightly monitored, including the systems you mention and many more.

Suffice to say, we let a few gaffes slip through and apologize for any confusion they caused. Glad you enjoyed the article. We'll get our facts straight next time, I promise.

More Gremlins Attack Dave

I'm still not sure how Dave Taylor positions his column in *Linux Journal*. It probably must be meant as a column for the pros—some kind of "who finds the bugs I smuggled in" thing. Surely it can't be for beginners who'd get frustrated by all the code that does not work the way the text makes you believe.

In his May 2008 column, Dave wants to give us advice on error handling and making scripts bulletproof, again without checking his own code snippets for errors.

The `2>&1 >/dev/null` output redirection will not work as described, because first, STDERR is redirected to where STDOUT is (currently still) wired to, and then STDOUT is sent to data nirvana, but redirected STDERR will not follow suit. The `>&1` redirection does not mean "pass it on to STDOUT" but rather "rewire yourself to where STDERR is right now". There are multiple possibilities to do it right, the most often used is `>/dev/null 2>&1`. This works because first STDOUT is plugged in to the "data store with endless capacity", and only then is STDERR told to put its hose into the same bucket.

--
Kurt Keller

Dave Taylor replies: *Jeez, must be gremlins-attack day or something. Yeah, you're right that the order of metacharacters in that particular line is wrong. Thanks for pointing it out!*

Debian Live

I just got a chance to read the May 2008 issue of *LJ*, and I wanted to write with respect to the article "Customizing Linux Live CDs, Part I". It is a nice article and covers similar techniques I used long ago when remastering Knoppix (I remastered only if I needed something beyond the knoppix.sh injection model). However, as the article is discussing Debian-based distributions, I think it only fair to mention Debian Live, which I and many others use to make live CDs of Debian. With Debian Live, making a custom live CD is far easier than the remastering described in the article. I think it would be worth *LJ* readers' time (remastering, that is) to take a look at Debian Live:

- Debian Live: **debian-live.alioth.debian.org**
- Debian Live Download Server: **live.debian.net**
- Debian Live Wiki: **wiki.debian.org/DebianLive**
- Debian Live irc—channel #debian-live on **irc.oftc.net**

--
Richard Nelson

Mick Bauer replies: *On the one hand, this series is intentionally Ubuntu-centric, and for Ubuntu fans, being able to customize one's favorite distro is worth learning a little command-line voodoo. It's also, I think, a good way to illustrate how to use compressed loopback filesystems.*

But, you're right. I'd be remiss if I didn't at least mention a simpler way to achieve a similar thing! So, in Part III of this article [see page 32], I mention Debian Live and cite the link to their Wiki (which includes ample links to downloads and so forth). Thanks for bringing it to my attention.

Good News

Mick Bauer's "Customizing Linux Live CDs, Part I" (*LJ*, May 2008) was a great article, and the timing was perfect (for me...and it's all about me, right?).

A buddy and I have been playing around with bootable-USB sticks using different distros. Ideally, we want a fully functional desktop OS that we literally can take with us anywhere. There are lots of apps we want that are not on the live CD. Since you (and pendrivelinux) have done the heavy lifting for us, setting up the remastered Ubuntu USB stick was a breeze. We're not quite done tweaking yet, but our current image is approaching 1.4GB. The final version will live on a 4GB stick, but a valuable side benefit is that the 2GB Flash drive I'm using for testing this will be passed around the office for people to give Linux a test-drive.

So, with a stroke of the pen, you've not only provided tremendous value for my subscription dollars, but you've also increased the ranks of Linux users!

Oh, I also like the line numbering scheme you used for your scripts.

--
Darrin Auxier

Mick Bauer replies: *Wow, what a thoughtful, gratifying message! It gave me a boost just as I was wondering if and how I'll make deadline for the next issue. It makes a difference, being reminded that people actually do find this stuff to be useful. (Usually, I just hear about the parts I get wrong!)*

Skype vs. Gizmo

The article on podcasting by Dan Sawyer in the May 2008 issue of *LJ* was of particular interest to me, and it confirmed that recording calls using Skype on Linux is a nontrivial issue. (I interview genre authors on my podcast, Radio Free Bliss.)

However, to say that "[t]here are a number of packages [that hijack the DSP with a middleware layer] that'll do

this—for a fee—on Windows and Mac" is not strictly true. Driven from Linux, I use PowerGramo with Skype on Windows, the basic (and very functional) version of which is free. I've had no problems using it.

As to why I choose to use Skype: well, most nontechnical people know the Skype name much better than they know Gizmo. And, for every ten people I've asked who have Skype, there are none who have Gizmo. It would be very arrogant of me to demand that my guests sign up for a completely new service, all for the sake of one 45-minute conversation. So, even though my main machine is Linux, running Mandriva, I keep a Windows machine around for podcast purposes. I fear, especially among the less technical, that it's going to be a Skype-Win world for the forseeable future.

--
KS Augustin

Dan Sawyer replies: *Thanks for the correction and the additional information. I too tend to do my Skyping on Windows, even though I actually record the calls on Linux. I do this because all my Linux boxen are 64-bit systems, and Skype, as yet, doesn't particularly play nice with 64-bit. Plus, running it on an emulation layer can get a bit twitchy. One of these days, it'll come out for 64-bit distros. Until then, I'll be using my Windows machine as a conference-call PBX.*

Such is life, sometimes.

Go Green Makes Reader See Red

Having read "Go Green, Save Green with Linux" in the April 2008 issue of *LJ*, I got red. James Gray spouting "our fragile planet's inability to support an SUV-lifestyle" is nonsense.

The planet will adapt. If the planet doesn't like what man is doing, then it will wipe him out. The human race is just a blink in time for this planet. It is a selfish attitude of personal survival that drives this fascist mindset.

"Mother Nature's Mayday" is a farce, or skillfully exploited situation. It is just a humanistic perspective applied to generate a human emotional response.

"Mother Nature" has no qualms, or an uneasy feeling or pang of conscience, as to conduct or compunction, about life and death.

The bottom line of this article is about the "bottom line". People are frustrated with wasting money on inefficient products.

--
Stephen Baker

James Gray replies: *Thank you for your reply. I appreciate your reading the article and value your feedback.*

Your point about the Earth "caring" whether humans survive or not is well taken. In the grand scheme of things, we are merely one small part of a resilient and dynamic natural system that doesn't choose its victims indiscriminately.

On the other hand, I also hope you will accept my writing "Nature's Mayday calls" for the metaphor that it is. Here, my intent was to illustrate how the planet is giving us clear feedback that our actions are causing drastic and perhaps permanent change to natural systems. Furthermore, although you appear to believe that humans should just act however they will and face the consequences, I personally feel that we humans have a moral obligation to treat our Earth home with utmost respect. I think it is in our enlightened self-interest to protect not only those natural systems that sustain us, but also to not adversely affect the results of billions of years of wondrous evolution.

Evolutionary biologists say that a sense of morality is hard-wired into our genes. I am surprised you would lump me together with Hitler simply for writing that my own moral compass leads me to convince others that better natural resource management is a positive thing.

Finally, though you dispute my point about the Earth's inability to support an SUV lifestyle for billions, please note that this assertion has been proven empirically in several studies. There are simply not enough resources for all six billion-plus humans to enjoy our level of material consumption. Please contact me if you would like to receive more information about these studies. Thanks again for your feedback.

LINUX JOURNAL

At Your Service

MAGAZINE

PRINT SUBSCRIPTIONS: Renewing your subscription, changing your address, paying your invoice, viewing your account details or other subscription inquiries can instantly be done on-line, **www.linuxjournal.com/subs**. Alternatively, within the U.S. and Canada, you may call us toll-free 1-888-66-LINUX (54689), or internationally +1-713-589-3503. E-mail us at subs@linuxjournal.com or reach us via postal mail, Linux Journal, PO Box 980985, Houston, TX 77098-0985 USA. Please remember to include your complete name and address when contacting us.

DIGITAL SUBSCRIPTIONS: Digital subscriptions of *Linux Journal* are now available and delivered as PDFs anywhere in the world for one low cost. Visit **www.linuxjournal.com/digital** for more information or use the contact information above for any digital magazine customer service inquiries.

LETTERS TO THE EDITOR: We welcome your letters and encourage you to submit them at **www.linuxjournal.com/contact** or mail them to Linux Journal, 1752 NW Market Street, #200, Seattle, WA 98107 USA. Letters may be edited for space and clarity.

WRITING FOR US: We always are looking for contributed articles, tutorials and real-world stories for the magazine. An author's guide, a list of topics and due dates can be found on-line, **www.linuxjournal.com/author**.

ADVERTISING: *Linux Journal* is a great resource for readers and advertisers alike. Request a media kit, view our current editorial calendar and advertising due dates, or learn more about other advertising and marketing opportunities by visiting us on-line, **www.linuxjournal.com/advertising**. Contact us directly for further information, ads@linuxjournal.com or +1 713-344-1956 ext. 2.

ON-LINE

WEB SITE: Read exclusive on-line-only content on *Linux Journal*'s Web site, **www.linuxjournal.com**. Also, select articles from the print magazine are available on-line. Magazine subscribers, digital or print, receive full access to issue archives; please contact Customer Service for further information, subs@linuxjournal.com.

FREE e-NEWSLETTERS: Each week, *Linux Journal* editors will tell you what's hot in the world of Linux. Receive late-breaking news, technical tips and tricks, and links to in-depth stories featured on **www.linuxjournal.com**. Subscribe for free today, **www.linuxjournal.com/enewsletters**.

diff -u

WHAT'S NEW IN KERNEL DEVELOPMENT

Linus Torvalds has called on all **Cogito** users to switch over to using **git** natively. When Linus first created git, he envisioned a tool that would provide a clean set of very low-level data-tracking actions. There was no point, he felt, in writing a bunch of complex, high-level revision control features, especially considering his existing penchant for system internals. The git tool was to be the "system call" layer, on top of which people could script their own favorite feature set. Cogito was the first attempt at such a wrapper, and for a while, it was the only way for people to use git without having to have a deep understanding of git's data-tracking concepts. But nowadays, git has added its own set of porcelain commands on top of the deeper plumbing. People still can use the lower-level commands to script their favorite front-end interface, but git's native high-level interface is likely to stay more up to date than any of those scripts, because it is so integrated into the original tool. Cogito itself, as Linus points out, has not been maintained for more than a year, and he says it's time for everyone just to switch over. This also will, he points out, have the added benefit that any problems uncovered by users will be debugged and fixed more easily, as the developers won't have to worry about which tool might have the bug.

Kernel development is always much messier than application development. When an application crashes, you can start to debug the core file or restart the application immediately for another test. When the kernel crashes, you typically have to reboot, which can be time-consuming. Kernel developers always are looking for ways to restart a test quickly or to identify potential bugs without actually having to experience the bad results of running into them. One technique that's been available for a while is to run the kernel as a user application itself on an already-running system, so that if the userland instance crashes, the rest of the system remains intact. That is so cool! Another attempt has recently come from **Thomas Gleixner**. His concept is to insert a whole new layer of debugging code in the kernel (it would be disabled for folks who just want to use their system) to track what happens to RAM after it has been allocated for clusters of variables. If the data suddenly changes when it wasn't supposed to, Thomas' code sees this as a big red flag and logs it. So, instead of discovering a bug because the entire system locks up, developers now, in many cases, can discover bugs before any symptoms affect the user experience. It's possible that Thomas will merge his code with something **Chris Mason** did a while ago. In Chris' project, a background thread would allocate memory, mark it with data, and then check periodically to see whether it had been corrupted. Thomas' and Chris' approaches are different and complementary in a couple different ways. For one thing, Thomas' code interacts with actual kernel objects that are currently in use, while Chris' checks an unrelated block of memory. Also, Thomas' code performs its checks after specific kernel operations, while Chris' does its checks after planned intervals of time. Both are good techniques, and it's likely that Thomas' code will tend to gather additional features over time.

Some kernel folks are considering alternatives to the **MAINTAINERS file**. **Krzysztof Halasa** has suggested the file is no longer necessary and should be replaced by formatted comments in the code itself. This way, people interested in finding out whom to talk to about a particular piece of the kernel would be able to find what they needed right in the part of the kernel in which they were interested. **Jan Engelhardt** is in favor of this, but added the suggestion that instead of comments, maintainer information should be embedded in the source code itself. That way, compiled into the kernel binary, it could be read via a simple command by anyone who was interested. But, Krzysztof points out that the only people who are going to be interested in this data are people hacking on the kernel, so there's no need to bloat the kernel binary with the information. Regardless, it does seem as though there is some support for reconsidering how maintainership information is handled. But, it may be that one of the MAINTAINERS file's greatest values is how visible it is. It's just fun to poke around in it! So undoubtedly, a wide range of issues will be hashed and rehashed by everyone before a decision is made.

It's probably safe to say that the **2.2 kernel** is in the utter deep dark icy freeze of death. The latest release candidate for 2.2.27 is from January 2005, and efforts to continue to patch it have been met with resistance. **Willy Tarreau** points out that any release, even a release candidate, might be interpreted by users as an indication that the 2.2 kernel was being maintained actively—an impression he did not feel corresponded to the truth. For one thing, he says that there are known security fixes that also have not been included in any of the patches. He feels the 2.2 kernel is just too out of date to bring back. But **Xose Vazquez Perez** feels that if the 2.2 kernel is going to be that deeply frozen, it shouldn't be listed on the front page of kernel.org as if it were a living kernel. If it's really dead, let it die, Xose argues. And, if it's not really dead, let it be updated. But, it may be that the 2.2 kernel still should be recognized just for historical value, even if it won't be developed anymore.

Apparently, **802.11** is complex—like, more than usual for the kernel. It turns out that wireless devices are regulated differently everywhere, so hardware manufacturers have started producing just a generic device, relying on the software to implement the regulations—ouch. **Luis R. Rodriguez** recently suggested constructing a massive database to keep track of the large and growing variety of legal restrictions the kernel will have to take account of in order to implement 802.11 properly. And, where would all this data be stored? In the kernel, naturally! Luis at first suggested an external Web site, something interactive that the whole community could participate in maintaining. But, as **Bruno Randolf** said, this would make the kernel sources, as far as 802.11 was concerned, dependent on this external source, while the kernel itself should be the true source, he said. So, the kernel probably will grow some very complex 802.11 legal information in the near future.

—ZACK BROWN

LJ Index, July 2008

1. Towns in Vermont that voted unanimously for Internet access: 10
2. Lowest percentage voting to approve Internet access: 80
3. Fiscal 2007 McKesson revenues in billions of dollars: 93
4. Fiscal 2007 McKesson software sales in billions of dollars: 1.9
5. Latest annual growth in McKesson software sales, in millions of dollars: 300
6. Number of health-care services applications offered by McKesson: 70
7. Number of McKesson applications already running on Linux: 50
8. Percentage of new McKesson applications planned to run on Linux: 100
9. Number of remaining applications McKesson plans to have running on Linux: 20
10. Top percentage of expected of Linux-based capital expense budget savings for health-care providers: 70
11. Percentage decrease in downtime for Novell SUSE Linux between Yankee Group's 2007 and 2007–2008 Global Server Operating Reliability Survey: 73
12. Percentage decrease in downtime for Red Hat Enterprise Linux between Yankee Group's 2007 and 2007–2008 Global Server Operating Reliability Survey: 73
13. Percentage decrease in downtime for Debian Linux between Yankee Group's 2007 and 2007–2008 Global Server Operating Reliability Survey: 25
14. Percentage increase in downtime for Windows Server 2003 between Yankee Group's 2007 and 2007–2008 Global Server Operating Reliability Survey: 25
15. Percentage of Yankee Group 2007–2008 survey respondents running at least one RHEL server: 26
16. Percentage of Yankee Group 2007–2008 survey respondents running at least one Novell SUSE server: 17
17. Percentage of Yankee Group 2007–2008 survey respondents running at least one Debian server: 24
18. Percentage of Yankee Group 2007–2008 survey respondents running at least one Ubuntu server: 22
19. Size in billions of dollars of the logistics and manufacturing applications market on Linux, by 2011: 1.2
20. Size in billions of dollars of the human capital management market on Linux, by 2011: 2

Sources: 1, 2: Tim Nulty | 3–10: *InformationWeek* 11–18: Yankee Group and the Institute for Advanced Professional Studies (IAPS) | 19, 20: IDC analyst Al Gillen, in *InformationWeek*

Kernel Candy

In April 2008 (at the time of this writing), the Linux Foundation published a progress report with the plain-wrapper title "Linux Kernel Development", authored by Greg Kroah-Hartman, Jonathan Corbet and Amanda McPherson. Relying heavily on Jonathan's gitdb and other tools, they probed the kernel.org Web site and the git kernel depository and organized the results into a neatly arranged assortment of tasty nuggets. Here are a few of them:

- For each day during the past 2.5 years, 3,621 lines were added, 1,550 lines removed and 1,425 lines changed. "That rate of change is larger than any other public software project of any size."
- Fifteen percent of kernel code contributions during the past three years have come from the top ten individual developers. Thirty percent has come from the top 30 developers.
- The top five individual developers, in order, were Al Viro, David S. Miller, Adrian Bunk, Raif Baechle and Andrew Morton. Each contributed more than a thousand changes.
- More than 70% of kernel development is being done by contributors being paid for their work.
- The size of the individual development community has doubled in the last three years.
- Of 31 listed corporate sources of kernel code changes (that is, companies employing individual developers contributing changes), the top two, with 13.9% and 12.9%, respectively, were None and Unknown. These were followed in order by Red Hat (11.2%), Novell (8.9%), IBM (8.3%) and Intel (4.1%). "The numbers presented are necessarily approximate", the report says.
- Companies outside the IT business contribute too. For example, "the 2.6.25 kernel will include an implementation of the PF_CAN network protocol, which was contributed by Volkswagen. PF_CAN allows for reliable communications between components in an interference-prone environment—such as that found in an automobile."

The concluding line is a model of hedged understatement: "With the current expansion of Linux in the server, desktop and embedded markets, it's reasonable to expect this number of contributing companies—and individual developers—will continue to increase."

Source: **https://www.linux-foundation.org/publications/linuxkerneldevelopment.php**.

—DOC SEARLS

Man vs. Myth

Greg K-H and the Kernel Driver Project

Don't tell Greg Kroah-Hartman that Linux hurts for device drivers. He's heard too much of that rap, and he's already done plenty to stop it. We should thank him and help pick up the ball. I'm doing both here.

The beginning of the end of the Missing Drivers Myth came at the 2006 Ottowa Linux Symposium, where Greg said, "Linux supports more different types of devices than any other operating system ever has in the history of computing."

Still, the OSDL (later the Linux Foundation) board—composed mostly of large vendors—listed device drivers as the #2 "most pressing issue". So the Linux Driver Project (LDP) was created. Alas, Greg reports on his blog, "No vendors showed up." But after he announced, "Tell me all of the hardware that you know of that is not supported by Linux!", he writes, "the response from users was overwhelming". Thus, a canonical wiki list was created at the LDP.

After this, Greg went to each vendor personally, and the conversation almost always went like this:

GREG: "What hardware do you ship that is not currently supported by Linux?"

VENDOR: "It all is."

GREG: "But wait, why are you claiming that 'Linux drivers' is your second most pressing issue today with Linux?"

VENDOR: "I don't know."

Thanks to those clues, missing drivers is out of the board members' top ten pressing issues.

But, there always is work to be done. As Greg puts it, that work falls into four categories of user complaints. Here they are, with excerpts of Greg's responses to each:

1. Printer and scanner support: "...already being handled very well by the Linux Printing Project and the SANE Project. Printer and scanner drivers in Linux are user-space programs and libraries and have nothing to do with the kernel at all. If you have any issues with these types of devices, please go ask the developers of those projects about them."

2. Older devices no longer manufactured that people really want to see working on their Linux machines someday: "...is hard. It would be great for Linux to support all of these older devices, but without the specs for the device, or in many cases, a company that is still in business, Linux support is going to be very difficult to achieve....Luckily, for almost all modern hardware devices, it is not necessary."

3. Wireless device support: "the Linux-Wireless group of developers has done an amazing amount of work in the past year, adding a whole new wireless protocol stack to the Linux kernel, as well as numerous different hardware drivers, some initially created by vendors and others created by reverse-engineering the hardware with no vendor help or approval. The latest kernel.org releases contain a raft of new hardware support for wireless drivers, and the number of active drivers in the queue to be added in the near future is quite large. Alas, there are still some wireless vendors that do not provide Linux support directly. Two of these, Atheros and Broadcom, have drivers created by the community through reverse-engineering efforts....Hopefully, this will change in the future."

4. Video input device support: "...there is an active Linux developer community in this area, but it seems to be hampered by a different development model...and a lack of full-time developers, not to mention a high degree of interpersonal conflicts that seem quite strange to outsiders. Support for a large majority of these devices is slowly trickling into the main kernel tree—the most important being the USB video class driver, which will support almost all new USB video devices in the future, thereby removing the major problem most users will face when purchasing a new video device."

In addition to further education, Greg has opened development by keeping all code related to the LDP in a quilt patch series that is automatically included in the linux-next-daily kernel releases, which are then contained in a git tree that "provides a place where developers can provide changes, updates and see where they can help out if they wish to do so in a much easier manner. It also provides a way for companies participating to observe the status of their code in a much more open manner."

It would be nice if Atheros and Broadcom were among those companies.

For more, visit **linuxdriverproject.org**.

Greg's blog post is at **www.kroah.com/log/linux/linux_driver_project_status-2008-04.html**.

—DOC SEARLS

They Said It

Where your talents and the needs of the world cross, there lies your vocation.

—Aristotle, www.quotationspage.com/quotes/Aristotle

All paid jobs absorb and degrade the mind.

—Aristotle, www.quotationspage.com/quotes/Aristotle

Thinking of and delivering IT as a service allows IT to become part of the business, and not merely the dumb bits behind it. Open source and SaaS make it all happen. Savvy IT shops will invest in both.

—Matt Asay, www.cnet.com/8301-13505_1-9915970-16.html?part=rss&tag=feed&subj=TheOpenRoad

Windows Is Collapsing: How What Comes Next Will Improve.

—Title of a Gartner presentation by Michael Silver and Neil MacDonald, www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9076698

If you know Linux, you're going to know we sell Dell products with Linux on them.

—Russ Ray, of Dell product marketing, www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9077678&intsrc=hm_list

Our hospitals aren't ready yet for Linux on the desktop, but it's coming....If you look at the total costs of hospitals and the pressure on hospitals to continue to lower their costs, it's coming.

—Michael Simpson of McKesson Provider Technologies, www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9052142

—DOC SEARLS

What Are They Using?

Nicco Mele: a Man and His Cave

Nicco Mele was born in West Africa, has lived all over the world, speaks many languages, and produces, among many other things, the Junglecast podcasts.

He also is probably the only human—and certainly the only geek—who has worked for a full house of presidential candidates: Howard Dean, Barack Obama, Hillary Clinton and John McCain. I first met Nicco in the server room of the Dean campaign in Burlington, Vermont, in the winter of 2004. The servers were running Linux. Geeks with laptops were flopped all over the place, hacking in Drupal and otherwise pioneering the political hackery now being leveraged in countless campaigns and political Web sites and services. Nicco ran that show.

Now he runs EchoDitto, an Internet strategy and consulting company with offices in New York, Washington, DC and Cambridge, Massachusetts, where we are neighbors and get to hang out. "At work, we rely heavily on customized Drupal and WordPress setups", he says. But what he'd rather talk about is his home tech life:

> I'm a believer in open source, well beyond just technology. For example, I strive for all of our home electronics to be open source wherever possible.

This begins with the Man Cave. When my wife and I, as newly-weds, bought our first home about 9 months ago, my wife—terrified of the proliferation of cables around our old apartment—offered me the basement for all of my technological needs. And lo! The Man Cave was born. With a 110" screen, a hi-def projector and surround sound—and complete darkness in the bright noon of day, the Man Cave is ideal for consuming *Battlestar Galactica*. Managing all this with open source? Enter my MythTV box, recently upgraded thanks to the wonders of—Woot!—Core 2 Quad Q6600 processor with four cores on one chip. Two 320GB 7200 RPM SATA drives for 640 gigs. Add some free software, and I've got a DVR and media server that is so crazy exciting, it's not funny. Now, if only I could get a voice-activated remote: "Computer, increase volume!"

For more, start with **nicco.org**, and follow links from there. There are many rich veins to mine.

—DOC SEARLS

Come to LinuxJournal.com and Stay a While

LinuxJournal.com is a pretty great on-line community. We now have a dedicated irc channel at #linuxjournal on irc.freenode.com (or at **www.linuxjournal.com/irc**) where the *Linux Journal* staff hangs out with readers and good conversations ensue. You might find me in there, so come say hi. You also are likely to find many of our regular authors and other staff members, so if you ever want to chat with us, we welcome you to drop in.

We also encourage you to drop by LinuxJournal.com and stay a while. You are very likely to find multiple articles that interest you. Be sure to check out the related links under each article to find other stories of interest. Search the site, or browse by topic, and you'll very likely find what you are looking for. With tens of thousands of articles at LinuxJournal.com, if it's about Linux and open source, we probably have it covered.

—KATHERINE DRUCKMAN

Open Source solutions for Enterprise Infrastructure Management

opengear

Next Generation Infrastructure Management

Complete Management LAN Gateway and Console Server in a single box

Opengear IM4200

- 8, 16, or 48 RS-232 Ports
- Dual Ethernet
- Dual Power AC or DC
- 512MB USB Flash
- Built-in Modem
- Embedded Nagios Checks
- Embedded IPMI Tools
- Cisco Rolled or Straight cable
- Service Processor Management iLO, RSA, DRAC, ALOM, IPMI

Tunnel HTTP/HHTPS/RDP/VNC/X/ TCP & UDP IPMI SOL over SSH using our free SDTConnector.

Live Online Demo

USA Sales Office
630 West 9560 South #A
Sandy, UT 84070
1-888-346-6853
sales@opengear.com

Australian Sales Office
44 Market Street, Level 26
Sydney NSW 2000
+61 2 9089 8940
sales@opengear.com.au

opengear.com

Olympic MIDs

From a Linux perspective, the most remarkable location at January's CES (Consumer Electronics Show) was Intel's Mobility booth. It was so crowded, I could barely elbow my way to the display cases. When I asked to see gear that ran Linux, I was told Linux was running on most of the gear they were showing there.

And sure enough, it was. I got to see and handle MIDs, or Mobile Internet Devices, from Lenovo, Clarion, Aigo, Samsung and Digifriends. Most appeared to be running China's Red Flag Linux and the CoolFox browser—obviously, a Mozilla derivative. The UIs were modeled on the iPhone, with square application icons and a "coverflow"-like file browser that let you fan horizontally or vertically through choices.

Since then, it has become clear that Intel and its OEMs are pushing to have some of these products ready for the Summer Olympics in Beijing. It should be interesting to see not only how much reporting in the wild is made possible by these new Linux handhelds, but also how many cool new hacks on them will be encouraged by the setting.

Of particular interest to me is the Lenovo Ideapad U8 Mobile Internet Device, for two reasons.

First, Lenovo is already the first major PC hardware OEM to market Linux aggressively. Its US index page (**lenovo.com/us/en**) at the time of this writing (in April 2008) says, "ThinkPad notebooks with Linux" right up front, rather than buried somewhere down the directory tree.

Second, the Ideapad U8 is an interesting device. It lacks the slide-out keyboard of the Nokia N810 (the Linux pioneer in the category), but it has a virtual QWERTY keyboard that can pop up along the bottom of the screen—plus a numeric hardware keypad on the right of the screen for T9 predictive text entry. In addition to the touchscreen (for finger or stylus), it has a hardware pointer reminiscent of the familiar red ThinkPad nub, but with no moving parts. And it has a bunch of other features you'd expect in a Linux handheld PC.

The Ideapad U8 was vetted at the Spring 2008 Intel Developer Forum in Shanghai, where Intel also shared details of the Menlow platform on which the device (along with many others) is based. Menlow puts the new Silverthorne CPU on the single-chip Poulsbo chipset. Significantly, at least one of the units being shown came with a colorful back badged with Olympics imagery.

An open question at this point is hackability. Nokia has openly courted and supported individual developers through the Maemo development platform. Nokia's purchase of Trolltech earlier this year also says it's serious about Linux development on mobile devices. Will Menlow-based MIDs veer in the "applianced" direction (like the iPhone) or the "generative" direction (like the Nokia MIDs)? For more about the difference, read "A Tale of Two Futures" in this month's EOF on page 96.

From what I gather by talking to Intel folks, these new MIDs are more likely to veer generative, on the model of the ASUS Eee PC, which ships with Xandros jiggered for "consumer" use, but which also remains completely hackable. We look forward to reports from *Linux Journal* readers as these new MIDs flow into the market starting this summer. May the best ones win.

—DOC SEARLS

_LJ_ pays $100 for tech tips we publish. Send your tip and contact information to techtips@linuxjournal.com.

EmperorLinux

...where Linux & laptops converge

Rugged Linux: The Tarantula

- Based on the Panasonic Toughbook CF-30
- Fully rugged, MIL-SPEC-810F tested: drops, dust, moisture, vibration, thermal stress, and more
- Shock mounted hard drive, magnesium alloy casing, fanless design

Features include:

- 1.6 GHz Core 2 Duo
- 13.3" XGA TouchScreen LCD, X@1024x768
- 80-200 GB hard drive (7200 rpm SATA), up to 4 GB RAM
- DVD±RW/DVD-RAM multi-drive
- Ethernet, 802.11a/b/g WiFi, Bluetooth, EVDO, GPS
- One year Linux tech support - phone and email
- Three year manufacturer's on-site warranty
- Choice of pre-installed Linux distribution:

Powerful Linux

Rhino D830/M6300

- Dell Latitude D830/Precision M6300
- Up to 17" WUXGA w/ X@1920x1200
- NVidia Quadro FX 3600M graphics
- 1.8-2.8 GHz Core 2 Duo/Extreme
- Up to 8 GB RAM
- 60-200 GB hard drive
- DVD±RW or Blu-ray
- Starts at $1350

Tablet Linux

Raven X61 Tablet

- ThinkPad X61 tablet by Lenovo
- 12.1" SXGA+ w/ X@1400x1050
- 1.6 GHz Core 2 Duo
- Up to 4 GB RAM
- 80-200 GB hard drive
- Pen/stylus input to screen
- Dynamic screen rotation
- Starts at $2150

Portable Linux

Toucan T61/T61p

- ThinkPad T61/T61p by Lenovo
- Up to 15.4" WUXGA w/ X@1920x1200
- NVidia Quadro FX 570M graphics
- 1.8-2.6 GHz Core 2 Duo
- Up to 4 GB RAM
- 80-250 GB hard drive
- 5.2-6.0 pounds
- Starts at $1530

www.EmperorLinux.com 1-888-651-6686

Model prices, specifications, and availability may vary. All trademarks are the property of their respective owners.

Integrating OpenID

Integrate OpenID into any Rails application, using off-the-shelf libraries and a bit of custom code.

REUVEN M. LERNER

The past few months, we've looked at two different ways to authenticate users coming to a Web site. First, we looked at OpenID, an increasingly popular distributed authentication system. With OpenID, users control their information, as well as which applications are allowed to use that information.

Last month, we looked at acts_as_authenticated, a plugin for the Ruby on Rails framework that is quite traditional, asking visitors to enter a user name and password in order to access restricted services.

This month, we take an initial look at how we might be able to incorporate OpenID—and by extension, a combination of OpenID and traditional authentication—into our own Rails applications. In OpenID lingo, we want our application to be a "consumer", asking an OpenID "provider" of the user's choosing for authentication information, rather than gathering and checking that information ourselves.

In order to incorporate OpenID into a Web application, we don't need to replace the whole cookie/session/login portion of the framework.

OpenID is a pretty well established standard, and integration into a Rails application isn't all that difficult. However, the number of OpenID-supporting libraries and plugins has gotten a bit out of control, such that it's sometimes hard to know (or believe) which ones actually work, not to mention which ones are easiest to work with.

Authentication and OpenID

Authenticating users for a Web site is normally a straightforward task. You ask users, via an HTML form, to enter their user names and passwords, and then compare that combination against the database. (For security purposes, of course, it's usually best to encrypt the password in the database, and then compare the encrypted input with what is in the database.) If the user name/password combination exists in the database, the user can log in.

Of course, HTTP is a stateless protocol, which means there isn't really any such thing as being "logged in". Rather, we rely on cookies, pieces of data provided by the server but stored in the user's browser, which are passed to the server with each subsequent HTTP request. In this system, logging in takes place when the server sets a cookie on the user's browser. In Rails and many other Web frameworks, cookies also are used to keep track of a user's "session", attributes associated with this user on this browser.

In order to incorporate OpenID into a Web application, we don't need to replace the whole cookie/session/login portion of the framework. Rather, we need to change the way we authenticate users, setting the login cookie after an OpenID provider has indicated that a user has been identified legitimately.

A traditional Rails-based login system would involve an HTML form, a controller action that compares the submitted form values against a database, and then a login page. To replace this with OpenID, we need to modify our controller such that it asks an OpenID server to authenticate the user.

But, wait a second. The whole point of OpenID is that users enter a URL (that is, their unique OpenID), and that they authenticate against a server associated with that URL. This means the HTML form needs to change, such that it asks for a URL instead of a user name and password.

Moreover, we have to take into account the fact that our server needs to redirect users to an OpenID server, which then will redirect back to our system, indicating whether the user has logged in successfully.

There are, as I indicated above, many Ruby- and Rails-related resources having to do with OpenID. Unfortunately, many of them are poorly documented, out of date or relatively hard to use. For example, there is a Ruby gem called openid_login and a plugin called open_id_authentication that might well work with a bit of hacking. But, their documentation is out of date, and I encountered problems with, among other things, the double suffixes (.html.erb) that Rails now uses with templates. So, although I'm sure it's possible to get this gem to work with OpenID and modern Rails installations, it probably will take time and effort—more than I would expect from a prepackaged solution.

Thus, my suggested solution to the whole question of OpenID is to use the simple, low-

level ruby-openid gem, which happens to have support for Rails applications built in. This gem is actually very well documented in its current form—version 2.0.4 at the time of this writing. But, be careful; much of the documentation you'll find on-line is out of date and implements OpenID-related functionality using the 1.x version of this gem with an older, incompatible API.

To install the gem, of course, we write:

```
gem install ruby-openid
```

We then create a controller for handling our OpenID-related actions:

```
script/generate controller openid new create complete openid_consumer
```

These four actions, the fourth of which is private, are what we'll need in order for people to log in with OpenID.

Now we can create an HTML form in a view; I created this simple view as login.html.erb within views/openid/new.html.erb:

```
<html>
<head>
    <title>Log in with OpenID</title>
</head>
<body>
    <% if not flash[:error].blank? %>
        <p><b><%= flash[:error] -%></b></p>
    <% end %>

    <% form_tag "/openid/create" do %>
      <%= text_field_tag "openid_url" %>
      <%= submit_tag "Log in with OpenID" %>
    <% end %>
</body>
</html>
```

Because everything between <% and %> in an ERb template is evaluated as Ruby code, we'll need to understand what is going on here. First, we create a form that is not connected to any object using the form_tag helper. (If the form were connected to an object, we would simply use the form helper.) We give it a URL of /openid, which we will discuss

1 2 3 4 5 6

Strategic Independence

PGI Unified Binary™ technology brings binary compatibility to Intel and AMD x64 processors for all of your HPC Fortran, C and C++ applications. Follow your own roadmap! PGI CDK compilers and tools are available directly from most cluster suppliers. Take a free test drive today at www.pgroup.com/reasons

PGI CDK® Cluster Development Kit®

The Portland Group, Inc. is an STMicroelectronics company. CDK is a trademark or registered trademark of STMicroelectronics. PGI, Cluster Development Kit, and PGI Unified Binary are trademarks or registered trademarks of The Portland Group, Incorporated. Other brands and names are the property of their respective owners.

in a little bit, when we look at routing.

The form contains a single text field, whose name and id attributes both will be set to openid_url. Modern browsers recognize this name and use it to fill in an OpenID URL automatically. A submit button and a closing end tag complete the form.

Thus, my suggested solution to the whole question of OpenID is to use the simple, low-level ruby-openid gem, which happens to have support for Rails applications built in.

Storing User Information

When we display this form in our browser, the user has one option—namely, to sign in with OpenID by entering a URL. The action (create) that is invoked has to find the user's OpenID server and redirect to that server. In order to do this, we need an instance of OpenID::Consumer, an object defined by the ruby-openid gem. Because we will continue to need this, we can create it as an instance variable:

```
def openid_consumer
 if @openid_consumer.blank?
   @openid_consumer =
     OpenID::Consumer.new(session,

OpenID::Store::Filesystem.new("#{RAILS_ROOT}/tmp/openid"))
 end

 return @openid_consumer
end
```

Notice that we're storing the OpenID information on the filesystem, in the tmp directory under the root of our Rails project directory. This is a bad idea when you have multiple Web servers, but is certainly good enough for a small or beginning site.

Now that we have a method named

Listing 1. openid_controller.rb

```
require 'openid'
require 'openid/store/filesystem'

class OpenidController < ApplicationController

 def openid_consumer
  if @openid_consumer.blank?
    @openid_consumer =
      OpenID::Consumer.new(session,
          OpenID::Store::Filesystem.new("#{RAILS_ROOT}/tmp/openid"))
  end

  return @openid_consumer
 end

 def new
  # Nothing to do here -- it's all in the form
 end

 def create
  # Get the OpenID parameter
  openid_url = params[:openid_url]

  # Make sure we got something
  if openid_url.blank?
    flash[:error] = "No OpenID was entered; try again"
    redirect_to :back
    return
  end

# Get an OpenID response
  openid_response = openid_consumer.begin openid_url

  home_url = url_for :controller => "openid", :action => "index"
  complete_url = url_for :controller => "openid", :action => "complete"
  openid_redirect_url = openid_response.redirect_url(home_url, complete_url)
  redirect_to openid_redirect_url

  return
 end

 def complete
  home_url = url_for :controller => "openid", :action => "index"
  complete_url = url_for :controller => "openid", :action => "complete"

  openid_response = openid_consumer.complete(params, complete_url)

  session[:openid] = openid_response.identity_url
  flash[:error] = "You have been logged in as '#{session[:openid]}'"
  redirect_to :action => "new"
  return
 end

 def clear_session
  reset_session
  flash[:error] = "Session cleared."
  redirect_to :action => "new"
 end

end
```

openid_consumer and an instance variable named @openid_consumer, we can implement the create action, to which our HTML form is going to be submitted:

```
def create
 # Get the OpenID parameter
 openid_url = params[:openid_url]

 # Make sure we got something
 if openid_url.blank?
   flash[:error] = "No OpenID was entered; try again"
   redirect_to :back
   return
 end

 # Get an OpenID response
 openid_response = openid_consumer.begin openid_url

 home_url = url_for :controller => "openid", :action => "index"
 complete_url = url_for :controller => "openid", :action => "complete"
 openid_redirect_url = openid_response.redirect_url(home_url, complete_url)
 redirect_to openid_redirect_url

 return
end
```

In other words, we get the user's OpenID URL, and we check that it wasn't blank. Then, we use our instance of OpenID::Consumer to begin the OpenID login process, using open_consumer.begin, passing it the user's OpenID URL. If all goes well, this returns an instance of SuccessRequest, which also hands us the URL to which we should redirect the user. (If the request fails, the response will be a subclass of OpenIDStatus.)

Completing the Login Process

When we send the user to the user's OpenID server, we have to provide two different URLs as arguments: one that we're calling home_url, and the other that we're calling complete_url. The former is the root URL of our site; typically, it'll be a top-level URL. The latter, complete_url, tells the OpenID server to which URL the user should be redirected after logging in. In both cases, I use the built-in Rails url_for method, which constructs a URL out of a controller and action name.

When the user returns from the OpenID server, it will be to the URL indicated in complete_url. This means we have to define our complete method as well:

```
def complete
 home_url = url_for :controller => "openid", :action => "index"
 complete_url = url_for :controller => "openid", :action => "complete"
 openid_response = openid_consumer.complete(params, complete_url)

 session[:openid] = openid_response.identity_url
 flash[:error] = "You have been logged in as '#{session[:openid]}'"
 redirect_to :action => "new"
 return
end
```

After defining home_url and complete_url once again, we invoke the complete method on our instance of OpenID::Consumer. If the response is good (and here we assume that it is, ignoring the possibility that we might have gotten an instance of OpenIDStatus back). Obviously, your real-life applications should include such a check.

Sure enough, when we put all this in place, it works! We can enter our user ID into the HTML form. We get verified by the user's OpenID server, even if that means another redirect. And, we get the user verified with basic information.

Conclusion

OpenID is a simple but powerful idea that is slowly but surely transforming the way we manage identities on the Internet. A growing number of applications use OpenID, and it is becoming increasingly popular among users as well.

Adding OpenID to an application does not need to be complicated or difficult. As I show this month, incorporating OpenID into a Rails application requires understanding one particular Ruby object, namely OpenID::Consumer, and the odd, redirect-based, three-part OpenID login system specification.■

Reuven M. Lerner, a longtime Web/database developer and consultant, is a PhD candidate in learning sciences at Northwestern University, studying on-line learning communities. He recently returned (with his wife and three children) to their home in Modi'in, Israel, after four years in the Chicago area.

Resources

OpenID: the main page for OpenID is **openid.net**. For documentation about the Ruby gem for OpenID, see **openidenabled.com/files/ruby-openid/docs/2.0.4/classes/OpenID/Consumer.html**.

OpenID on Rails: the main Wiki page for this is **wiki.rubyonrails.org/rails/pages/OpenID**.

There are a number of blog postings and tutorials about OpenID and Rails, some of which are more out of date than others. Perhaps the best one is **railscasts.com/episodes/68**, which is a nice visual introduction (along with source code) about what is happening.

MARCEL GAGNÉ

You Look Marvelous on the Web!

Looking good is easy for our regular guests. However, looking good on the Web takes a little more work, which doesn't mean it can't be a lot of fun. With a little help from your Linux system, your smile will shine on-line!

Yes, François, I think it would be great to add a gallery of our regular guests on the restaurant's Web site, but I do have a couple concerns. First and foremost, I really don't think you should call it a "Rogue's Gallery". Second, why on earth are you coding HTML by hand? That's going to take forever, and our guests will be here momentarily. Lucky for you, tonight's menu has some great free software for your Linux system that will make creating the gallery a breeze—later, though. I can see our guests arriving as we speak.

Good evening and welcome one and all to *Chez Marcel*! Your tables are ready, as are we to serve you. My faithful waiter, François, will fetch your wine while I introduce tonight's featured Linux software. François, to the wine cellar. *Vite!* In the South wing, you'll find a case of 2003 Sariza from Bulgaria. The Sariza is a great medium-bodied red wine that I'm sure you'll enjoy.

This is a wonderful tool that allows you to create a presentation of images in a hurry, whether for your Web site, for your family or for your company.

I must tell you that François had an excellent idea that involved creating a Web photo gallery. Before I show you how easy it can be to create such a gallery, I should tell you about a package you need to have on your system—a package that will let you do all sorts of magical things with images. No, I'm not talking about The GIMP. The package is ImageMagick, and if you don't have it installed already, you should do so now, as we'll need its tools later. What sorts of tools? Well, for instance, you can find some interesting information about a media file by using the identify command:

```
$ identify myphoto.jpg
myphoto.jpg JPEG 800x1161 DirectClass 271kb
```

As you can see, the photo is 800 pixels wide by 1,161 pixels in height, and it is a JPG image. What if I wanted to create a small 150-pixel-wide thumbnail from this image? ImageMagick has a tool for that as well. It's called mogrify:

```
mogrify -geometry 150 myphoto.jpg
```

And, just like that, we have a 150-pixel-wide photo. Of course, you might want to have a backup of the original.

Although not a Web album, you can use the montage command to create a montage, much like a photographic contact sheet:

```
montage -geometry +5+5 -size 150 -frame 20 lugnuts/*.jpg lugnuts.png
```

The resulting image (Figure 1), complete with a nice beveled frame, can be printed and stored in an physical album. Pretty cool, *non*?

Figure 1. The ImageMagick montage command makes it possible to create quick-and-easy contact sheets.

To get those images to the Web, we have other tools. Some require that you have server-side access to the server, and these can be quite complex and feature-rich. I'll show you one of those shortly. In the meantime, let's assume you don't have shell access to your Web site or you aren't allowed to install programs or run scripts. You might well be in a bind if you want those pictures

on your own Web site. Fear not. This is where iGal and our friends, the ImageMagick group of tools, come into play. iGal is a simple Perl script, originally written by Eric Pop (at Stanford). These days, iGal is maintained and updated by Wolfgang Trexler. This is a wonderful tool that allows you to create a presentation of images in a hurry, whether for your Web site, for your family or for your company. You can get the latest iGal from Wolfgang's site at **trexler.at/igal**.

Installing this is child's play, because iGal (which stands for Image GALlery generator), being a Perl script, is already source. Visit the site and download the latest tarred and gzipped bundle. Then, extract the package and install it:

```
tar -xzvf igal-1.4.7-wt.tar.gz
cd igal-1.4.7-wt
sudo make install
```

To make your instant slideshow, change directory to where you already have a collection of images, and type the following:

```
igal -xy 150 --bigy 800
```

That's all there is to it. You don't need either the -xy 150 or the --bigy 800 options I added. The first creates an HTML slideshow with a title page made up of thumbnails scaled to a maximum of 150 pixels along their longest dimension. The bigy option is useful if you have very large images. It takes your large photos and creates images of the selected y dimension (in this case, 800 pixels). To see the full-size image, your visitors just have to click the 800-pixel image. This whole process may take a minute or two, depending on the size of your images and the quantity. In the directory, you'll also find an index.html page, your original images, thumbnail versions of these (prefixed with .thumb_) and cross-linked HTML pages for each image. It should look something like what you see in Figure 2.

The only real editing that I wind up doing is changing the title of the index.html file. By default, the title for that page is "Index of pictures", and I tend to like something a bit more descriptive. When you run iGal, the default caption for each image is the image name itself. You can change that without editing all the images by running iGal with the -c option. This generates a file called .captions. Lines in the file look something like this:

```
img_0261.jpg ----
img_1400.jpg ----
```

To create captions for your images, append the text you want to the image:

```
img_0261.jpg ---- A picture of me with Tux
```

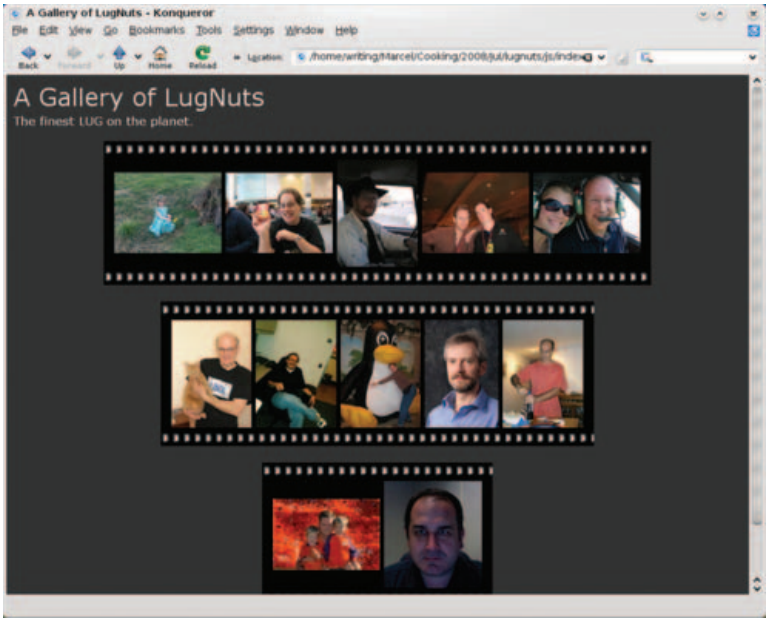


Figure 2. A Web Gallery, Courtesy of iGal

Save the file, rerun iGal with the -c option again, and all your images will have your selected captions. If you choose the -C option (uppercase C), you will get your captions, but the image names are preserved. Note that you should remove the .captions file first. Remember, all of this happens on your local PC. When you are done, all you need to do is transfer the directory, complete with HTML files, images and thumbnails, to your hosting provider. No server-side code is needed.

Perl is cool, of course. But a good old-fashioned bash script is equally cool, and that's the heart of Eduardo Sztokbant's Shalbum. You can get Shalbum at **shalbum.blogspot.com**. Like iGal, Shalbum uses ImageMagick to perform its magic. Similarities aside, Shalbum has some interesting additional features that set it apart, including forward and back thumbnails with the main image view, a built-in slideshow function and more. To install Shalbum, extract the tarred and gzipped bundle, then run a `make install`:

```
tar -xzvf shalbum-1.12.tar.gz
cd shalbum-1.12
sudo make install
```

To create your Web album, copy the images you want into a folder. With that part complete, it can be as simple as running the shalbum command:

```
shalbum -t "Proud members of the WFTL-LUG" -T 200 -C 4
```

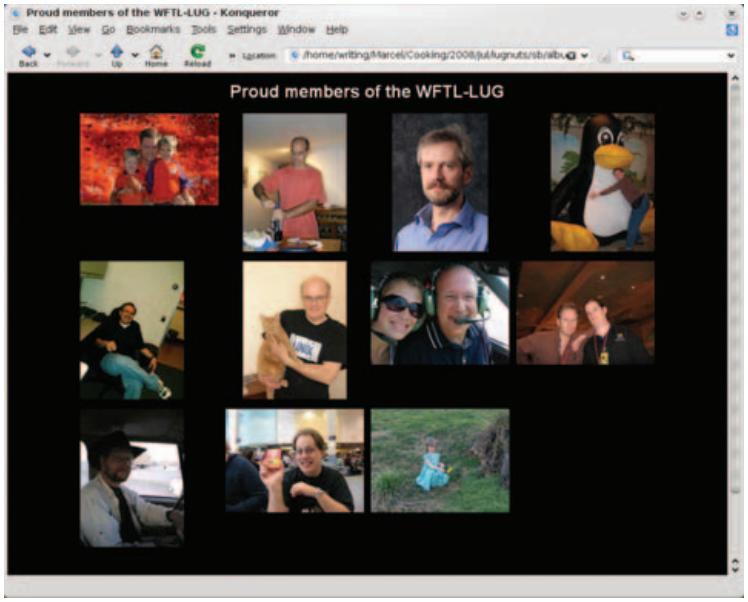


Figure 3. Quick, easy and stylish—Shalbum uses bash to generate a Web album.

The result of this command is shown in Figure 3. What I've done with the above command is pass a title for the album (the -t option), selected a 200-pixel thumbnail (the -T option) and specified that the main page should display four columns of thumbnails. The resulting album is generated in a subfolder called—wait for it—album. You can transfer that entire folder to your Web site, and you are good to go. The presentation for Shalbum is nice.

Each photo displays not only forward and back links to navigate through the gallery, but also shows the previous and next thumbnail at the bottom of each page. At the top of the gallery, you'll see a link labeled Play. That starts the slideshow. You can specify the number of seconds between each photo by using the -S option.

Perl is cool, of course. But a good old-fashioned bash script is equally cool, and that's the heart of Eduardo Sztokbant's Shalbum.

The final item on tonight's menu is far more complex and does require server-side access, as well as an Apache server with PHP and MySQL. The result is fantastic and yet amazingly easy to work with. It's called ZenPhoto, and it is probably the nicest and easiest Web photo gallery program I've seen so far. Despite its ease of use, the feature list is nothing short of impressive. EXIF and IPTC support is built in, as is support for video (Flash, QuickTime and 3GP). Images are uploaded via the Web interface, but you also can use FTP or SCP if you prefer. The interface makes it easy to edit as you go, make comments, tag photos, rate individual albums or photos, and generate albums from searches. You can watermark images, password-protect the gallery or individual albums, set up RSS feeds, allow users to comment and more. There can be multiple galleries and even sub-galleries. Have I mentioned that ZenPhoto is also themeable?

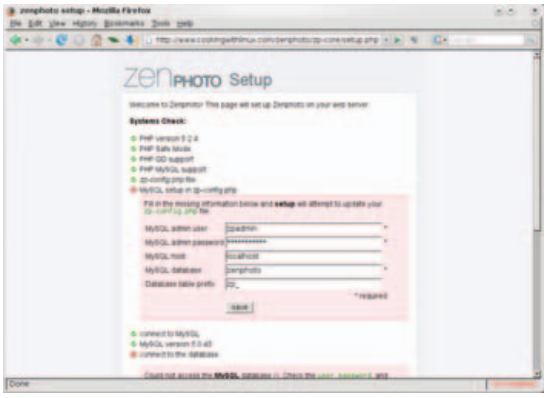


Figure 4. Installation is guided with pointers to help you set up your database.

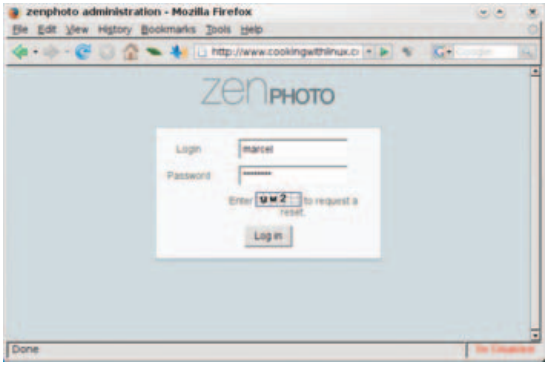


Figure 5. Two steps later, you are ready to log in and start uploading photos.

To get started, visit the ZenPhoto Web site at **www.zenphoto.org**, and download the latest source. Extract the bundle into your Web server's hierarchy. Of course, only you will know where exactly, but you pretty much can put it anywhere you like under those constraints. ZenPhoto does not need to be at Web root. That said, you do need to have administrative access to the system, or at least MySQL. Extract the package into your chosen location with this command:

```
tar -xzvf zenphoto-1.1.5.tar.gz
```

The resulting directory will be called zenphoto, which is fine, but you may choose to rename the folder to something that makes more sense to you, like myphotos or myalbums.

Installation is very simple. Just point your browser to the ZenPhoto installation address—for instance http://mywebsite.dom/zenphoto. If you are accessing ZenPhoto for the first time, it immediately will take you to the setup screen (Figure 4).

The setup screen checks to make sure you have the right software installed, including PHP support and modules. In my example, I haven't yet created my database. Use whatever tools you are comfortable with (for example, Webmin, PHPMyAdmin or the command line) to create a database and a user that has permissions to update that database:

```
mysqladmin -u root -p create zenphoto
mysql> grant all on zenphoto.* to
 ➥'zpadmin'@'localhost'
 ➥identified by 's3cr3tp4sswd';
mysql> flush privileges;
```

Click the Save button to verify your database setup, and you now should see a screen with a collection of friendly green check marks and a large blue Go! link at the bottom. If there are any issues, they will be highlighted for you (the wrong password, perhaps). Assuming all has gone well, click Go!. ZenPhoto creates the necessary tables, then provides you with a screen to set your admin user name and password. The only thing left to do is log in.

When you log in to the admin interface, you are presented with several tabs that let you define security, edit existing galleries and comments, adjust the layout and theme, and a whole lot more. What you most likely will want to do at this point is create an album and upload some photos. To create your first album, click the Upload tab. If you haven't done so already, create a new album by providing a name. Then, one by one, you can browse for your photos and add them to the list (Figure 6). The default allows for five individual photos, and you can upload them five at a time, but if you prefer, you can click the

he.net

Full 100 Mbps Port

Full Duplex

$2,000/month

Colocation
Full Cabinet
42U
$400/month

IP Transit
Gigabit Ethernet

BGP+IPv6+IPv4
Special
$5/Mbps

Order Today!

email sales@he.net or call 510.580.4190

he.net/ip_transit

Figure 6. Creating albums and uploading photos take only a few clicks.

Figure 7. ZenPhoto Gallery Built on the Default Theme

Add more upload boxes link at the bottom. When you are done, click the Upload button.

Your final gallery is ready soon after the photos are uploaded. Thumbnails are generated automatically. By default, the gallery is public, and the only password so far is to protect the administrative interface, in which you are currently working. To leave the administration screens and view your gallery, click the View Gallery link at the top right, and you'll be immediately transported to the public face of the gallery (Figure 7). Assuming you are logged in as the administrator, you will see an Admin Toolbox link in the upper right that lets you jump back to admin mode.

Now, I vaguely recall mentioning that ZenPhoto is themeable. In fact, the package comes with several themes, like the Stopdesign theme shown in Figure 8. Most themes have additional options that can be tweaked in the admin interface. These let you define the number of thumbnails displayed, the order in which they appear, photo download options, comment capabilities and more. You also can browse and download additional themes from the ZenPhoto Web site.

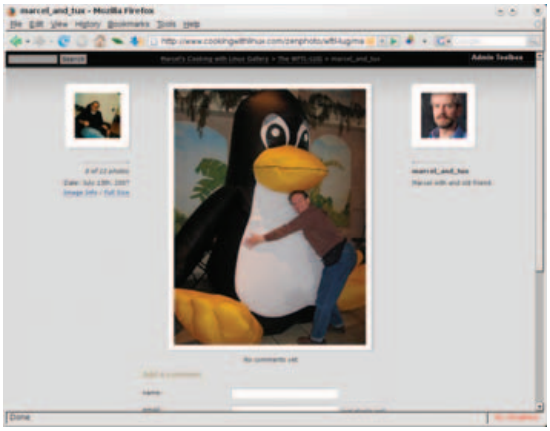


Figure 8. Different themes provide your albums with a style to reflect your own taste.

Once again, *mes amis*, we are out of time, though I must admit, François' gallery looks amazingly good with your smiling faces peering out from our Web site. If you haven't had the opportunity to do so yet, I hope you'll send us your photos shortly. In the meantime, the hour is truly getting late, and we must soon be on our way. François will, of course, refill your glasses a final time before we bid each other farewell. Raise your glasses, *mes amis*, and let us all drink to one another's health. *A votre santé! Bon appétit!*■

Marcel Gagné is an award-winning writer living in Waterloo, Ontario. He is the author of the *Moving to Linux* series of books from Addison-Wesley. Marcel is also a pilot, a past Top-40 disc jockey, writes science fiction and fantasy, and folds a mean Origami T-Rex. He can be reached via e-mail at marcel@marcelgagne.com. You can discover lots of other things (including great Wine links) from his Web site at www.marcelgagne.com.

Resources

iGal: **trexler.at/igal**

ImageMagick: **www.imagemagick.org**

Shalbum: **shalbum.blogspot.com**

ZenPhoto: **www.zenphoto.org**

Marcel's Web Site: **www.marcelgagne.com**

The WFTL-LUG, Marcel's Online Linux User Group: **www.wftl-lug.org**

The Straight Talk People℠
SINCE 1991

ABERDEEN

HOW MUCH STORAGE DO YOU NEED?

- Aberdeen Stirling Storage Servers deliver vast and flexible expansion capabilities in a single storage server without additional controllers
- Expand up to 328TB with Aberdeen DAS and JBOD units without any performance degradation
- SAS *and* SATA support concurrently in same server (separate arrays)
- Four Gigabit Ethernet Ports allow network teaming
- Dual Quad-Core Intel® Xeon® Processors with up to 1600 MHz Front Side Bus
- iSCSI support with available NAS versions
- Multiple available expansion slots

The Aberdeen line of expandable storage servers provide a full spectrum of scalable, maximum capacity storage solutions for data backup, media sharing storage, content creation, streaming media, nearline storage, and post-production needs. These servers feature high performance Quad-Core Intel Xeon processors and enterprise-level SATA drives providing over 800MB/s internal transfer rates, while being robust enough to provide up to 328TB of scalable storage without added controllers.

STORAGE SERVER

Expandable Storage Solution

- Dual Quad-Core Intel® Xeon® processors with up to 1600MHz FSB
- Up to 64GB 800 ECC FBDIMM Memory
- Supports both SAS & SATA Storage Drives
- Internal OS Hard Drives Included
- RAID 0, 1, 5, 6, 10 Capable
- Redundant Power Supply
- SAS & iSCSI Expansion Ports
- Windows & Linux NAS Available
- **5-Year Warranty**

3U 12TB Starting at	$7,995
4U 16TB Starting at	$9,995
5U 24TB Starting at	$12,995
6U 32TB Starting at	$17,995
8U 40TB Starting at	$20,995

+

JBOD STORAGE

Add up to 128TB

- Daisy-Chain JBOD Boxes
- Storage Server RAID Array
- Redundant Power Supply
- SATA & SAS Drive Support
- **5-Year Warranty**

16TB JBOD $7,995

+

XDAS STORAGE

Add up to 160TB

- Daisy-Chain DAS Unit + JBOD Expansion Boxes
- 2U, 3U, 4U Units Available
- RAID 0, 1, 5, 6, 10 Capable
- Redundant Power Supply
- SATA & SAS Drive Support
- **5-Year Warranty**

16TB DAS $12,995
16TB JBOD $9,495

= EXPAND UP TO 328TB

intel Xeon® inside™
Powerful. Efficient.

Intel, Intel Logo, Intel Inside, Intel Inside Logo, Pentium, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. For terms and conditions, please see www.aberdeeninc.com/abpoly/abterms.htm. lj026

888-297-7409
www.aberdeeninc.com/lj026

Of Movies, Trivia Games and Twitter

DAVE TAYLOR

How to write a movie trivia game for Twitter.

During the past few months, I have become an addict. In fact, I went from being a skeptic to being an evangelist, in a way that probably makes me a poster case for a 12-step program. What is this evil thing that's sucked up my brain and passion? It's not illegal; it's not something I have to hide from my children; but, yes, it's pretty geeky, and it's one of the fastest-growing services in the Web 2.0 universe: Twitter.

What I find most compelling about Twitter is that it's both popular and nascent, and as a result, you can see its best practices evolve before your eyes. Even in the few months I've been active with the service, it has gone from just personal updates (as in "Eating burger at McD's. Back to meetings in :30") to more business uses and news dissemination ("Flash: Redbox hacked by card sniffers. See…").

In a nutshell, Twitter lets you send very short messages to dozens, hundreds or even thousands of followers, and from a Linux/shell scripting perspective, it's very cool because the API lets you send messages easily with a single line of code. But, let's get there in a bit. First, we need something to transmit.

Movie Trivia? Sure!

Because I can't seem to shake my enthusiasm for writing games as shell scripts (speaking of psychological curiosities, that's another one for you), I thought it would be interesting to write a movie trivia game for Twitter. So, that's what we'll do.

What I find most compelling about Twitter is that it's both popular and nascent, and as a result, you can see its best practices evolve before your eyes.

The challenge is to figure out where the data will come from. I mean, I built up a huge database of word history trivia for **etymologic.com**, and my buddy Kevin Savetz and I wrote more than 500 computer trivia questions for **trivial.net**, and it's a huge amount of effort. Since creating those sites, I've become too lazy to repeat the effort, so the question is to identify a spot where I can leverage or repurpose existing movie information that will lend itself to a trivia game.

For this effort, I'll use the Internet Movie Database (**www.imdb.com**), which has an extraordinary amount of interesting movie trivia deep in its database. One place to start is its random movie quote feature, at **www.imdb.com/Games/randomquote.html**, but truth be told, that trivia is so darn obscure, I've never been able to identify any of the quotes, and I'm quite a movie fanatic.

Let's make this more complicated instead, and start with the IMDb top 250 movies list and isolate the quotes and trivia from those movies. That list is at **www.imdb.com/chart/top**, and if you crack it open, you'll see that each movie is referenced with a URL of this form http://www.imdb.com/title/tt0068646/.

This means a simple grep can pull out the URL of each and every one of the top 250 movies. Utilizing curl, here's everything you need:

```
curl -s http://www.imdb.com/chart/top | \
sed 's/</\
/g' | grep '/title/tt' | more
```

The output isn't quite what we want, but it's getting pretty close to a usable database with just this simple command, not even enough to justify a shell script:

```
a href="/title/tt0068646/">The Godfather
a href="/title/tt0111161/">The Shawshank Redemption
a href="/title/tt0071562/">The Godfather: Part II
a href="/title/tt0060196/">Buono, il brutto, il cattivo, Il
a href="/title/tt0110912/">Pulp Fiction
```

To strip out only what we need, because we really just want to have a file of 250 URLs of the top 250 movies, we merely need a tiny addition:

```
curl -s http://www.imdb.com/chart/top  | sed 's/</\
/g' | grep '/title/tt' | cut -d\" -f2
```

And, here's the result:

```
/title/tt0068646/
/title/tt0111161/
/title/tt0071562/
/title/tt0060196/
/title/tt0110912/
...many, many lines skipped...
/title/tt0325980/
/title/tt0061809/
/title/tt0113247/
```

It's easy to drop this all into a data file, fixing the URLs as we go along so that they are fully qualified, with a simple additional call to sed like this:

```
| sed 's/^/http:\/\/www.imdb.com/'
```

Now we have a data file full of URLs, like this:

```
http://www.imdb.com/title/tt0068646/
```

Visit this URL, and you'll find that it's the #1 top movie on IMDd, the brilliant film *The Godfather*.

Scraping Data for Fun

Okay, so we've figured out how to get a list of the top 250 movies according to IMDb voters, but the question is, "how can we get useful information at this point?" The answer is by going to each and every page and scraping the content thereon.

Look at the page for *The Godfather*, and immediately a simple trivia question game comes to mind: in what year was a particular popular movie released?

This can be done by simply grabbing the title of the page, which just so happens to be the film name and year of release:

```
curl -s http://www.imdb.com/title/tt0068646/ | grep '<title>'
```

It's not quite what we want, but pretty darn close:

```
<title>The Godfather (1972)</title>
```

It's close enough that we now can write a short script that takes an IMDb movie title URL and outputs the movie name followed by a pipe symbol (a convenient field separator) and the year the film was released:

```
#!/bin/sh

# given an IMDb film URL, output title & release year

curl -s "$1" | \
  grep '<title>' | cut -d\> -f2 | cut -d\< -f1 | \
  sed 's/([0-9][0-9][0-9][0-9])/| &/' | sed 's/(//;s/)//'

exit 0
```

(The complicated sed regular expression is to ensure that we don't merely match the open parenthesis, just in case the movie title includes parentheses.)

With that written, now we simply can pour the list into the script and pull a quick list of the top ten films:

```
for name in $(cat top250.txt)
do
./get-film-info.sh $name
done | head -10
```

And, here's the output:

```
The Godfather | 1972
The Shawshank Redemption | 1994
The Godfather: Part II | 1974
Buono, il brutto, il cattivo, Il | 1966
Pulp Fiction | 1994
Schindler's List | 1993
One Flew Over the Cuckoo's Nest | 1975
Star Wars: Episode V - The Empire Strikes Back | 1980
Casablanca | 1942
Shichinin no samurai | 1954
```

Okay, so we've figured out how to get a list of the top 250 movies according to IMDb voters, but the question is, "how can we get useful information at this point?"

Cool. Now we're getting somewhere. Let's stop here, and next month, I'll look at pulling out a random entry from the 250 entries, then generate three random numbers numerically close to the correct year and present all four as possible answers to the question, "when was *XX* released?"

For now, I think I'll pop *Casablanca* in to my Blu-ray player and relax while the team at *Linux Journal* struggles with laying out the column. See ya later, shweetheart.■

Dave Taylor is a 26-year veteran of UNIX, creator of The Elm Mail System, and most recently author of both the best-selling *Wicked Cool Shell Scripts* and *Teach Yourself Unix in 24 Hours*, among his 16 technical books. His main Web site is at www.intuitive.com, and he also offers up tech support at AskDaveTaylor.com. Follow him on Twitter if you'd like: twitter.com/DaveTaylor.

MICK BAUER

Customizing Linux Live CDs, Part III

Further notes on custom live CD security.

The past couple months, I've been showing how to create your very own customized Ubuntu live CD. In "Customizing Linux Live CDs, Part I" (*LJ*, May 2008), I provided a basic procedure for mounting an Ubuntu Desktop 7.10 ISO image; removing, adding and updating its software packages; and repacking it into a new ISO image.

In Part II (*LJ*, June 2008), I showed how to create an encrypted virtual disk volume using TrueCrypt and explained how to use it in conjunction with your customized live CD—for example, by mounting it over the live CD default user's Documents folder.

This month, I wrap up the endeavor with some odds and ends, including my thoughts on network dæmons and firewall scripts, on plausible deniability scenarios and why you probably don't need to bother trying to enable user logins with your live CD.

To Log On, or Not to Log On

As I was wrapping up Part II last month, I mentioned that the default account on Ubuntu live CDs, ubuntu, has no password. And, I implied that next time we might talk about "fixing" that.

At least, that's what was lurking at the back of my mind when I wrote the article. Why not, I wondered, set a password for the ubuntu account on the live CD, and configure GDM to start with a logon prompt?

But, the more I think about this, the less I think it's worth the effort. Let me take a few minutes to discuss why that may be.

Security is all about risk management. What controls can be employed to reduce or eliminate the risk of some bad thing happening? Is that risk likely enough to be worth the trouble of the controls? Does the control itself add *other* risks?

We set up accounts with passwords in order to mitigate the risk that some unauthorized person may gain access to system resources or data. On a system that has multiple users, or that is reachable over networks (especially if it's always connected), this is a serious risk.

But on an ephemeral system, such as a live CD with no hard drive of its own, there are better ways to protect access and data. Access is controlled easily by setting up your live CD so that when booted, it doesn't run any network services to which unauthorized users can connect. You can protect your personal data by keeping all of it elsewhere—for example, on a TrueCrypt-encrypted volume on a USB drive, which I showed you how to set up last month.

The sad fact of the matter is that anybody with physical access to your live CD in any form (burned onto a CD, or stored as an ISO image) can simply recustomize it using the same procedure you used and delete the password field in your custom user account's entry in /etc/passwd. Or, more likely, the attacker can skip customizing it all together and simply mount and copy the interesting parts of your squashfs image. No boot, no login!

This is the same reason that with "normal" Linux systems (hard drive or Flash-based systems) physical access is so important. Unless you're using encrypted system volumes, anybody with physical access to your Linux computer can reboot from a live CD, mount your hard drive, and copy and alter system files at will.

So again, the best way to protect the data you use with your live CD is to store it on an encrypted volume—either one small enough to fit on your live CD image (assuming you can live with read-only access to that data), or one stored on a USB drive. And, the best way to control access to your live CD system is not to run any network services.

Network Services and Ubuntu Live CDs

The good news here is that by default, on Ubuntu Desktop 7.10, there are only two network dæmons that run by default: the CUPS printing system and the Avahi dæmon, which is part of the Zeroconf system for automated file/music sharing and Voice-over-IP client discovery. And, of these two things, only Avahi is problematic, because CUPS listens only on the local loopback interface—by default, CUPS doesn't accept connections from nonlocal processes.

How "problematic" is Avahi? Actually, not necessarily very much so at all. Truth is, I'm not aware of any critical security vulnerabilities in Avahi. However, it *is* the only thing standing between you and a system that accepts no foreign connections! If you disable Avahi, your system will be *completely* unresponsive to port scans and security scans. If a house with locked doors is secure, a house with no doors at all is *extremely* secure.

Disabling Avahi is a very simple step to add to the process of customizing an Ubuntu live CD (see the Appendix for the commands described in Part I of this series). Once you've mounted your ISO, mounted your

squashfs image, and chrooted yourself into your live CD image's root filesystem (steps 00 through 12 in the Appendix), you need to issue only one command:

```
12.5-# update-rc.d -f avahi-daemon remove
```

You could, of course, also run a personal firewall script to be extra safe. But in this context (bootable-CD-based desktop), I'm not convinced it's worth the trouble, if it's possible to run without network dæmons in the first place. First, you can't necessarily be sure what your local IP address and Ethernet interface names will be, if you're going to run your live CD from random hardware, such as coffee-shop workstations. This makes it difficult to write things like anti-IP-spoofing rules.

Second, neither Ubuntu nor Debian (on which it's based) has a native firewall script service. If they did, you simply could add your firewall rules to an existing script somewhere in /etc, as with RHEL and SUSE. Instead, with Debian and Ubuntu, you either need to create your own startup script or install additional software like Firestarter on your live CD image, *and* configure that software on some *other* system the way you want it on the live CD, *and* copy the resulting configuration file(s) over to your live CD's filesystem.

Again, in this context, going network-dæmon-free is much simpler. Note, however, that this is one of very, very few situations in which I recommend against using iptables for local protection. Ordinarily, that is an important protection!

Plausible Deniability, Live CDs and TrueCrypt

Suppose you're a human-rights activist working in a country with a paranoid, totalitarian government, and you use a live CD for sending factual reports to the press about local civil-rights abuses. Suppose further you want to prevent your live CD or the accompanying TrueCrypt volume you keep on your USB Flash drive from being used as direct or circumstantial evidence that you've been "committing treason".

I have three easy suggestions for you. First, *don't* customize your live CD; instead, use a *standard* live CD from Ubuntu Desktop, Linux Mint or whatever your favorite distribution is. If you've got a lot of

Expert included.

Dave is the Silicon Mechanics expert who heads up the Product Development team here at Silicon Mechanics, and he's proud to announce a selection of **virtualization** solutions. You probably don't need Dave to give you a list of reasons to consider **virtualization** — server consolidation, energy efficiency, optimized resource utilization, improved operational flexibility, and better disaster recovery and security — but what Dave can't wait to tell you is what makes this Silicon Mechanics **virtualization** product so special.

intel Xeon® inside™
Powerful. Efficient.

Start with the quality and reliability of the Rackform iServ R450-VE, with four Intel® Xeon® Processors 7300 Series, built on a carefully engineered and solidly crafted Intel EPSD platform. Add leading virtualization software technology, VMware® ESX Server. Combine with the team of experts responsible for industry-leading service and technical support that distinguish Silicon Mechanics. It's Dave's foolproof recipe for virtualization success.

When you partner with Silicon Mechanics, you get more than solid Intel engineering and VMware virtual infrastructure technology — you get an expert like Dave.

Find the selection of Silicon Mechanics virtualization solutions at www.siliconmechanics.com/virtualization.

Silicon Mechanics and the Silicon Mechanics logo are registered trademarks of Silicon Mechanics, Inc. Intel, the Intel logo, Xeon, and Xeon Inside, are trademarks or registered trademarks of Intel Corporation in the US and other countries. VMware is a registered trademark of VMware, Inc.

SILICON MECHANICS
visit us at www.siliconmechanics.com
or call us toll free at 866-352-1173

Ubuntu 8.04 and Debian Live

There are two things that are mostly out of scope for this article, but worth bringing to your attention nonetheless. The first is that Ubuntu Desktop 8.04 will have been available for at least a month by the time you read this, but it was still in beta testing at the time I wrote this article.

Needless to say, I didn't have enough time to do a comprehensive check of my live-CD-customizing procedure (see Appendix) against Ubuntu 8.04. But, I did mess around with it enough to determine that my procedure is *probably* 100% compatible and relevant with Ubuntu 8.04.

The only strangeness I encountered is that the squashfs image on the Ubuntu 8.04 live CD uses a new version of the squashfs file format. You won't be able to use earlier versions of Ubuntu to remaster 8.04 images unless you compile a kernel (or at least the squashfs kernel module) from raw source from kernel.org. Support for the new version of squashfs has not, to my knowledge, been backported to the kernel in Ubuntu 7.10.

Second, an alert reader recently pointed out to me that Debian, a distribution that normally ships on 21 CDs, now has a live CD version that is very easily customized. See Resources for a URL to the Debian Live Project's Wiki.

customized but mundane settings for your desktop manager, you can store them in an unencrypted loopback file image on your USB drive and manually mount it over /etc or your home directory.

In some places in this crazy world of ours, simply possessing a CD containing Tor, Privoxy and other privacy/anonymity tools is all the proof somebody needs that you're up to no good. Besides, this has the added advantage of being less work than using a customized live CD!

Second, use a TrueCrypt hidden volume. Keep only boring things in the nonhidden part of your TrueCrypt volume.

You can refer to the TrueCrypt link in the Resources section of this article for more information, but suffice it to say that this feature takes advantage of the fact that once you create a TrueCrypt volume, its size remains constant. Empty space is filled with random data. Or, as the case may be, with random data plus a hidden volume that is impossible to distinguish from the random data, except by someone who knows both that the TrueCrypt volume contains a hidden volume and the hidden volume's passphrase.

My third suggestion is to rename the TrueCrypt binary you'll need to keep on your USB drive (because you're using a stock Linux live CD), and while you're at it, make sure your TrueCrypt volume (or volumes) isn't named conspicuously. Both the TrueCrypt binary itself (which, by default, is named truecrypt) and TrueCrypt volumes can be called whatever you like.

So, there's nothing to stop you from renaming truecrypt to something inconspicuous like cooking-schools.dat, and your TrueCrypt volume file to checkered-pants-sources.dat. Anybody who executes cooking-schools.dat will, of course, immediately see the TrueCrypt GUI, but why would someone try to execute what appears to be a data file? Note that the *only* feasible way to identify a TrueCrypt volume as such is to try to mount it with TrueCrypt.

By telling you these three things, naturally I trust you'll use this knowledge for good, and not for evil—for example, by committing real kinds of treason that don't involve simply speaking the truth.

Parting Notes

In this series of columns, I've really only gotten you started down the custom live CD path, but hopefully well enough for you to figure out more ways to use and customize Ubuntu live CDs on your own. Here are a few things you might have fun figuring out:

- Pre-installing and preconfiguring Firefox plugins, such as NoScript and RefControl.
- Incorporating dmcrypt for encrypted system volumes.
- Pre-installing and preconfiguring the bittorrent and bittorrent-gui packages.
- Customizing GNOME for maximum elite-looking-ness.

Whether you're an intrepid human-rights activist or simply someone with a need for a maximally portable Linux system, live CDs are a handy, simple and potentially safe way to run Linux without changing or leaving any trace of itself on the hardware on which it's run.

By the way, I'm taking next month off from the Paranoid Penguin (though not from being paranoid, of course), but I'll be back in two months. Until then, be safe!■

Mick Bauer (darth.elmo@wiremonkeys.org) is Network Security Architect for one of the US's largest banks. He is the author of the O'Reilly book *Linux Server Security*, 2nd edition (formerly called *Building Secure Servers With Linux*), an occasional presenter at information security conferences and composer of the "Network Engineering Polka".

Appendix

Here's the complete procedure I described in Part I for adding and removing packages in a custom Ubuntu live CD, in the form of a raw list of all commands described in that article. The $ prompts indicate commands executed as an unprivileged user; the # prompts show commands that are executed by root:

```
00-$ dd if=/dev/cdrom of=./ubuntu-7.10-desktop-i386.iso

01-$ mkdir -p ./isomount ./isonew/squashfs ./isonew/cd
➥./isonew/custom

02-$ sudo mount -o loop ./ubuntu-7.10-desktop-i386.iso ./isomount/

03-$ rsync --exclude=/casper/filesystem.squashfs -a ./isomount/
➥./isonew/cd

04-$ sudo modprobe squashfs

05-$ sudo mount -t squashfs -o loop
➥./isomount/casper/filesystem.squashfs ./isonew/squashfs/

06-$ sudo rsync -a ./isonew/squashfs/ ./isonew/custom

07-$ sudo cp /etc/resolv.conf /etc/hosts ./isonew/custom/etc/

08-$ sudo cp /etc/apt/sources.list ./isonew/custom/etc/apt/

09-$ sudo chroot ./isonew/custom

10-# mount -t proc none /proc/

11-# mount -t sysfs none /sys/

12-# export HOME=/root

13-# apt-get remove --purge `dpkg-query -W --showformat='${Package}\n'
➥|grep openoffice`

14-# apt-get remove --purge `dpkg-query -W --showformat='${Package}\n'
➥|grep gimp`

15-# apt-get update

16-# apt-get install tor privoxy

17-# apt-get dist-upgrade

18-# apt-get clean

19-# rm -rf /tmp/*

20-# umount /proc/

21-# umount /sys/

22-# exit

23-$ chmod +w ./isonew/cd/casper/filesystem.manifest

24-$ sudo chroot ./isonew/custom dpkg-query -W --showformat='${Package}
➥${Version}\n' > ./isonew/cd/casper/filesystem.manifest

25-$ sudo cp ./isonew/cd/casper/filesystem.manifest
➥./isonew/cd/casper/filesystem.manifest-desktop

26-$ sudo mksquashfs ./isonew/custom
➥./isonew/cd/casper/filesystem.squashfs

27-$ sudo rm ./isonew/cd/md5sum.txt

28-$ sudo -s

29-# cd ./isonew/cd

30-# find . -type f -print0 | xargs -0 md5sum > md5sum.txt

31-# exit

32-$ cd ./isonew/cd

33-$ sudo mkisofs -r -V "Ubuntu-Live-PrivateSurf" -b
➥isolinux/isolinux.bin -c isolinux/boot.cat -cache-inodes
➥-J -l -no-emul-boot -boot-load-size 4 -boot-info-table
➥-o ~/Ubuntu-Live-7.10-PrivateSurf.iso .
```

Resources

Ubuntu Community Wiki Page on How to Customize Ubuntu Live CDs: **https://help.ubuntu.com/community/LiveCDCustomization**

Information about Zeroconf and Avahi on Ubuntu: **https://help.ubuntu.com/community/HowToZeroconf**

The Hidden Volume Information Page on the Official TrueCrypt Web Site: **www.truecrypt.org/hiddenvolume.php**

The Debian Live Wiki: **wiki.debian.org/DebianLive**

Download Site for the NoScript Plugin for Firefox: **https://addons.mozilla.org/en-US/firefox/addon/722**

Download Site for the RefControl Plugin for Firefox: **https://addons.mozilla.org/en-US/firefox/addon/953**

KYLE RANKIN

Migrate to a New Hard Drive

Storage needs always seem to grow, but when you are ready to upgrade to a new hard drive, how do you transfer all those files? Read on for a tried-and-true method to migrate all your old files to a new drive.

In another article in this issue of *Linux Journal* [page 84], I talk about my experiences with the new solid state drive (SSD) I installed on my laptop. One of the things I didn't mention in the article was how I transferred all my data and settings to the new drive. There are a number of ways to solve this problem. For instance, you could image the old drive onto the new one and then grow the last partition to fill up the presumably larger disk (which wouldn't work for me, as my new SSD actually was substantially smaller than the old drive). Other people just re-install their OS every time they get a new drive and then transfer their /home directory and other settings, but I've always had just enough custom programs and settings on my laptop for that method to be a pain. You also could use rsync with certain flags to migrate the files, and although I do like that method for network transfers, for local transfers, it can be a hassle, because it first must scan through the entire drive before it begins.

I've done many hard drive migrations during the years with a tried-and-true combination of find piped to cpio. I like this method because it uses common tools that are sure to be installed, it starts immediately and doesn't need to scan the drive, and with the right flags, it correctly can handle (and avoid) special filesystems, such as /proc, /sys and so on. So far, it hasn't failed me, and this migration was no exception. However, this time, I did run into a few gotchas that I will talk about shortly. First, onto the basic steps.

You can use any partitioning tool that works for you—from fdisk to qtparted.

1. Move to a Safe State

You don't want files to be changed as you are copying them, so you don't want to do this migration from your normal desktop environment. Typically, I boot in to a rescue disk like Knoppix, so that the filesystem stays frozen. Other times, I simply switch to single-user mode, so most files won't change. For desktop systems, I generally just connect both drives directly to the system, and for laptops, I use a USB hard drive adapter, so that both can be connected at the same time. For my last migration, I didn't happen to have a USB adapter for a 1.8" drive, so I transferred the data to an intermediate drive first, then installed the new drive and transferred again.

2. Partition Your New Drive and Format the Filesystems

You can use any partitioning tool that works for you—from fdisk to qtparted. This may sound obvious, but make sure that you allocate plenty of room to fit your existing data, and if you move to a larger hard drive, plenty of room to grow. Once you partition the drive, use mkfs or your preferred formatting tool to write a filesystem to each partition (or mkswap for the swap partition).

3. Mount the New Partitions

Create mountpoints under /mnt for the new partitions you have created. For my example, I have a root partition at /dev/sdb1 and a home partition at /dev/sdb3, so I would type as root:

```
mkdir /mnt/sdb1
mkdir /mnt/sdb3
mount /dev/sdb1 /mnt/sdb1
mount /dev/sdb3 /mnt/sdb3
```

If you run this from a rescue disk, you also need to make sure your source partitions are mounted as well.

4. Run the find | cpio Spell

Now this spell doesn't have a lot to it, but it's funny how you memorize scripts like this over the years after using them and passing them along to friends. First, change to the root level of the partition you want to copy and then execute the command as root. So, to migrate my root partition from single-user mode, I did the following:

```
cd /
find ./ -xdev -print0 | cpio -pa0V /mnt/sdb1
```

To migrate from a rescue disk, the command is almost identical, but you change to the mountpoint of

the source partition instead (I mounted it at /dev/sda1):

```
cd /mnt/sda1
find ./ -xdev -print0 | cpio -pa0V /mnt/sdb1
```

The find command searches through the entire root partition for files and directories. The -xdev flag tells find to stay within the current mounted filesystem. Otherwise, when find gets to /home, it would copy the contents of that directory as well and potentially fill up the new partition. It then passes the files to cpio, which places them under my new mountpoint while preserving permissions, symlinks and other settings. The cpio command also outputs one dot for each file it copies, so you can have some sense of its progress. However, what I typically do is go to another terminal and monitor the output of df so I can watch it grow:

```
watch df
```

Once the first find | cpio command completes, repeat it for each of your other partitions. In my example, if I were in single-user mode, I'd do the following:

```
cd /home
find ./ -xdev -print0 | cpio -pa0V /mnt/sdb3
```

If I were using a rescue disk, I'd do this:

```
cd /mnt/sda3
find ./ -xdev -print0 | cpio -pa0V /mnt/sdb3
```

5. Update fstab

What you do during this step will vary a bit, depending on how you set up your partitions. If you moved your partition layout around, you need to edit the /etc/fstab file on your new root partition so that it reflects the new drives you have set up.

Traditionally, this has been a simple step for me, because I try to order the partitions the same and generally don't have to touch fstab, but on this last migration, I had to add an extra step due to Ubuntu's use of UUIDs to reference partitions. A lot of modern distributions don't refer to partitions by their device name. Instead, a unique identifier called the UUID is assigned to each partition. If you see UUID=longstringofhex in your /etc/fstab, this means you.

Now, you have two choices here. The first choice is to change all these UUID lines to reference the actual device. This will work, and is less prone to typos that will make the system not boot, but you will lose the advantage of UUIDs. The other method is to reference the UUIDs for your new partitions and put them in place of the old UUIDs. You can find the list of disk-to-UUID mappings under /dev/disk/by-uuid:

```
greenfly@minimus:~$ ls -l /dev/disk/by-uuid/
total 0
lrwxrwxrwx 1 root root 10 2008-04-06 16:00
 ➥634719fd-a6da-4fee-8646-0d485d7681db -> ../../sda2
lrwxrwxrwx 1 root root 10 2008-04-06 16:00
 ➥665d7008-fde9-4055-8af9-483697acb005 -> ../../sda1
lrwxrwxrwx 1 root root 10 2008-04-06 16:00
 ➥cf3892fd-e3d8-446f-8552-4c633be9c382 -> ../../sda3
```

Of course, you always could choose a hybrid of the two approaches, and set the hard device names in the fstab for the first boot, and then once you have confirmed the system boots, you then can update fstab with UUIDs.

6. Update GRUB

As with fstab, if you changed your partition layout, you need to update your GRUB configuration under /boot/grub/menu.lst (or on some systems, in /boot/grub/grub.conf) to reflect your changes. Also, GRUB can reference drives by UUID, so if you see references to UUID in the GRUB configuration file, be sure to update it to reflect the new values. Once the file has been updated, chroot into your new root partition's mountpoint and then run grub-install:

```
chroot /mnt/sdb1 /usr/sbin/grub-install --recheck /dev/sdb
```

Change /mnt/sdb1 and /dev/sdb to reflect your new mounted root partition and its disk device, respectively. If the chrooted grub-install doesn't work, you typically can use your rescue disk (or single user) grub-install with the --root-directory option:

```
/usr/sbin/grub-install --recheck --root-directory /mnt/sdb1 /dev/sdb
```

7. (Optionally) Update the Initial Ramdisk

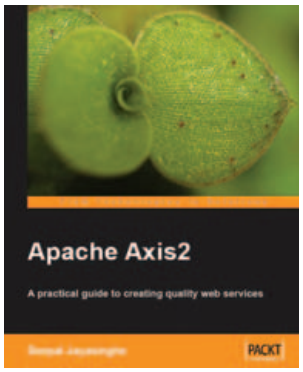
After I used my new system for some time, I noticed it wouldn't resume correctly from hibernation. It seemed like each time the swap partition would get corrupted. After some troubleshooting, I found that the root cause was a hard-coded resume device based on UUID that is put in the initial ramdisk for the machine. You may or may not run into this issue, depending on your Linux distribution, as each distribution manages its initrd differently. But, here is the fix for my Ubuntu system. I was able to find the offending reference to the old UUID in /etc/initramfs-tools/conf.d/resume. All I needed to do was update that file on the new drive to point to the new UUID for my swap partition, then run update-initramfs from the new system, and reboot.■

Kyle Rankin is a Senior Systems Administrator in the San Francisco Bay Area and the author of a number of books, including *Knoppix Hacks* and *Ubuntu Hacks* for O'Reilly Media. He is currently the president of the North Bay Linux Users' Group.

Plat'Home's OpenMicroServer

Although Plat'Home Co., Ltd., has been serving up Linux to the Japanese market since 1992, the company is just now bringing its OpenMicroServer product to North American shores via its US subsidiary. OpenMicroServer is a small, tough, easy-to-use, easy-to-configure, low-cost Linux server solution. It provides high reliability to customers who do not have much extra room and are likely to ignore the machine for weeks or months after installation. Key features include compact design (9"x4"x1.3"), integrated Power over Ethernet, stable long-term operation up to 122°F when using PoE functionality (based on a 625-day endurance test), 400MHz AMD Alchemy (MIPS) processor, two Gigabit Ethernet ports, one 100MBit Ethernet (PoE capable) port, two USB 2.0 ports and two serial ports. Plat'Home is proud of its product's "Japanese characteristics", meaning it doesn't stand out, and it doesn't complain. It just gets the job done.

www.plathome.com

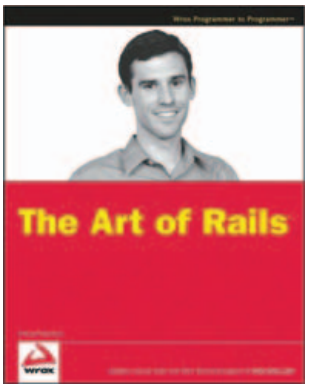


Deepal Jayasinghe's *Apache Axis2* (Packt Publishing)

Quicker than most to find a new and interesting open-source topic, Packt Publishing has released Deepal Jayasinghe's new book *Apache Axis2*. Apache Axis2 is a core engine for Web services with two different implementations: Apache Axis2/Java and Apache Axis2/C. This book takes readers through the basics of Web services and Axis2, as well as details of Axis' architecture. It is a step-by-step practical guide that uses many real-life examples. Some of the topics covered include installation, AXIOM, pipes and interceptors, module concepts, session management and more. The book assumes familiarity with Web standards, such as SOAP, WSDL and XML parsing.

www.packtpub.com

Edward Benson's *The Art of Rails* (Wrox)



Author Edward Benson's intent with his new book *The Art of Rails*, published by Wrox, is to pick up where the API leaves off and explain how to turn good Rails code into beautiful Rails code: simple, effective, reusable and evolvable. Benson wants you to think like a Rails developer with quality, elegance and maintainability in mind. *The Art of Rails* blends design and programming, identifying and describing the very latest in design patterns, programming abstractions and development methodologies that have emerged for the modern Web. Readers will learn topics such as techniques for organizing code between and within Model, View and Controller; how to think like a REST-based developer and use Rails 2.0 to translate these thoughts into code; advanced Ruby and meta-programming; design patterns for AJAX, Web APIs, HTML decomposition and schema development; and behavior-driven development. The book is designed to advance the skills of developers already familiar with Rails.

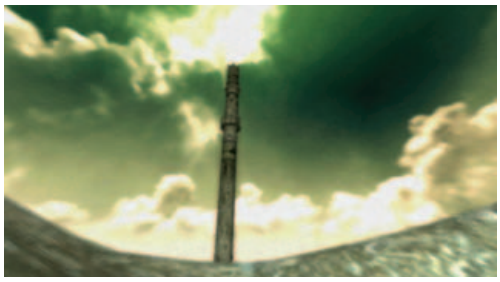
www.wrox.com

FreeIPA

Version 1.0 of the FreeIPA Project is now official. FreeIPA is an integrated security information management solution that combines Linux (currently Fedora, Fedora Directory Server, MIT Kerberos and NTP), with a Web interface and command-line administration tools. Currently, FreeIPA supports identity management, and plans to support policy and auditing management will follow in future releases. The project developers state that the use of standard protocols, such as LDAP and Kerberos, allows for easy integration of other OSes into an IPA realm for centralized identity management. The developers also encourage testing and deployment of FreeIPA and are seeking feedback from the field.

www.freeipa.org

Paradox Interactive's *Penumbra: Black Plague*



Announcing more new games on the Linux platform is such a treat. The game developer Paradox Interactive and the two-man Swedish developer team, Frictional Games, have released a Linux version of its popular game *Penumbra: Black Plague*. The *Penumbra* series, which includes the new *Penumbra: Black Plague* and its prequel *Penumbra: Overture*, is a first-person adventure game that focuses on story, immersion and puzzles. Instead of using violence to progress, players must use their wits to guide Philip on his quest to unravel the past. Paradox says that *Penumbra* "is very different from other adventure games". The games feature a 3-D engine that utilizes cutting-edge technology, and it has an advanced physics system that creates a new level of environmental interaction. Players can open drawers, pull levers, pick up objects and more, using natural mouse movements creating a highly interactive and dynamic game world. The next game in the series, *Penumbra: Requiem*, is due out in Summer 2008, and it also will offer a Linux version.

www.PenumbraBlackPlague.com

Trusted Computer Solutions' Security Blanket Enterprise Edition

Trusted Computer Solutions has released version 2.0 of Security Blanket application, which also now includes an enterprise edition. Security Blanket is an automated system lockdown and security management tool for Linux. Security Blanket 2.0 enables system administrators to configure and enhance the security of their Linux platforms automatically by simplifying the hardening process that must be undertaken on a regular basis to meet security compliance requirements. Various security guidelines are included. Other key advancements in version 2.0 include the ability to manage local and remote Linux servers from a centralized management console and a Web-based interface for remote access. The application supports Red Hat Enterprise Linux versions 4 and 5, CentOS versions 4 and 5, and Oracle Enterprise Linux versions 4 and 5, and it runs on SELinux, enforcing the default targeted policy provided with the operating system.

www.trustedcs.com

Tripwire Enterprise 7.1

The idea behind Tripwire Enterprise, upgraded recently to version 7.1, is to keep your IT expertise from walking out the door. The solution maximizes the use of IT expertise throughout an organization by capturing and replicating this knowledge across all IT systems. It ensures IT configuration integrity across the entire IT infrastructure and manages internal and external policies. Two key features include Golden Policies and Remediation Advisor. Golden Policies capture and replicate gold configurations and act like a "consultant in a box" that maximize the value of IT experts by replicating their IP (optimal configuration settings) across the IT infrastructure. In addition, Golden Policies help ensure IT personnel are proactively made aware of any configuration drift. The Remediation Advisor functionality provides step-by-step remediation, based on a wide variety of external IT resources, keeping staff from hours of research and reducing the time and effort needed to remediate problems.

www.tripwire.com

Ixonos Mobile Television Reception Solution

In a move to help device manufacturers create novel mobile services of interest to large consumer audiences, Ixonos has developed a television reception solution for its Linux-based mobile platform. The solution enables reception of, among others, DVB-T and DVB-H transmissions and also can repeat them over a local area network. The video player used for TV reception also is the multimedia player for the entire platform and is based on the open-source MPlayer. Ixonos claims to be among the first companies to bring DVB-H reception to a Linux-based smartphone. The latest H.264 and AAC technologies are used as video and audio codecs.

www.ixonos.com

Please send information about releases of Linux-related products to newproducts@linuxjournal.com or New Products c/o *Linux Journal*, 1752 NW Market Street, #200, Seattle, WA 98107. Submissions are edited for length and content.

Fresh from the Labs

Pondus

(www.ephys.de/software/pondus)

First up this month is a weight manager, Pondus. According to its Web site: "Pondus is a personal weight management program written in Python and Gtk+2 released under the GPL. It aims to be simple to use, lightweight and fast. The data can be plotted to get a quick overview of the history of your weight and is stored in XML files for easy access and modification with other programs." Simple it is, indeed, and the installation isn't too shabby either. Pondus lets you track your weight over a period of time and displays your progress with a graph. It also switches between metric and imperial measurements.

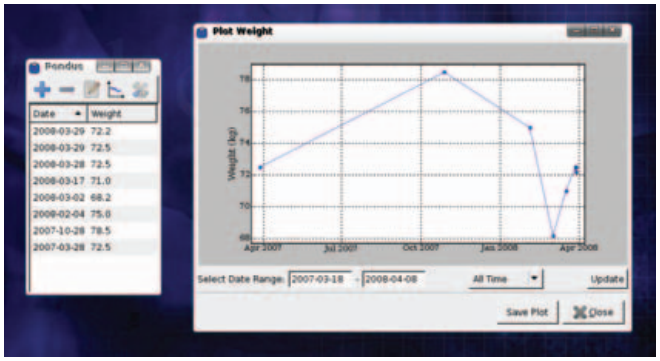


Figure 1. The Lovely Minimalism of Pondus and My Weight Chart

Installation In terms of dependencies, you need a few Python-related libraries installed before you can start. Whenever you compile something, the installer invariably asks for the development files, so make sure you install the python-dev files first. If you still run into problems, some Googling turned up a posting in a forum about some of the packages on which Pondus depends:

- python 2.4.4-6
- python-gobject 2.14.1-1
- python-gtk2 2.12.1-1
- python-matplotlib 0.90.1-2
- python-support 0.7.6

Once the dependencies are out of the way, download the source package from the project's Web site, extract the contents and open a terminal in the new folder.

As root, enter the command:

```
# python setup.py install
```

If all goes well, Pondus should compile and even install itself in your Applications menu. On my system, I found a new menu entry under Utilities→Pondus. If you can't find Pondus on your menu, you can start it by entering the command `pondus`.

Usage Pondus is *very* minimalist, but that's not necessarily a bad thing. Upon entering Pondus, you'll see a small window with five buttons. The first adds a line of data—that being your weight and the date for entry. The second removes a line, and the third edits a line of data. Once you have entered some weights and times, you then can display it as a graph by clicking the fourth button. If you want to switch between pounds and kilograms, the fifth button opens the settings window and lets you change that (this is one of only two options, the other is to remember the window size).

Tracking your progress is really what Pondus is about though, so you will want to jump over to the graph section—the fourth button, or Plot data. Click the button, and a window titled Plot Weight appears with a neat line graph representing your weight over a period of time. If you look at the bottom right, there's a drop-down box with All Time written in it. This allows you to filter out the rest of the information to what you've had over the past year, or just the past month. If you want to filter your time to something more specific, on the bottom left are two fields called Select Date Range. Enter the start date you want to see in the first field and the end date in the second field, click Update on the far right, and the graph will update with the selected information. For those of you who want to save a copy of what your progress has been, clicking the Save Plot button at the bottom lets you save your graph as a .png file. I'm a weedy little runt myself, so I'm not trying to lose weight, but rather *gain* it, so check my screenshot for an example (which is in kilograms by the way, I don't way 72 pounds).

Overall, Pondus is a very simple and clean application that will appeal to many new PC users, as it sits in a nice and small window and doesn't baffle you with a zillion options. I'm guessing that Pondus probably will add more features over time, but hopefully not too many, as doing so might alienate its target audience. It's a lovely, neat little program.

BeeDiff

(www.beesoft.org/beediff.html)

BeeDiff (beediff) is a GUI program for comparing text differences between two files, and any differences will be highlighted in different colors depending on the type of difference. Any differences found then can be deleted or copied between files. BeeDiff is developed with new Qt4 libraries, and as such, it's very quick and lightweight. It also happens to be quite easy to install, which is another bonus.

Installation For installation purposes, you have two choices: an i586 binary or a source tarball.

If you choose the binary, first download the provided tarball and extract the contents to a folder of your choice.

Then, as root or using sudo, copy the binary to /usr/bin or your preferred binary directory to run BeeDiff system-wide.

Next, copy the included icon, beediff.png, to /usr/share/icons or whichever icon directory you prefer.

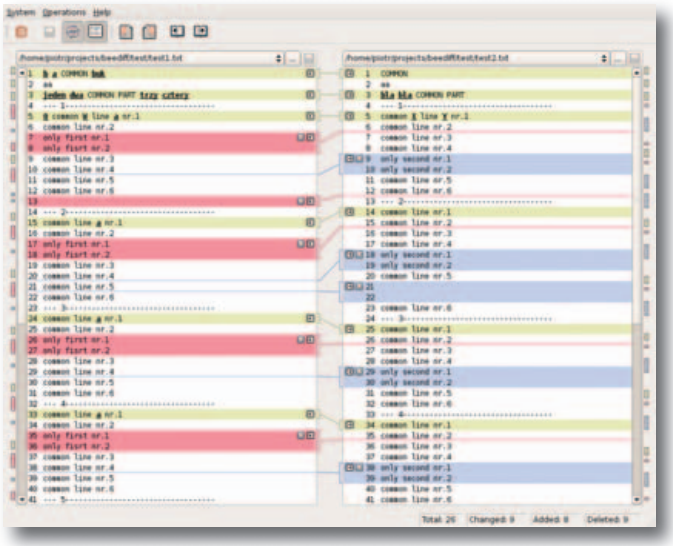


Figure 2. BeeDiff—the Insomniac Coder's Best Friend

If you would rather run BeeDiff locally, you can run it by entering:

```
$./beediff
```

from whichever directory you've extracted it to. If your system is set up right, you also might be able to run it simply by clicking on it in your file manager.

If you would prefer to compile it (if you don't have an i586-compatible processor, for example), that's also very easy. Simply download and extract the tarball provided on the Web site, and open a terminal in the new beediff folder.

As root or sudo, run the command:

```
# ./install
```

and the script will do all the compiling and installation for you.

Once this is done, entering `beediff` at the command line should launch the program.

At first, I had an error running the binary or compiling it, and it was due to having old libraries installed. When I went to run the binary, I got this:

```
nhoj@ubuntu:~/src/beediff_1.5_i586/beediff$ beediff
beediff: symbol lookup error: beediff: undefined symbol:
_ZN10QBoxLayout10setSpacingEi
```

And, I got this with the source compilation:

```
QBtSeparator.cpp:139: error: 'const class QColor'
 ➥has no member named 'darker'
QBtSeparator.cpp:142: error: 'const class QColor'
 ➥has no member named 'darker'
QBtSeparator.cpp:145: error: 'const class QColor'
 ➥has no member named 'darker'
make: *** [tmp/QBtSeparator.o] Error 1
install: cannot stat `beediff': No such file or directory
nhoj@ubuntu:~/src/beediff$
```

BeeDiff requires at least Qt 4.3—Piotr, the author was using 4.3.2. Install the latest version you can along with the development libraries. Once I installed these, BeeDiff ran, binary and source included.

Usage BeeDiff is pretty much geared for comparing two files that have the same origin, so comparing pieces of code and scripts will be the best use of BeeDiff's abilities. Fire up BeeDiff and once inside, you'll notice two main panes. Here, you will load a text file into each one. The left pane is the original, and the right pane is for comparing against it. On the top right of each pane is a button to browse for the file you want to load.

Once loaded, any different lines will be highlighted in different colors:

- Red: lines that have been deleted.
- Blue: lines that have been added.
- Yellow-green: lines that have been changed.

After analyzing what lines are different, you then can take several actions. Along the toolbar to the right (and in the menu under Operations) are four icons: Remove all from left, Remove all from right, Merge all to left and Merge all to right. The Remove buttons obviously delete the text in question, but the Merge buttons let you grab any divergent lines and copy them across to the other file and save it—very handy.

BeeDiff is another no-nonsense application that does what it says on the tin and doesn't pretend to be anything else. This program should save scripters and coders many a late night of headaches and may prove to be quite handy in this time of common allegations between companies and projects of "stolen code".

youtube-dl

(www.arrakis.es/~rggi3/youtube-dl)

First up, this is a YouTube downloader. Not very interesting, as everyone has used them before, right? Indeed, but a few months back, YouTube changed some embedding options, rendering most of these lovely tools useless. Well, this little script has been updated and downloads YouTube videos just fine. To install, simply save the URL provided onto your hard drive—that's it! Make sure you save the filename as is though, not with an .html extension.

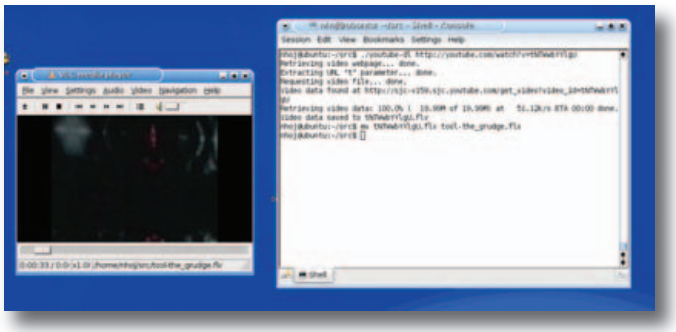


Figure 3. youtube-dl—a Groovy Command-Line Utility to Save YouTube Videos to Hard Drive

To use it, open a console in the directory where you saved the file. Make sure you can execute the file by entering:

```
$ chmod u+x youtube-dl
```

Now you're ready to go! Find your favorite YouTube clip and copy its URL. Go back to your console and enter `./youtube-dl`, and paste the clip's URL after it, like so:

```
$ ./youtube-dl http://youtube.com/watch?v=tNTWwbYYlgU
```

youtube-dl now saves it to your hard drive, and it even has a spiffy text-based progress monitor. Once downloaded, the filename just looks like random garbage. Rename the file to *nameofyourvideo*.flv (the .flv extension is the most important part), and open it with a strong video player such as VideoLAN or MPlayer.

getmsmp3

(MySpace Downloader, psydev.sourceforge.net/new/misc/getmsmp3)

This clever little script runs on anything that has a Net connection and Perl. It grabs MP3s from a MySpace page and saves them locally. Like youtube-dl, this is not new, but it automates a number of things and does it locally from your hard drive without weird requirements. The best part is that it grabs all the songs and saves them in the format of [band] - [song title].mp3 automatically. Like youtube-dl, simply save the project file to your hard drive and flag it as executable, like so:

```
$ chmod u+x getmsp3
```

Now, simply run the script and enter the URL of the band you want after the command:

```
$ ./getmsmp3 http://www.myspace.com/soundskp
```

Of course, we can't encourage you to download illegally, so I've provided you with the URL of our own band, which you're welcome to download (slightly redundant though, as we've provided the option to download our files anyway).

Like youtube-dl, this is not new, but it automates a number of things and does it locally from your hard drive without weird requirements.

FoxyTag

(www.foxytag.com)

Here's a project I'm dying to see the outcome of—a free (as in beer) speed camera warning system designed to run across a large range of mobile phones and GPS devices. FoxyTag is a collaborative system designed to encourage users to share speed camera data—the more users and feedback, the more reliable the system becomes. The system doesn't merely assume a speed camera is in one place either. Users have the options to report a permanent camera or the installation or removal of a mobile camera.

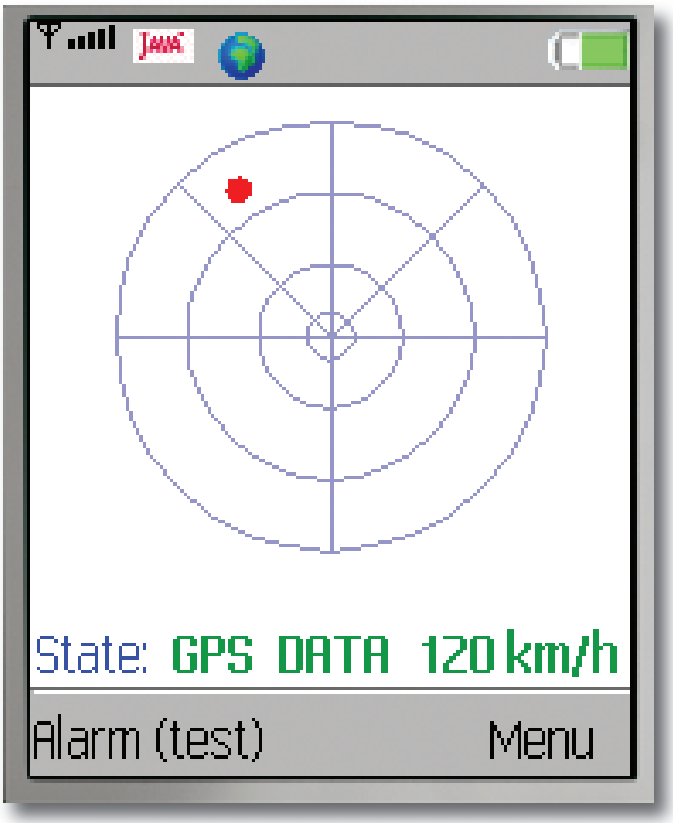


Figure 4. The People's Speed Camera Locator—FoxyTag

However, Michel Deriaz, the project's leader at Geneva University, isn't trying to promote speeding or unsafe driving. According to FoxyTag's Web site:

> FoxyTag motivates neither speeding nor any other risky behavior, but allows drivers to concentrate on the road instead of having their eyes fixed on the speedometer, by fear of being flashed. We observe that drivers tend to brake suddenly when they see a speed camera (even if they are not speeding), which can provoke traffic jams or even accidents (chain collisions or slidings, like in this video [see the Web site for the link]). FoxyTag signals in advance the presence of speed cameras, so that drivers have enough time to check their speed and adapt it if necessary.

As for mobile phones, any Java mobile phone with MIDP 2.0, CLDC 1.1 and Bluetooth should be compatible. For GPS systems, any Bluetooth GPS should be compatible (including GPS modules of some navigation systems), and Michel recommends a Sirf III GPS. Unfortunately, I have neither. Hopefully, we can rustle up the needed hardware and cover this project further. I'd love to see the results of this one.■

John Knight is a 23-year-old, drumming- and climbing-obsessed maniac from the world's most isolated city—Perth, Western Australia. He can usually be found either buried in an Audacity screen or thrashing a kick-drum beyond recognition.

Cost Effective 1U Servers Designed for Data Centers

with Customized Programs from ZT Systems

ZT Systems®

Your data center hardware requirements are growing every day.

With ZT Systems, you get validated, high-quality servers featuring the Quad-Core Intel® Xeon® processor 5400 Series for maximum performance and efficiency – but that's only part of the solution. ZT Systems now offers customizable programs designed to enable the ultimate in value and ease of implementation for organizations operating infrastructure-scale data centers.

ZT Systems 1101Ri 1U Rack Server

Affordable Single-Socket Solution

Starting at Only

$999

ZT1101Ri-82-C00001

- Quad-Core Intel® Xeon® Processor X3220 (2.40G,8M,1066MHz)
- 2GB DDR2 800 Unbuffered SDRAM
- Intel® 3200 Server Chipset
- (2) 320GB SATAII 32MB Hotswap Hard Drives
- Supports RAID 0,1,5,10 JOBD
- (1) Intel® Gigabit Ethernet Connection
- 400W high-efficiency power supply
- Red Hat® Enterprise Linux® 5
- Supports Intel® System Management Software 2.0
- 3 Year Limited Warranty and 24x7 Telephone support

ZT Systems 1202Ri 1U Rack Server

Dual-socket Data Center Server

Starting at Only

$1499

ZT11202i-84-C00002

- Quad-Core Intel® Xeon® Processor X5335 (1.86G,8M,1066MHz)
- 4GB DDR2 FB-DIMM ECC SDRAM
- Intel® 5000 Server Chipset
- (4) 500GB SATAII 32MB Hotswap Hard Drives
- Supports up to 4 SATA Drives & RAID 0,1,5,10 JOBD
- (2) Intel® 82563EB Gigabit Ethernet Ports
- 400W high-efficiency power supply
- Red Hat® Enterprise Linux® 5
- Supports Intel® System Management Software 2.0
- 3 Year Limited Warranty and 24x7 Telephone support

ZT Systems 1203Ri 1U Rack Server

Dual-socket server with 6 hard drives

Starting at Only

$1799

ZT1203Ri-84-C00003

- (2) Quad-Core Intel® Xeon® Processor X5405 (2.0G,12M,1333MHz)
- 2GB DDR2 667MHz ECC/Registered SDRAM
- Intel® 5100 Server Chipset
- (4) 120GB 2.5" SATA Hotswap Hard Drives
- Supports up to 6 (2.5") SATA Drives & RAID 0,1,5,10 JOBD
- (2) Intel® 82573 Gigabit Ethernet Port
- 500W high-efficiency power supply
- Red Hat® Enterprise Linux® 5
- Supports Intel® System Management Software 2.0
- 3 Year Limited Warranty and 24x7 Telephone support

ZT Systems 1201Ti 1U Twin Node Server

High density with two server nodes in 1U

Starting at Only

$2299

ZT1201Ti-84-C00004

- Single Quad-Core Intel® Xeon® Processor X5320 (2G, 12M, 1333MHz) per node
- 2GB DDR2 667MHz ECC/REG. SDRAM per node
- Intel® 5100 Server Chipset
- Single 320GB SATAII 32MB Hotswap Hard Drive per node
- Supports up to 2 Hard Drives per node RAID 0,1,5,10
- Intel 82571EB Dual Port Gigabit Controllers (2 Ports per node)
- Shared 600W high-efficiency power supply
- Red Hat® Enterprise Linux® 5
- IPMI 2.0 supported
- 3 Year Limited Warranty and 24x7 Telephone support

Contact Us Today to Learn About ZT Server Solutions and Data Center Programs Customized to Fit Your Needs

ZT Systems

(866) 888-6669

DataCenterSales@ZTSystems.com

www.ztsystems.com/datacenter

intel Xeon inside™

Powerful. Efficient.

©2008 ZT Systems. All rights reserved. Intel, the Intel logo, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries. The word mark "The Green Grid" and The Green Grid design mark(s) are service marks of The Green Grid Association.

HARDWARE

Over-the-Air Digital TV with Linux

A review of some current state-of-the-art digital TV tuner cards, focusing on how well they support digital television in Linux.

ALOLITA SHARMA

My ten-year-old TV set gave out recently. Being a Linux geek, I use a variety of open-source distributions on my notebook and desktop. So, the demise of my TV was a great opportunity to see if I could watch television on Linux instead of getting another TV set. It's just in time too, because over-the-air television broadcasts in the US will convert to all digital in February 2009. So, it was exciting to switch over to digital TV on my desktop.

In my quest to understand the state of digital TV (DTV) on Linux, I looked at digital TV tuner cards, antennas and accessories. I chose to set up MythTV, and by the end of the entire experience, I had a cool digital TV right on my Linux desktop with Picture-in-Picture and remote control. It was enough high-definition (HD) TV to turn me into a serious couch potato. I'm happy to report that Linux, along with hardware support from digital TV tuner cards, video cards, LCD monitors and rich software, such as MythTV, is ready for prime time.

I'm happy to report that Linux, along with hardware support from digital TV tuner cards, video cards, LCD monitors and rich software, such as MythTV, is ready for prime time.

Digital TV Environment

For this review, I used a PC with an Intel Core 2 Duo 3GHz, with 4GB memory, an NVIDIA 8800 GT graphics card, and a 750GB SATA hard disk. The display was a Samsung SyncMaster 245BW, with a resolution of 1920x1200. I ran Ubuntu 7.10, with all the latest updates, as my operating system. Using a powerful graphics card was essential for viewing HD programs on a high-resolution, wide-screen display.

A good antenna also is a critical component of the DTV setup. I tested both indoor and outdoor antennas, and discovered that the reception improved dramatically when using an outdoor antenna. The reception also improved with amplified indoor antennas. Standard indoor antennas performed adequately only when positioned very carefully. Because HDTV content is high-resolution (1920x1080), if your signal is weak, you may see a lot of artifacts. Frequent artifacts result in a very poor viewing experience. Hence, choosing an amplified indoor antenna or an outdoor antenna is recommended. Standard-definition TV (SDTV) is not as high-resolution and has greater tolerance for weak broadcast signals. But, even here a good antenna is essential.

I used MythTV (version 0.20.2) to view over-the-air DTV channels. MythTV is an open-source home entertainment software application for Linux and Mac OS. It has grown to become one of the most comprehensive, feature-rich platforms for viewing and recording television programming from over-the-air and cable broadcasts. I also used another open-source software application called tvtime (version 1.0.2) to view over-the-air analog NTSC channels.

So, what can we watch? There are a lot of over-the-air programs available in all major US metropolitan areas. For example, ABC, CBS, FOX and NBC offer standard and high-definition programming in addition to analog NTSC. In my location, the San Francisco Bay area, local public broadcasting stations (PBS) broadcast high-quality educational and topical content in HDTV format from 5pm to 6am each day.

Digital Tuner Card Evaluation

To evaluate various digital tuner cards, I tested first whether the hardware was recognized by Linux at boot time by checking the system logs. If it wasn't recognized, I had to find and build a device driver manually. Once this step was successful, I configured the tuner card within MythTV. As a part of configuration, MythTV scans for channels available in the broadcast area. On average, it took MythTV about seven minutes to find more than 25 digital channels. Once the channels were found, we were ready to watch digital TV.

I evaluated a range of digital tuners that included PCI, PCI Express and USB bus types.

pcHDTV HD-5500

The pcHDTV HD-5500 is a PCI card and is the only hardware designed and marketed to support Linux right out of the box. The HD-5500 supports digital (ATSC), analog (NTSC) and unencrypted cable TV signals. This low-profile PCI card provides a coaxial input for a TV antenna, a stereo audio output jack for analog TV and a nine-pin port for an adapter cable. The adapter cable provides inputs for S-Video and stereo audio, an RCA video output, and an IR transmitter (to control a set-top box). pcHDTV ships a CD with the HD-5500 tuner card, which includes drivers for

2.4 and older 2.6 kernels, command-line tools to capture and manipulate digital (ATSC) data streams and signals, and a version of the Xine video player customized to support HDTV.

This tuner card worked out of the box—configuration was as easy as installing the card into the PCI slot of my desktop test machine. Both the digital (ATSC) and analog (NTSC) tuners on the hardware were recognized right away and were fully functional at system bootup. Configuring the tuner as a "DVB DTV capture card (v3.x)" in MythTV was simple. If I had installed additional pcHDTV cards, I could have tested out multicard features, such as Picture-in-Picture (PiP) in MythTV. pcHDTV claims you can put up to four such cards in a single system.

The picture quality for both high-definition and standard-definition programs was superb. To top it off, the whole idea of having an end-to-end Linux DTV solution with a no-fuss setup and a great viewing experience is just plain cool. The HD-5500 is an ideal choice for a desktop Linux system. It would be nice to have a USB version for laptops as well. A remote control, as offered by several other DTV tuner products, would be icing on the cake.

I had the chance to catch up with pcHDTV's CEO, Jack Kelliher, by e-mail (see the Interview with Jack Kelliher, CEO and Cofounder of pcHDTV sidebar).

Pinnacle PCTV HD Pro Stick

Pinnacle's PCTV HD Pro Stick is a USB 2.0 tuner card that supports both digital (ATSC) as well as analog (NTSC) TV signals. This tuner card comes with a portable telescopic antenna, a mini-remote control with batteries, a USB extender cable and an A/V adapter cable with inputs for S-Video, composite video (RCA) and stereo audio (1/8" jack). An integrated infra-red (IR) receiver on the device supports the remote control. This tuner card is slightly bigger than a typical USB thumbdrive and has a coaxial TV antenna input on one end and a USB connector on the other. When plugged in to my system, the tuner's width blocked adjacent USB ports. However, using the USB extender cable that came with the tuner card circumvented this problem. The portable antenna that comes with this card is good enough for viewing nearby HDTV channels, but it always helps to have an amplified antenna.

Out of the box, this card is not supported under Linux. To get this card to work with Linux, I had to download and compile its driver. I discovered that this tuner is based on an Empiatech EM2880 chipset, whose support has been added to the latest development version of the Video for Linux Digital Video Broadcasting (v4l-dvb) Project. I downloaded the v4l-dvb development source code and compiled the neces-

Building Device Drivers for Pinnacle PCTV HD Pro Stick and Hauppauge WinTV-HVR-950 on Ubuntu 7.10

First, enable universe and multiverse package repositories by selecting System→Administration→Software Sources from the GNOME desktop menu. Click on the tab labeled Ubuntu Software, and make sure the boxes are checked for Community-maintained Open Source software (universe) and Software restricted by copyright or legal issues (multiverse). Click Close.

Next, apply all latest updates from Ubuntu by selecting System→Administration→Update Manager from the GNOME desktop menu, apply all system updates, and reboot the system.

Then, in a terminal, do `sudo su` to become the root user. Install the necessary packages to build em28xx kernel modules:

```
aptitude install mercurial build-essential linux-source
```

Download firmware version 4, necessary for USB tuner cards:

```
wget -q http://konstantin.filtschew.de/v4l-firmware/firmware_v4.tgz
➥-O /usr/local/src/firmware_v4.tgz
```

Unpack the firmware files into /lib/firmware:

```
tar xzf /usr/local/src/firmware_v4.tgz -C /lib/firmware
```

Grab the latest copy of the V4L DVB source code from **mcentral.de**:

```
cd /usr/local/src
hg clone http://mcentral.de/hg/~mrec/v4l-dvb-kernel
```

Compile the V4L DVB drivers:

```
cd /usr/local/src/v4l-dvb-kernel
make
make install
```

And, finally, reboot the system.

The pcHDTV HD-5500 is a PCI card and is the only hardware designed and marketed to support Linux right out of the box.

Building a Device Driver for Hauppauge WinTV-HVR-1800 on Ubuntu 7.10

First, enable universe and multiverse package repositories, by selecting System→Administration→Software Sources from the GNOME desktop menu. Click the tab labeled Ubuntu Software, and make sure the boxes are checked for Community-maintained Open Source software (universe) and Software restricted by copyright or legal issues (multiverse). Click Close.

Next, apply all latest updates from Ubuntu by selecting System→Administration→Update Manager from the GNOME desktop menu, and apply all system updates. Then, reboot the system.

Next, launch a terminal and do `sudo su` (to become the root user). Install the necessary packages to build the latest v4l-dvb drivers:

```
aptitude install mercurial build-essential linux-source
```

Grab the latest copy of V4L DVB source code from **linuxtv.org**:

```
cd /usr/local/src
hg clone http://linuxtv.org/hg/v4l-dvb
```

Compile the V4L DVB drivers:

```
cd /usr/local/src/v4l-dvb
make
make install
```

And, reboot the system.

sary modules for the running kernel (see the Building Device Drivers for Pinnacle PCTV HD Pro Stick and Hauppauge WinTV-HVR-950 on Ubuntu 7.10 sidebar). Compiling the kernel modules went through without a hitch, and after a system reboot, the Pinnacle PCTV HD Pro Stick was recognized by the system. Its digital (ATSC) tuner was registered as a DVB front end (/dev/dvb/adapter0), and its analog (NTSC) tuner was accessible as a v4l video device (/dev/video0).

After configuring the tuner as a "DVB DTV capture card (v3.x)" in MythTV, I was able to tune into standard-definition and high-definition programs. Picture quality for HD programs was fantastic, with astonishing crispness and clarity compared to SD programs. The viewing experience for this USB device was excellent.

If you're a couch potato, a working remote control in MythTV is absolutely essential.

I used tvtime to evaluate the analog TV performance. The picture quality was significantly better for stations with transmission towers that were geographically closer. Initially, sound did not work in tvtime. Using sox to route audio from tvtime to the default ALSA sound device solved the problem. Research on the Web indicated that many others have faced this same issue when using tvtime.

Hauppauge WinTV-HVR-950

The Hauppauge WinTV-HVR-950 is a USB 2.0 HDTV tuner card, which is very similar to the Pinnacle PCTV HD Pro Stick, except the Hauppauge device does not come with a remote control. Like the Pinnacle PCTV HD Pro Stick, it supports digital (ATSC), analog (NTSC) as well as unencrypted cable TV signals. This card comes with a portable telescopic antenna, USB extender cable and A/V adapter cable with inputs for S-Video, composite video and stereo audio. I found that the antenna provided with this card was not as good as the one supplied with the Pinnacle PCTV HD Pro Stick. In fact, I had to use a better indoor amplified antenna or an outdoor antenna instead. This card also blocked adjacent USB ports when plugged in to my system.

Hauppauge does not officially support Linux for this product. But, this tuner card also is based on the Empiatech EM2880 chipset (same as Pinnacle's PCTV HD Pro Stick). Hence the same kernel modules built for the HD Pro Stick worked with this tuner. Follow the steps in the Building Device Drivers for Pinnacle PCTV HD Pro Stick and Hauppauge WinTV-HVR-950 on Ubuntu 7.10 sidebar to build the driver.

The viewing experience for both SD and HD programs and analog TV was similar to that provided by the Pinnacle PCTV HD Pro Stick.

Hauppauge WinTV-HVR-1800

The Hauppauge WinTV-HVR-1800 is a PCI Express x1 tuner that also supports digital (ATSC), analog (NTSC) and unencrypted cable TV signals. This card has coaxial inputs for analog cable TV, digital ATSC/QAM TV and FM radio. It also has inputs for S-Video/composite and L/R stereo audio. An integrated hardware MPEG-2 encoder offloads the system processor when recording analog TV or cable channels. Accessories include a remote control, USB IR receiver and IR transmitter cable (to control a set-top unit).

Out of the box, this tuner is not supported by Linux. However, you can get the digital (ATSC) tuner to work reliably with Ubuntu 7.10 after you

Table 1. Tuners at a Glance

Tuner	Overall Rating (5 stars are best)	Out-of-the-box Linux support	Price	Interface	Vendor Web Site (Linux helpfulness—5 stars is best)	Accessories
pcHDTV HD-5500	★★★★★	Yes	$129	PCI	★★★★	A/V adapter cable
Hauppauge WinTV-HVR-1800	★★★★	No	$120	PCI Express x1	★★★	Remote control, USB IR receiver, A/V adapter cable
Pinnacle PCTV HD Pro Stick	★★★	No	$90	USB 2.0	★	Telescopic portable antenna, remote control, USB extender cable, A/V adapter cable
Hauppauge WinTV-HVR-950	★★	No	$70	USB 2.0	★★★	Telescopic portable antenna (lower quality than the one by provided the Pinnacle Pro Stick), USB extender cable, A/V adapter cable

build its driver (see the Building a Device Driver for Hauppauge WinTV-HVR-1800 on Ubuntu 7.10 sidebar). The analog TV features for this tuner could not be evaluated under Linux, because the driver does not yet support the analog circuitry.

After setting up the tuner in MythTV, I was able to view both HD and SD programs. There was no noticeable difference in picture quality compared to the USB tuners. The remote control worked well and is fully supported by LIRC (Linux Infra-Red Control) using the Windows Media Centre Remotes (new version Philips, et al.) configuration settings in LIRC. If you're a couch potato, a working remote control in MythTV is absolutely essential.

Digital TV on Linux Is Here Today

You can experience a great HDTV show on your desktop with the right equipment and some tweaking. If you're looking to build a full-fledged media center based on Linux, MythTV does it all—from program listings, Picture-in-Picture, remote control support, Web administration with MythWeb, programmable recording, to watching your favorite shows. If you're a minimalist

Interview with Jack Kelliher, CEO and Cofounder of pcHDTV

AS: How did you become interested in building HDTV cards for Linux? Was it because the technology is open source, or was it a personal interest?

JK: Actually, both. I had an early HD card for Windows but almost exclusively used Linux and wanted one for Linux. As there were open-source MPEG players available and a niche market for Linux, I felt that it was a very doable project that could grow into a small business.

AS: What have been your challenges in making the pcHDTV products successful?

JK: Of course, the first challenge was developing the card, drivers and modifying a player to handle HD playback, followed by problems in production. We thought we had a fairly small window of opportunity, as the FCC was planing to enforce the broadcast flag [a set of status bits sent in the data stream of a digital television program that indicates whether the data stream can be recorded, or if there are any restrictions on recorded content], which would not have been very compatible with Linux. Luckily, the Supreme Court struck this down, although Congress has considered it a couple times since.

AS: What do you find exciting about Linux after many years of working with the technology?

JK: The extraordinary advances in open-source software, like MythTV, and very usable video viewing, editing and animation applications—even medical applications, like MRI viewers.

AS: How do you see your products evolving?

JK: We want to support PCI Express in the future, and we are considering a small USB product as well.

AS: What are your thoughts about targeting your product for European users (non-ATSC users)?

JK: We have been and are continuing to work on a world-wide solution, and this has been using up most of our time. The project is fairly large for us, but we hope to introduce something by the end of the year.

and don't want to dedicate an entire system, Me-TV looks promising.

It goes without saying that you should do your homework before buying components for your DTV Linux system. Visit popular on-line forums, such as the MythTV Wiki, LinuxTV Wiki, Ubuntu Forums and Ubuntu Wiki to tap into the wisdom of the crowd.

In summary, my recommendations for a great DTV experience include outdoor antenna, fast multicore processor, medium to high-end video card with at least 256MB video memory and a fast high-capacity hard drive. For your desktop, the pcHDTV HD-5500 works right out of the box. The Hauppauge WinTV-HVR-1800 is a close second. For your laptop, the Pinnacle PCTV HD Pro Stick and the Hauppauge WinTV-HVR-950 USB tuners work well.■

Alolita Sharma has been involved with open source since the early days of Linux. As a software engineer and industry consultant, she promotes disruption through open source. She is cofounder and CEO of Technetra and OSI Board Member. She can be contacted at alolita.sharma@gmail.com.

Resources

pcHDTV HD-5500: **pchdtv.com/hd_5500.html**

Pinnacle PCTV HD Pro Stick: **www.pinnaclesys.com/PublicSite/us/Products/Consumer+Products/PCTV+Tuners/PCTV+Analog_Digital+PVR/PCTV+HD+Pro+Stick.htm**

Hauppauge WinTV-HVR-950: **hauppauge.com/site/products/data_hvr950.html**

Hauppauge WinTV-HVR-1800: **hauppauge.com/site/products/data_hvr1800.html**

MythTV: **mythtv.org**

LinuxTV: **linuxtv.org**

tvtime: **tvtime.sourceforge.net**

LIRC: **www.lirc.org**

Xine: **xinehq.de**

Me-TV: **https://launchpad.net/me-tv**

Continuous Data Protection
The Future of Data Centers

Can your backup software do this?	R1Soft CDP Server	Acronis® True Image	EMC Retrospect ®
Daily Backups	✔	✔	✔
Hourly Backups	✔	Not Supported	Not Supported
Open File Backups	✔	✔	Not Supported
Bare-Metal Restore	✔	✔	Not Supported
Continuous Data Protection	✔	Not Supported	Not Supported
Restore Linux LVM	✔	Not Supported	Not Supported
Restore Linux Software RAID	✔	Not Supported	Not Supported
Easy To Use Web Interface	✔	Not Supported	Not Supported
Manage Thousands of Servers	✔	Not Supported	Not Supported
Control Panel Integration	✔	Not Supported	Not Supported
	$99 /server †	$699 /server	You Can't Afford It

(NEW) - CDP for MySQL add-on Now available | True-Granular Restore™ | Store over 50 recovery points per-day
Bare-Metal Restore for MySQL Servers | Restore tables or databases to original or alternate locations and more..

Data Centers serious about uptime and performance use R1Soft.

For more information visit: **www.r1soft.com** or call us at **800-956-6198**

†Price includes $600 Data Protection Server cost. Assumes minimum ratio of 25 protected servers per Data Protection Server

R1Soft
Continuous Data Protection®
For LINUX & WINDOWS

Copyright 2007 Righteous Software Inc All Rights Reserved.
R1Soft is a trademark of Righteous Software Inc. Other names may be trademarks of their respective owners.

KEEP ON BLOGGING IN A FREE WORLD

WordPress founder and open-source crusader Matt Mullenweg talks to us about his evolving and ever-popular blogging platform.

KATHERINE DRUCKMAN

PHOTO BY RICHARD WHEELER

Unless you've been living under a rock for a few years, you most certainly have heard of WordPress, one of the most popular blogging platforms around that also happens to be 100% open source. There are a few iterations of WordPress that power personal and corporate blogs as well as collections of blogs for large institutions, like Harvard University. And, WordPress.com brings blogging to anyone with just a few clicks.

WordPress has a tremendous community both in terms of size and enthusiasm, and many people make their living implementing and using it. Since its first release in 2003, the community has grown with the application, and WordPress' commitment to open source is as strong as ever. I have a WordPress blog, and a lot of my friends have WordPress blogs, so I was thrilled to have the opportunity to ask founding developer, Matt Mullenweg, some questions about his project.

KD: For those (few) not familiar with WordPress, what features or qualities set WordPress apart from other blogging platforms and content management systems?
MM: WordPress aspires to be invisible. If we do our job, you completely forget it's there and just focus on what matters to you and your users—your writing.

WordPress was created as a blogging tool, but more and more savvy developers are discovering it shines as a general CMS as well, and its user-friendly interface works well with people of any experience level.

KD: I believe it is safe to say that since WordPress' inception, blogging as a medium has evolved by leaps and bounds. In what ways has this evolution met or exceeded your expectations? Any surprises?
MM: The biggest surprise to me is that when WordPress started, the common assumption was that blogging had peaked or would soon, and that the software market for it was already saturated. That turned out not to be the case.

I also didn't expect the embrace of rich media that has transformed blogging over the past several years. I mean, we called it WordPress, not PhotoPress or VideoPress. The written word is still the heart of what we do, but people's imaginations have been captured by podcasting, videocasting and photoblogs.

KD: What advice do you have for those of us trying to scale our WordPress sites? Are there any other open-source tools you would suggest in this arena?
MM: Ninety percent of scaling happens before a person ever reaches WordPress. You need a server that's configured to serve static files efficiently, perhaps with a reverse proxy. You need a database tuned to handle the size of your dataset well. (Most blogs are only a few megs of data.) Finally, where core WordPress is very scalable (we served more than 140 million uniques using it last month on WordPress.com), there are some plugins that can slow your site down, so be watchful of performance after turning on a new plugin.

KD: Speaking of gargantuan sites, there are some pretty popular sites out there using WordPress—**icanhascheezburger.com** comes to mind. Are there any other popular or otherwise noteworthy sites that have impressed you with their implementation?
MM: Cheez is one of my favs. I subscribe to more than 300 blogs, so it's tough to name favorites.

Implementation-wise, I've been impressed with:

- **allthingsd.com**, from the *Wall Street Journal*.
- **particletree.com**, a Web development magazine.
- **autoshows.ford.com**, which talks about Ford's concept cars.
- **www.andyroddick.com**, the tennis player.
- **71miles.com**, a travel guide and directory.
- **www.thinkvitamin.com**, another great Web magazine.
- **www.futureofthebook.org/gamertheory**, an avant-garde interactive book.

Content-wise, I enjoy:

- **gigaom.com**
- **politicalticker.blogs.cnn.com**
- **stuffwhitepeoplelike.wordpress.com**
- **zeldman.com**
- **www.svextra.com/blogs/gmsv**
- **diveintomark.org**

And so many more! But I'll stop now.

KD: There has been some controversy recently, which erupted from some dialogue with Six Apart regarding competition from Movable Type, another blogging platform that has multiple licenses. You responded by being something of a crusader for open source. Is that a fair assessment? How do you feel your team's contributions to open source affect your place in the emerging blogosphere?
MM: I consider myself a strong proponent of open source. I would like to think the fierce competition and success WordPress has shown in the market was a factor in Six Apart's effort to remain relevant and put Movable Type under an open-source license.

I think before, when we were open source and they were proprietary, people sometimes chose WordPress because of its license and freedom, but growth hasn't slowed since they switched, so now I suppose people are more influenced by functionality and our broad community in their decision to use WordPress over other software.

KD: You frequently have reiterated your commitment to open-source ideals and GPL licensing. How has this commitment factored into the development of your company, Automattic? How do you use open-source technology to achieve your goals?
MM: When I set out to create Automattic, it was an interesting dilemma—in our society, it seemed the best way to have an impact on the world was working within a for-profit framework, but at the same time, I'd seen multiple examples of "open-source companies" suffocating the communities they grew from.

I came across an interesting hack though—by keeping WordPress.org a separate entity from Automattic and basing our business entirely on GPL code, you create a balance that aligns the fiduciary responsibilities of the corporation with the interests of the community at large. In the long term—10, 20 years from now—it still will be in the best interest of Automattic to support the broader community as much as possible, because its own business succeeds when they do.

I didn't want WordPress to be a one-company project, so by separating out the nonprofit and for-profit sides and making some explicit decisions about businesses Automattic would never enter, we created a lot of room for other companies to embrace, support and build on top of WordPress. Hopefully, we also set a good example of how to contribute back to the community.

It was the best way I could think of to ensure that the principles I believe in would endure beyond my personal involvement or control of either organization. (But I still look both ways when crossing the street.)

KD: I noticed WordPress is licensed under GPL v.2. What are your thoughts on GPL v.3?
MM: I haven't researched it enough to have a strong opinion yet, but I am generally supportive of the efforts of the Free Software Foundation and donate to them regularly.

KD: Spam plagues us all in the world that is Web, and your Akismet Project has been a very popular weapon. Akismet is closed source, but has an

open-source plugin, and there has been criticism for that. How did you arrive at your current approach with Akismet in particular?

MM: That was a tough one. Basically, what it came down to is I had been creating antispam plugins for a long time, and every iteration would work for a shorter and shorter period of time until it was literally a matter of hours before spammers would download my plugin, see what it did and circumvent it.

Akismet was created to break this cycle, to provide a long-term solution to spam, and the best way I could see to do that was a centralized service that could adapt to spammers' tactics as fast as they were changing them. At the time, the decision was weighing the good to the world of the Akismet algorithms and code being open source vs. the good to the world of solving people's spam problems.

So, we made the decision at the time to err on the side of stopping spammers, and the community was very supportive. It's entirely within the realm of possibilities that the more generally useful parts of it will be open-sourced in the future.

KD: There was an announcement in February about a security issue that needed immediate attention—how do you and your team address the security of WordPress? How do you balance the desire to add features with the security risks related to change?

MM: Well, security is always a priority over new development, for obvious reasons. It's not about an audit or single event though. It's a mindset that has to pervade everything you do. I'm a very trusting guy, so early on it was difficult for me to think about how bad actors could exploit a system—for example, when I co-created the open ping service Ping-O-Matic, which is the update ping equivalent of an open relay. It still runs today, but it's attacked by spammers constantly.

I think the most important thing with regards to security is that you're transparent and responsive. When a legitimate problem comes in, we'll get a fix into the hands of users as fast as we can. As WordPress has grown in popularity, there have been many eyes on it, and over time, the nature of new vulnerabilities has become more benign. Until DJB writes blog software, I think we'll be one of the best out there with regard to security, not because we have a perfect history, but because we've learned from many mistakes. When you dig in to WordPress, you'll find a lot of security foundations with nonces, header-splitting protection, HTML sanitation, encrypted cookies, salted passwords and so on.

KD: WordPress is going on five years as an application. Are there any technical decisions you made in the beginning that you regret today? Are there dark corners you would love to clean up but never get around to it?

MM: Absolutely! Tons of stuff. But I've seen the mistake of starting from scratch one too many times. Some projects survive it, like Mozilla/Phoenix/Firefox emerged from the ashes, but more often than not, the engineer-led ground-up rewrite is a good time to call the peak of a product.

Our approach is more iterative. If you compare 1.0 to 2.5, they are like night and day, but the transformation happened bit by bit, release by release. We maintain as much backward-compatibility in the process as we can.

KD: How is development on WordPress organized? Who decides when it is time to call it 2.5?

MM: It's very ad hoc. My role is BDFL, there are four committers, and then dozens and dozens of developers who contribute patches large and small. Shipping is the hardest part. It's so easy to fall into an extended development cycle where you just endlessly noodle and perfect every little thing.

That's actually what happened between 2.0 and 2.1. Now we have more of an Ubuntu approach to releases, where they're more time-based. The discipline has been good for the project. The releases are just as stable as before, we just get cool new features into the hands of users three times a year instead of only once.

Of course, there's nothing quite like working on a Web service like WordPress.com or Akismet. It's so nice to be able to deploy updates 20 times a day to a completely homogeneous environment where you control all the variables. It spoils you.

KD: WordPress supports plugins—some are minor, and others bend WordPress in new directions. Any favorites, or ideas for a plugin you wish you could download right now?

MM: It's cheating a little bit, but things I like as plugins often end up getting built in to the product. I think the main two I have on my site are Akismet and WordPress.com Stats, both of which are from Automattic.

In terms of what I want, I'd love to see something that allowed blog readers to suggest tags or categories, and then those could go into a queue for moderation by the blog author. I'd also love to be able to point the uploader to a local directory on the server or a URL and let it slurp up the images from there, much like Gallery does it.

I bet both of those exist already, with so many thousands of plugins, sometimes the hardest thing is just finding what's already out there.

KD: You just released 2.5 in March—do you already have ideas about what you'll be working toward in 2.6 or 3.0?

MM: The best ideas come right before a release, because you're in "ship" mode, and you have a thousand great ideas that if you just could slip this one great thing in...but you know you can't, because then you'd have to start the testing cycle over again. I have a list (15–20 things long) of features and improvements I'd love to see happen in 2.6, and I'm sure the other developers do as well. We also have all the great ideas that the community proposes and votes for in our Ideas forum. Pretty soon, we'll do an IRC meetup and hash out a rough outline and get cracking, and thus, the cycle begins again.■

Katherine Druckman is an HTML-flinging, PHP-hacking *LJ* Webmistress by day, and a refined connoisseur of classic architecture and fine Chinese ceramics by night. She usually can be found surrounded by the charm of aging Texas buildings from the pioneer days or appreciating ceramics of the Song and Qing dynasties. Well, either that or sitting in a comfy chair with a laptop. Yeah, probably the laptop thing.

WordPress is the Content Manager System winner in the 2008 *LJ* Readers' Choice Awards. See our June issue for more details.

USING WEBKIT IN YOUR DESKTOP APPLICATION

Include Web content in your desktop application with WebKit.

BENJAMIN MEYER

Qt always has included the ability to render basic HTML and download files with HTTP. With the release of version 4.4.0, this has been taken to a whole new level with the inclusion of WebKit. Developers who use Qt now can utilize WebKit for everything—from simple HTML document viewers to full-blown Web browsers. Trolltech always has been known for creating high-quality APIs that are easy and intuitive to use, and this is just as true with QtWebKit, the integration of WebKit with Qt.

The WebKit rendering engine is used by Safari and has its roots in the KDE Project's KHTML engine, which drives the Konqueror Web browser. Licensed under the LGPL, this open-source engine has been praised for its performance and low memory usage. It was the ideal choice for small devices, such as the Nokia S60 and the iPhone. Beyond Web browsers, WebKit is used by many applications, including Adium, Colloquy, MSN Messenger and Mac OS X's Dashboard. With the addition of the Qt port to WebKit, there no doubt will be many more cross-platform applications in the near future that take advantage of this engine.

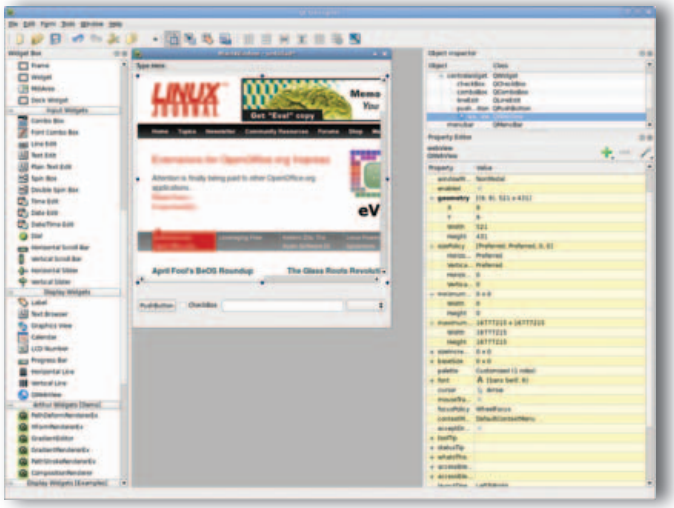


Figure 1. Designer lets you easily create forms with widgets, including WebKit.

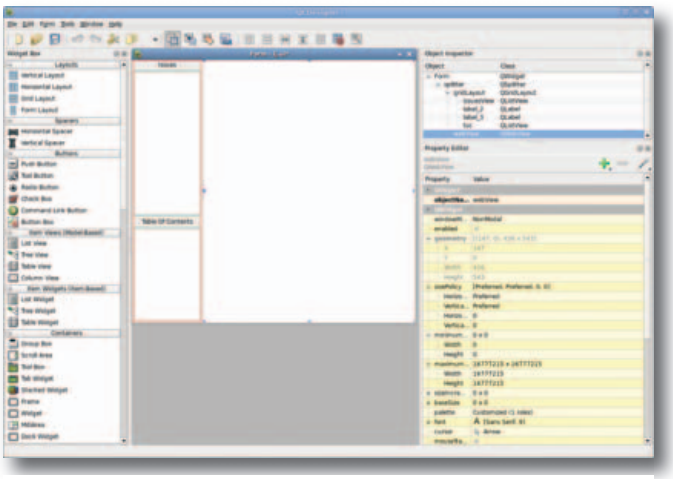


Figure 2. Designer shows off the form for our application.

QtWebKit provides developers with a handful of useful classes. At the very top level, there is QWebView, which is a QWidget with a number of convenience functions, such as setUrl(), loadProgress() and reload(). Inside Qt Designer, the GUI builder for Qt applications, you even can drag a QWebView into a form and set the URL. QWebView is built on top of QWebPage, which contains the Web content, history and settings. QWebPage is not a widget, but was built to be used on many surfaces, including QGraphicsView, Qt's canvas widget. Supporting QWebView and QWebPage are classes that let you build plugins, access the page history and walk the frames.

A lot of the fun of having WebKit in your application is having it pull content from the Internet. Qt 4.4.0 introduces new networking classes, including an all new HTTP implementation. QNetworkAccessManager handles all network requests and replies with support for HTTP 1.0, 1.1 and SSL. A custom cookie jar and proxy configuration also can be set. Qt's source code includes a demo browser and example applications that show off how to use many of the features of these classes.

Qt always has provided fantastic cross-platform support with integration into the desktop. With the introduction of QtWebKit, developers now can make a cross-platform desktop application for a Web site. *Linux Journal* provides a digital subscription that lets you download older issues. The Web site is very simple and a perfect candidate for building a small application around. Although the Web site does have the table of contents, it does not provide a way to search all the available issues for articles. So the application I am going to make provides an easy way to search through the issues and let you download them.

The GUI for the application was made with Qt Designer and has a matching main window class that contains the functionality. To compile the project, Qt includes a cross-platform build tool called qmake. Beyond the normal qmake template when using QtWebKit, the Qt variable also needs WebKit to be specified for the library to be linked in. Our application's project file (lj.pro) consists of the following:

```
TEMPLATE = app
QT += WebKit
FORMS += lj.ui
SOURCES += main.cpp mainwindow.cpp
HEADERS += mainwindow.h
```

Like most Qt applications, main.cpp contains only a small amount of code. It constructs a QApplication and the main window, and then starts the event loop. By setting the application name, we tell QtWebKit to include it in the user agent string automatically. That way, if Qt's networking in your application starts behaving badly, Web site developers know whom to contact. The user agent string is, of course, fully customizable by subclassing QWebPage if you need to. Here's the main.cpp file:

```
#include <qapplication.h>
#include "mainwindow.h"

int main(int argc, char **argv)
{
    QApplication app(argc, argv);
    app.setApplicationName("LinuxJournalDigital");
    MainWindow mainwindow;
    mainwindow.show();
    return app.exec();
}
```

The interface was built in just a few minutes using Qt Designer. On the left-hand side are two QListViews. The one on the top will contain the list of available issues, and the bottom one will contain the table of contents of the currently selected issue. On the right-hand side is a QWebView.

The interface file is turned into a header file (ui_lj.h) by uic during the compilation process. ui_lj.h contains the generated Ui_Form class along with all the objects in the interface. The main window class definition is a subclass of QMainWindow and Ui_Form. The only new objects in the MainWindow classes are the models that are used to contain the list of issues

A lot of the fun of having WebKit in your application is having it pull content from the Internet.

and the proxy, which is used for searching. The mainwindow.h file is as follows:

```
#include <QtGui/QtGui>
#include <QtWebKit/QtWebKit>
#include "ui_lj.h"

class MainWindow :
    public QMainWindow, public Ui_Form
{
    Q_OBJECT
public:
    MainWindow();

private slots:
    void downloadFinished();
    void clicked(const QModelIndex &);
    void activated(const QModelIndex &);
    void downloadRequested(const QNetworkRequest &);
    void downloadingIssueFinished();
    void downloadProgress(qint64, qint64);

private:
    QStandardItemModel *issues;
    QSortFilterProxyModel *proxy;
    QStringListModel *tocModel;
};
```

mainwindow.cpp contains all the application's plumbing. The MainWindow constructor sets up the interface, creates the toolbar and begins to fetch the available issues. setupUi() is declared in the generated interface header and populates the central widget with the widgets that were specified in the interface file. The toolbar is populated with actions for the Web page and a line edit for searching. Rather than create and set up each QAction manually, QWebPage has built-in actions that can be used. Here's mainwindow.cpp:

```
#include "mainwindow.h"

#define SERVER "https://secure.linuxjournal.com/" \
                "allsubs/"

MainWindow::MainWindow() : QMainWindow()
{
  QWidget *w = new QWidget;
  setCentralWidget(w);
  setupUi(centralWidget());

  connect(issuesView, SIGNAL(activated(QModelIndex))
         ,this, SLOT(activated(QModelIndex)));
  connect(issuesView, SIGNAL(clicked(QModelIndex)),
          this, SLOT(clicked(QModelIndex)));
  issues = new QStandardItemModel(issuesView);
  proxy = new QSortFilterProxyModel(issues);
  proxy->setSourceModel(issues);
  proxy->setFilterCaseSensitivity(Qt::CaseInsensitive);
  proxy->setFilterRole(Qt::UserRole + 1);
  issuesView->setModel(proxy);
  connect(
    webView, SIGNAL(statusBarMessage(QString)),
    statusBar(), SLOT(showMessage(QString)));
  connect(webView->page(),
    SIGNAL(downloadRequested(QNetworkRequest)),
    this, SLOT(downloadRequested(QNetworkRequest)));
  tocModel = new QStringListModel(this);
  toc->setModel(tocModel);

  QToolBar *bar = addToolBar(tr("Navigation"));
  bar->addAction(
    webView->pageAction(QWebPage::Back));
  bar->addAction(
    webView->pageAction(QWebPage::Forward));
  bar->addAction(
    webView->pageAction(QWebPage::Stop));
  bar->addAction(
    webView->pageAction(QWebPage::Reload));
  bar->addSeparator();

  QLabel *label = new QLabel("Search:", bar);
  bar->addWidget(label);
  QLineEdit *search = new QLineEdit(bar);
  QSizePolicy policy = search->sizePolicy();
  search->setSizePolicy(QSizePolicy::Preferred,
                        policy.verticalPolicy());
  bar->addWidget(search);
  connect(search, SIGNAL(textChanged(QString)),
    proxy, SLOT(setFilterFixedString(QString)));
  QUrl home(SERVER "dlj.php?action=show-account");
  webView->load(home);

  setWindowTitle("Linux Journal Digital Archive");

  QNetworkAccessManager *networkManager =
    webView->page()->networkAccessManager();

  QUrl url(SERVER "dlj.php?action=show-downloads");
  QNetworkRequest request(url);
  QNetworkReply *r = networkManager->get(request);
  connect(r, SIGNAL(finished()),
          this, SLOT(downloadFinished()));
}
```

When the application launches, the user will see the main login page, and in the background, the "show-downloads" page is downloaded from *Linux Journal*. In an ideal world, *Linux Journal* would provide a simple XML file with all the available issues, table of contents and download location, but because this is just a demo, this information is acquired the hard way. It does this by using a regular

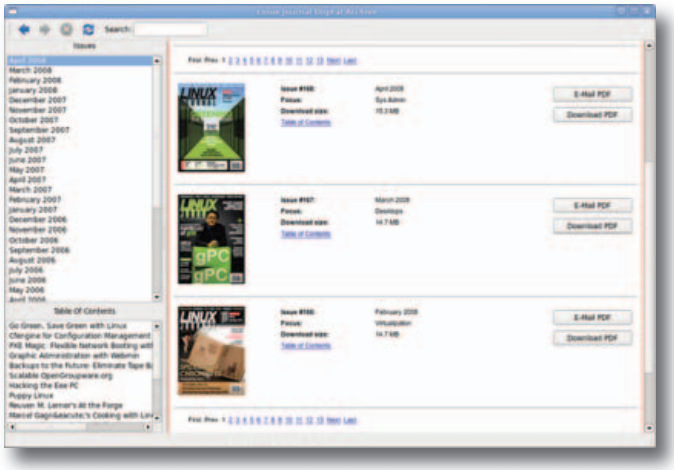


Figure 3. The application with all the found issues and the table of contents of the currently selected issue.

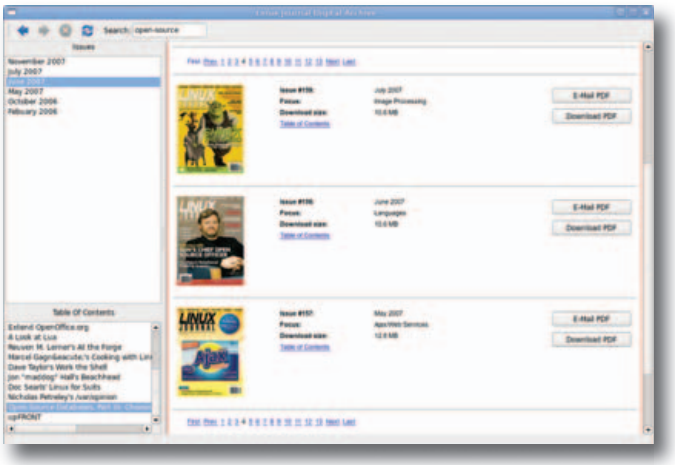


Figure 4. On-the-fly searching filters the issues only to those that contain the search string.

expression to find any available issues, which is listed at the top of every Web page:

```
void MainWindow::downloadFinished()
{
  QNetworkReply *reply =
      ((QNetworkReply *)sender());
  QByteArray data = reply->readAll();
  QTextStream out(&data);
  QString file = out.readAll();

  // The first page, find all of the pages that
  // we can download issues from and fetch them.
  if (issues->rowCount() == 0) {
    QRegExp rx("show-downloads&row_offset=[0-9]*");
    QStringList pages;
    int pos = 0;
    while (pos != -1) {
      pos = rx.indexIn(file, pos + 1);
      QString page = rx.capturedTexts().first();
      if (!page.isEmpty() && !pages.contains(page))
        pages.append(page);
    }
    QNetworkAccessManager *networkManager =
      webView->page()->networkAccessManager();
    foreach (QString page, pages) {
      QUrl url(SERVER "dlj.php?action=" + page);
      QNetworkReply *reply =
          networkManager->get(QNetworkRequest(url));
      connect(reply, SIGNAL(finished()),
              this, SLOT(downloadFinished()));
    }
  }
```

Each Web page also contains several issues, usually three. Another regular expression is used to find each issue and the table of contents for that issue. After they are extracted, the data is put into the model where it is displayed:

```
QRegExp issue("class=\"data-data\">([a-zA-Z]* " \
              "20[0-9][0-9])</td>");
  int pos = 0;
  while (pos != -1) {
    pos = issue.indexIn(file, pos + 1);
    QString page = issue.capturedTexts().value(1);
    QStandardItem *item = new QStandardItem(page);
    if (!page.isEmpty()) {
      item->setData(reply->url(), Qt::UserRole);
      item->setFlags(Qt::ItemIsSelectable
                     | Qt::ItemIsEnabled);
      issues->insertRow(issues->rowCount(), item);
    }

    // Now that we have an issue, find the
    // table of contents
    QRegExp toc("<div id=\"toc-div-[0-9]{1,4}\".*" \
                "</div>");
    toc.setMinimal(true);
    toc.indexIn(file, pos);
    QStringList list =
       toc.capturedTexts().first().split("<li>");
    for (int j = list.count() - 1; j >= 0; --j) {
      QString s = list[j].simplified();
      if (!s.endsWith("</li>"))
        list.removeAt(j);
      else {
        s = s.mid(0, s.length() - 5);
        list[j] = s;
      }
    }

    // The table of contents is joined
    // together in one string and is used
    // by the proxy for searching
    item->setData(list.join(" "),
                  Qt::UserRole + 1);

    // Save TOC which will be used to populate the
    // TOC list view if this issue is clicked on
    item->setData(list, Qt::UserRole + 2);
  }
}
```

Qt always has provided fantastic cross-platform support with integration into the desktop.

The proxy is set to filter on `Qt::UserRole + 1`, which contains the full table of contents for each issue. When you type in the search box, any issue that doesn't contain the string will be filtered out.

When an issue is clicked, the table of contents is fetched out of the issuesView model and inserted into the tocModel where it is displayed in the lower list view:

```
void MainWindow::clicked(const QModelIndex &index)
{
  QVariant v = index.data(Qt::UserRole + 2);
  tocModel->setStringList(v.toStringList());
}
```

When an issue is activated (depending on the platform, this could be a double-click or single-click), the URL is fetched out of the issue model and set on the QWebView:

```
void MainWindow::activated(const QModelIndex &index)
{
  webView->load(index.data(Qt::UserRole).toUrl());
}
```

Once the user clicks on the download issue button, the Web site confirms authentication and then forwards to a Web page to download the actual file. Once there, downloadRequested() is called. From here on out, the example deals mostly with the new networking code. QWebPage has a built-in QNetworkAccessManager that is used to fetch the PDF:

```
void MainWindow::downloadRequested(
        const QNetworkRequest &request)
{
  // First prompted with a file dialog to make sure
  // they want the file and to select a download
  // location and name.
  QString defaultFileName =
    QFileInfo(request.url().toString()).fileName();
  QString fileName =
    QFileDialog::getSaveFileName(this,
                                 tr("Save File"),
                                 defaultFileName);
  if (fileName.isEmpty())
    return;

  // Construct a new request that stores the
  // file name that should be used when the
  // download is complete
  QNetworkRequest newRequest = request;
  newRequest.setAttribute(QNetworkRequest::User,
                          fileName);

  // Ask the network manager to download
  // the file and connect to the progress
  // and finished signals.
  QNetworkAccessManager *networkManager =
    webView->page()->networkAccessManager();
  QNetworkReply *reply =
    networkManager->get(newRequest);
  connect(
    reply, SIGNAL(downloadProgress(qint64, qint64)),
    this, SLOT(downloadProgress(qint64, qint64)));
  connect(reply, SIGNAL(finished()),
          this, SLOT(downloadIssueFinished()));
}
```

Because *Linux Journal* PDFs are large files, it is important to give notification on the download progress. The simplest method is to update the status bar with the progress:

```
void MainWindow::downloadProgress(qint64
            bytesReceived, qint64 bytesTotal)
{
  statusBar()->showMessage(QString("%1/%2")
           .arg(bytesReceived)
           .arg(bytesTotal), 1000);
}
```

When the PDF has finished downloading successfully, the filename and location that were chosen by the user before are retrieved, and the full file is saved to disk:

```
void MainWindow::downloadingIssueFinished()
{
  QNetworkReply *reply = ((QNetworkReply*)sender());
  QNetworkRequest request = reply->request();
  QVariant v =
    request.attribute(QNetworkRequest::User);
  QString fileName = v.toString();
  QFile file(fileName);
  if (file.open(QFile::ReadWrite))
    file.write(reply->readAll());
}
```

Conclusion

The inclusion of WebKit into Qt provides the opportunity for a number of very interesting applications to be developed. It will no doubt be utilized in many different types of applications, from desktop Web applications to applications that use it to display and manipulate HTML. QtWebKit has come a long way since the port was initially started two years ago. It will be exciting to see how QtWebKit improves and what people create with it.■

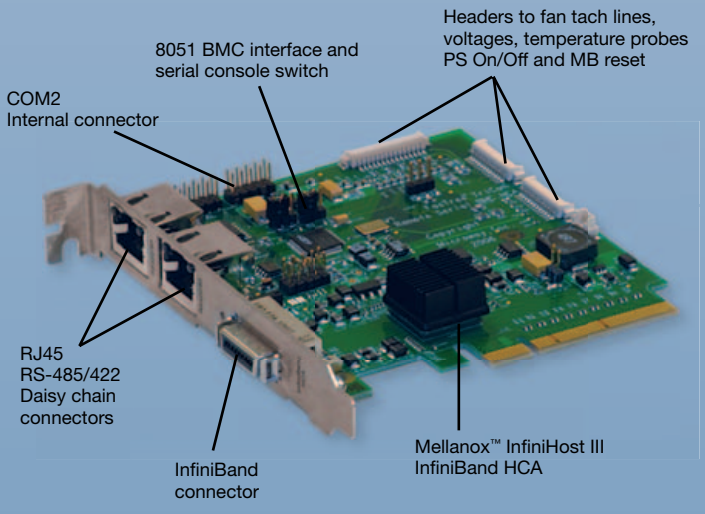
Benjamin Meyer is a happily married hacker. He has been developing with Qt for 11 years, and he has worked on many tools and applications, such as KDE's audiocd ioslave, System Settings, KAutoConfig, KAudiocreator, QAutotestGenerator, Valgrind Tools, NetflixRecommenderFramework, Zaurus applications and much more. He collects Transformers and runs the site toybin.org.

Affordable InfiniBand Solutions

4 Great Reasons to Call Microway NOW!

1 TriCom™

- DDR/SDR InfiniBand HCA
- "Switchless" serial console
- NodeWatch web enabled remote monitor and control



2 FasTree™

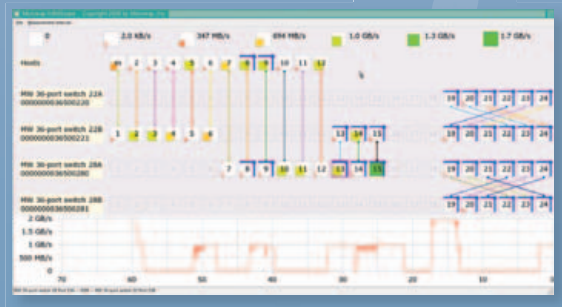
- DDR InfiniBand switches
- Low latency, modular design
- 24, 36 and 48 port building blocks

3 ServaStor™

- Extensible IB based storage building blocks
- Redundant and scalable
- Parallel file systems
- Open source software
- On-line capacity expansion
- RAID 0,1,1E, 3, 5, 6, 10, 50

4 InfiniScope™

- Monitors ports on HCA's and switches
- Provides real time BW diagnostics
- Finds switch and cable faults
- Lane 15 interface
- Logs all IB errors



Upgrade your current cluster, or let us design your next one using Microway InfiniBand Solutions.

*To speak to an HPC expert call **508 746-7341** and ask for technical sales or email **sales@microway.com***
www.microway.com

Microway®
Technology you can count on℠

DOJO
THE JAVASCRIPT TOOLKIT WITH INDUSTRIAL-STRENGTH MOJO

Featuring a rich standard library, an extensive collection of turn-key widgets, a unit-testing framework, and build tools for minifying your source files, it's no wonder that Dojo is a key part of products from industry giants, such as AOL, Sun Microsystems, BEA and others.

MATTHEW RUSSELL

A number of JavaScript toolkits have emerged to automate common Web development tasks and simplify creating rich user interfaces. Of all the contenders, Dojo stands out as the industrial-strength JavaScript toolkit because of its incredible depth and breadth. It features an extensive JavaScript library, a system of rich turn-key widgets, a collection of specialized subprojects, build tools and a unit-testing harness. Regardless of what your project entails, it is almost a certainty that Dojo can simplify the development and maintenance required. This article systematically focuses almost exclusively on some of the most fundamental constructs in the toolkit's highly optimized kernel, commonly referred to as Base.

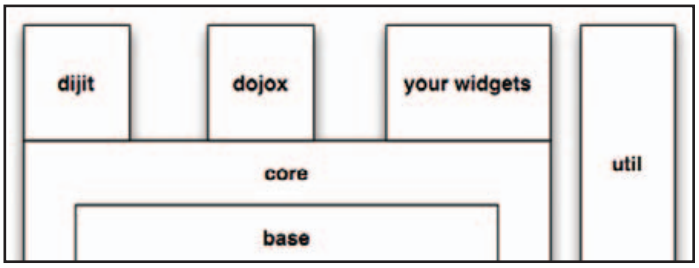


Figure 1. Toolkit Architecture

DOJO: THE JAVASCRIPT TOOLKIT YOU'VE ALWAYS WANTED

Variations among Web browsers have made developing applications for the Web really messy work. Working around subtle variations in JavaScript implementations, wrangling the Document Object Model (DOM) and normalizing content rendering across browsers can be downright tormenting at times, and unfortunately, a nontrivial portion of the investment in developing a solid Web application is spent re-inventing this kind of brittle boilerplate. Although many technologies have evolved to mitigate these kinds of issues, one you especially should be aware of the next time you decide to build a Web application is Dojo, the industrial-strength JavaScript toolkit.

In short, the Dojo toolkit is a liberally licensed client-side technology that can supplement virtually any aspect of Web development. It features a tiny but fully featured JavaScript standard library that insulates you from the bare metal of the browser, a large subsystem of widgets that snap into a page with little to no JavaScript required, and a suite of build tools for minifying and consolidating resources as well as writing unit tests. Knowing that industry giants, such as AOL, IBM, BEA and Sun Microsystems, are on board with Dojo should give you a boost of confidence if you're leery of trying something else in an ecosystem that's littered with half-baked inventions that don't always come though on their promises to deliver all-encompassing solutions.

This remainder of this article works through the bulk of the toolkit's most fundamental JavaScript programming constructs that will benefit you regardless of the size or scope of your project.

FIREBUG

The Firebug add-on for Mozilla Firefox is a terrific tool that can benefit your Web development efforts in ways you can't even imagine. Of particular interest is its console, which allows you to execute arbitrary JavaScript code—a great advantage when learning a new technology, as you'll commonly want to test out new ideas in an interactive fashion.

GETTING DOJO TO WORK IN LESS THAN ONE MINUTE

Although you could download Dojo from its official Web presence and set up a local installation, the easiest way to get started with Dojo is to use the latest version of Dojo that is hosted on AOL's Content Delivery Network (CDN). The following page skeleton demonstrates that the minimal effort required to put Dojo to work from the CDN is a SCRIPT tag that loads Dojo into the HEAD of the page; it is especially noteworthy that the SCRIPT tag incurs the cost of one request to the Web server that delivers a gzipped payload of approximately 29kB and provides the latest 1.1.x release that is available. In Dojo parlance, the good that the SCRIPT tag provides is called Base, because it provides the base for the toolkit, and because everything you'll use is contained in the base-level dojo.* namespace. Speaking of which, with the exception of the dojo identifier itself, the global page-level namespace is otherwise preserved:

```
<html>
    <head>
        <title>Putting Dojo To Work</title>

        <!-- Loading Dojo requires only one SCRIPT tag -->
        <script type="text/javascript"
            src="http://o.aolcdn.com/dojo/1.1/dojo/dojo.xd.js">
        </script>

        <script type="text/javascript">
            dojo.addOnLoad(function() {
                /* safely use any code that relies on dojo.*
                   functions in here ... */
            });
        </script>
    </head>
    <body>
        <a href="http://dojotoolkit.org">Dojo</a>
    </body>
</html>
```

To summarize, the dojo.addOnLoad block fires once the asynchronous loading of the dojo.xd.js file and any dependencies specified via dojo.require statements (more on these in a bit) have completed, and this is necessary in order to prevent any race conditions that might occur without it. Basically, the dojo.xd.js file providing Base accomplishes feats such as normalizing DOM events and provides you with a number of useful utilities for accelerating application development.

FETCHING NODES

As a natural starting point for our discussion, consider the following snippet from a Web page:

```
<form name="foo" action="/bar">
  <label>A form with name="foo"</label>
</form>

<div id="foo">
  A div with id=foo
</div>
```

The excerpt is trivial, and it should be obvious that running a function as simple as `document.getElementById("foo")` would *always* return the DIV element as a result. As a Linux user, you even could use a Gecko-based browser, such as Firefox, or a KHTML-based browser, such as Konqueror, to test the page and verify it for yourself. However, you may be alarmed and shocked to learn that running the very same test in Internet Explorer versions 6 or 7 returns the FORM element instead of the DIV element! This particular bug arises, because the name and id attribute namespaces are merged for IE. As it turns out, the DIV would have been returned if the FORM had not appeared first in the document, so this bug is especially tricky. At any rate, Dojo provides the dojo.byId function that works just like document.getElementById—except that it accounts for this particular issue. Use it to stay safe and to save some typing.

MANIPULATING ARRAYS

Although the Array data type is one of the most commonly used, not all arrays are created equal—at least not among the various JavaScript implementations. Fortunately, Dojo's Array facilities provide an easy-to-use abstraction, ensuring that the code you write will work anywhere, and you won't be left scratching your head staring at a big semantic bug that's painful to track down. Consider the following (seemingly innocent) block of code:

```
var a = getMetasyntacticVariables();
if (a.indexOf("foo") != -1) {
  /* do something... */
}
```

Although you might swear that there couldn't possibly be anything wrong with that code, that's because you're probably (again) using and testing with a nice KHTML- or Gecko-based browser. The Trident-based Internet Explorer has its own notions of what an Array should and shouldn't do, and the indexOf method isn't one of them. In other words, you can expect for your code most likely to outright fail if you try to invoke the indexOf function on an Array when the code runs in IE. In this particular case, you could use the dojo.indexOf function to produce code safely that is portable across browsers:

```
var a = getMetasyntacticVariables();
if (dojo.indexOf(a, "foo") != -1) {
  /* do something... */
}
```

Other useful Array methods available via the dojo.* namespace include map, filter, every, some, lastIndexOf and forEach. They all work as described in the Mozilla Developer documentation.

At first glance, the forEach method may seem a bit redundant, because JavaScript provides a for loop construct, but forEach provides one particularly important feature that often escapes even many senior-level JavaScript programmers: block level scope. To illustrate, first consider the following two approaches to iterating over an Array:

```
// Approach 1:
var arr = getSomeArray();
for (var i in arr) {
  /* manipulate arr[i] */
}

/* The last value of i is available here because the
   for loop does not have its own block level scope.
   Ditto for any temporary variables
   defined between the braces. */

// Approach 2:
var arr = getSomeArray();
dojo.forEach(arr, function(item) {
  /* manipulate item */
});
/* Neither item nor any temporary variables are
   available here because the scope of the anonymous
   function protected this outer scope from it. */
```

STYLING NODES

Another function you'll use quite often for DOM manipulation is dojo.style, which acts as a setter when you pass it a node and a map of style properties as parameters and a getter when you pass it a node and a particular style property. In addition to providing an intuitive one-stop shop for style, it protects you from a number of DOM-based browser-specific quirks that otherwise would creep up on you. Here's how it works:

```
// Set some style properties..
var fooNode = dojo.byId("foo");
dojo.style(fooNode, {
    color : "red",
    background : "white",
    border : "blue"
});

/* ... Lots of interesting things
       happen in the meanwhile ... */

// Get a style property such as width...
var props = dojo.style(fooNode, "width");
```

On a related tangent, you can use any combination of the dojo.hasClass, dojo.addClass and dojo.removeClass functions to inspect and manipulate classes in the same intuitive manner:

```
var fooNode = dojo.byld("foo");
if dojo.hasClass(fooNode) {
  // do something...
  dojo.addClass(fooNode, "bar");
} else {
  //do something else...
  dojo.removeClass(fooNode, "baz");
}
```

QUERYING THE DOM

The dojo.query function provides a natural segue into general-purpose DOM manipulation and is based on CSS3 selectors. For example, you might use the following logic to query the DOM for any anchor tags and temporarily highlight the background to be yellow for a mouse-over event:

```
dojo.query("a")
.onmouseover(function(evt) {
  dojo.style(evt.target, {background : "yellow"});
})
.onmouseout(function(evt) {
  dojo.style(evt.target, {background : ""});
});
```

The statement inside the dojo.addOnLoad block queries the DOM for any anchor tags using the "a" CSS3 selector and returns a collection of nodes as a specialized subclass of Array called dojo.NodeList. Each of the dojo.NodeList

methods is then successively applied to the collection of nodes with the final result being returned so that it can be captured into a variable if desired. The dojo.NodeList class provides a number of useful methods, such as addClass, removeClass, style and the various Array functions that you already have seen. For example, if you are seduced by the elegant dot notation that dojo.NodeList provides, you may find yourself with an expression like the following:

```
// find anchors that are direct descendants of divs
var highlyManipulatedNodes = dojo.query("div > a")
.addClass("foo")
.removeClass("bar")
.onmouseover(function(evt) { /* ... you ... */})
.map(function(item) { /* ... get ... */})
.filter(function(item) { /* ... the ... */})
.forEach(function(item) { /* ... idea ... */});
```

It is especially noteworthy that the dojo.NodeList methods named after and triggered by DOM events, such as onmouseover or onblur, accept a single parameter that is a W3C *standardized* event object, so you are freed from the development and maintenance of yet another layer of subtle incompatibilities when developing a Web application. In fact, the next section investigates the very mechanism that makes this possible.

EVENT CHAINING

It's quite often the case that you'll need to chain together some events arbitrarily to produce an action/reaction effect. The dojo.connect function provides a seamless interface for arbitrarily connecting events and JavaScript Objects. For example, you already know that you could hook up a handler when a user mouses over a specific node by using dojo.query and assigning a function via dojo.NodeList's onmouseover method like so:

```
dojo.query("#foo") //find the node with id=foo
.onmouseover(function(evt) { /* ... */ });
```

An alternative implementation via dojo.connect is the following statement, which assembles the connection and returns a handle that can be disconnected later manually if the situation calls for the relationship to be torn down. For example, it's generally a good idea to tear down the handle explicitly before destroying nodes that are involved in the connection:

```
var h = dojo.connect(dojo.byId("foo"), "onmouseover", function(evt) {
  /* ... use the normalized event object, evt, here ... */
});

/* Later */
dojo.disconnect(h); //tidy up things...
```

Although the net effect is the same for the two implementations presented, dojo.connect seamlessly allows you to provide Objects as the context. For example, the following variation illustrates how to fire off an event handler whenever a particular function is invoked:

```
var obj = { // a big hash of functions...
  foo : function() { /* ... */ },
  bar : function() { /* ... */ }
}

// set the handler to fire whenever obj.foo() is run
dojo.connect(obj, "foo", function() {
  /* ... a custom event handler ... */
});

obj.foo(); // the custom handler fires automatically
```

If you want to use a particular scope with the custom handler, you can wedge it in as a third parameter. The parameters are all normalized internally. Here's how it would work:

```
var obj1 = { // a big hash of functions...
  foo : function() { /* ... */ },
  bar : function() { /* ... */ }
}

var obj2 = { // a big hash of functions...
  baz : function() { /* ... */ }
}

// fire the handler whenever obj.foo() is run
dojo.connect(obj1, "foo", obj2, "baz");

obj.foo(); // fire obj2.baz right after obj1.foo
```

PUB/SUB COMMUNICATION

Although dojo.connect provides a kind of direct action/reaction style of communication, the publish/subscribe metaphor has many highly applicable use cases in loosely coupled architectures in which it's not prudent for objects or widgets to know about one another's existence. This metaphor is simple enough to set up. The dojo.publish function accepts a topic name and an optional Array of parameters that should be passed to any subscribers. Becoming a subscriber to a topic is done through the dojo.subscribe function, which accepts a topic name and a function that is executed in response to the published topic. Here's a working example with a couple Function Objects:

```
function Foo(topic) {
  this.greet = function() {
    console.log("Hi, I'm Foo");

    /* Foo directly publishes information,
       but not to anywhere specific... */
    dojo.publish("/lj/salutation");
  }

}

function Bar(topic) {
  this.greet = function() { console.log("Hi, I'm Bar"); }

  / * Bar directly subscribes to information,
      but not from anywhere specific */
```

```
  dojo.subscribe("/lj/salutation", this, "greet");
}

var foo = new Foo();
var bar = new Bar();

foo.talk(); //Hi, I'm Foo...Hi, I'm Bar
```

A couple variations on the pub/sub metaphor are available, but the vanilla dojo.publish/dojo.subscribe functions relay the general idea. Any situation in which you cannot (for whatever reason) expose an API might be a prime opportunity to take advantage of pub/sub communication in your application.

OBJECT-ORIENTED JAVASCRIPT

JavaScript *is* an object-oriented programming language, but unlike the class-based languages of Java or C++, it uses prototypal inheritance as its mechanism of choice instead of a class-based paradigm. Consequently, mixing properties into object instances as part of a "has-a" relationship is often a far more natural pattern than attempting to mimic class-based patterns that espouse an "is-a" relationship. Consider the following example that adds in a collection of properties to an object instance all at once using dojo.mixin:

```
var obj = {prop1 : "foo"}

/* obj gets passed around and lots of
   interesting things happen to it */

// now, we need to add in a batch of properties...
dojo.mixin(obj, {
  prop2 : "bar",
  prop3 : "baz",
  prop4 : someOtherObject
});
```

The dojo.extend function works much like dojo.mixin except that it manipulates a constructor function's prototype instead of the specific object instances.

Of course, there are some design patterns that do lend themselves to inheritance hierarchies, and the dojo.declare function is your ticket to mimicking class-based inheritance if you find yourself in a predicament that calls for it. You pass it the fully qualified name of the "class" you'd like to create, any ancestors that it should inherit from, and a hash of any additional properties. The dojo.declare function provides a built-in construction function that gets run, so any parameters that are passed in can be handled as needed. Here's a short example demonstrating a Baz class that multiply inherits from both a Foo and a Bar class:

```
//create an lj.Foo that doesn't have any ancestors
dojo.declare("lj.Foo", null,
{
  /* custom properties go here */
  _name : null,
  constructor : function(name) {
    this._name = name;
  },
  talk : function() {alert("I'm "+this._name);},
  customFooMethod : function() { /* ... */ }
});

//create an lj.Bar that doesn't have any ancestors
dojo.declare("lj.Bar", null,
{
  /* custom properties go here */
  _name : null,
  constructor : function(name) {
    this._name = name;
  },
  talk : function() {alert("I'm "+this._name);},
  customBarMethod : function() { /* ... */ }
});

//create an lj.Baz that multiply inherits
dojo.declare("lj.Baz", [lj.Foo, lj.Bar],
{
  /* custom properties go here */
  _name : null,
  constructor : function(name) {
    this._name = name;
  },
  talk : function() {alert("I'm "+this._name);},
  customBazMethod : function() { /* ... */ }
});

//parameters get passed into the special constructor function
bartyBaz = new lj.Baz("barty");
```

When each of the dojo.declare statements is encountered, internal processing leaves a function in memory that can be readily used to instantiate a specific object instance with the new operator—just like plain-old JavaScript works. In fact, the bartyBaz object is one such instantiation. It inherits the customFooMethod and customBarMethod from ancestors, but provides its own talk method. In the event that it had not provided its own talk method, the last one that was mixed in from the ancestors would have prevailed. In this particular case, the ancestors were [lj.Foo, lj.Bar], so the last mixed in ancestor would have been lj.Bar. If defined, all classes created with dojo.declare have their parameters passed a special constructor function that can be used for initialization or preprocessing.

NOTE

Strictly speaking, there aren't classes in JavaScript—only objects exist. When discussing simulated inheritance situations, however, it is not uncommon to use the word "class" as though classes really do exist, provided appropriate caveats (like this one) are applied.

SERVER COMMUNICATION

No discussion of a JavaScript toolkit would be complete without a mention of the AJAX and server-side communication facilities that are available. Dojo's support for server-side communication via the XMLHttpRequest (XHR) object is quite rich, and the dojo.xhrGet function is the most logical starting point, because it is the most commonly used variant. As you might have suspected, it performs a GET request to the server. Unless you configure it otherwise, the request is asynchronous and the return type is interpreted as text. Here's an example of typical usage:

```
dojo.xhrGet({
  url : "/foo", //returns {"foo" : "bar"}
  handleAs : "json", // interpret the response as JSON vs text
  load : function(response, ioArgs) {
    /* success! treat response.foo just like a
       normal JavaScript object */
    return response;
  },
  error : function(response, ioArgs) {
    /* be prepared to handle any errors that occur here */
    return response;
  }
});
```

A point wasn't made of it, but the reason that both the load and error function returns the response type is because Dojo's I/O subsystem uses an abstraction called a Deferred to streamline network operations. The Deferred implementation was adapted from MochiKit's implementation (which was, in turn, inspired from Python's Twisted networking engine). The overarching concept behind a Deferred is that it provides a uniform interface that drastically simplifies I/O by allowing you to handle asynchronous and synchronous requests the very same way. In other words, you can chain callbacks and errbacks arbitrarily onto a Deferred, regardless of whether the network I/O is in flight, threw an Error or completed successfully. Regardless, the callback or errback is handled the same way. In some situations, Deferreds almost create the illusion that you have something like a thread at your disposal.

Here's an updated example of the previous dojo.xhrGet function call that showcases the dojo.Deferred object that is returned:

```
var d = dojo.xhrGet({
  url : "/foo", //returns {"foo" : "bar"}
  handleAs : "json", // interpret the response as JSON instead

  load : function(response, ioArgs) {
    /* success! treat response.foo just
```


SMALL, EFFICIENT COMPUTERS WITH PRE-INSTALLED UBUNTU.

GS-Lo8 Fanless Pico-ITX System
Ultra-Compact, Full-Featured Computer
Excellent for Industrial Applications

3677 Intel Core 2 Duo Mobile System
Range of Intel-Based Mainboards Available
Excellent for Mobile & Desktop Computing

DISCOVER THE ADVANTAGE OF MINI-ITX.
Selecting a complete, dedicated platform from us is simple: Pre-configured systems perfect for both business & desktop use, Linux development services, and a wealth of online resources.

ubuntu solution provider

LOGIC SUPPLY
www.logicsupply.com

```
        like a normal JavaScript object */
    return response; // pass into next callback
  },
  error : function(response, ioArgs) {
    /* be prepared to handle any errors that occur here */
    return response; //pass into next errback
  }
});

/* The xhrGet function just fired. We have no idea if/when
   it will complete in this case since it's asynchronous.
   The value of d, the Deferred, right now is null since it
   was an asynchronous request */

//gets called once load completes
d.addCallback(function(response) {
  /* Just chained on a callback that
     fires after the load handler with the
     same response that load returned. */
     return response;
});

d.addCallback(function(response) {
  /* Just chained on another callback that
     fires after the one we just added */
     return response;
});

d.addErrback(function(response) {
  /* Just added an errback that
     fires after the default error handler */
     return response;
});

/* You get the idea... */
```

Again, the beauty of a Deferred is that you treat it as somewhat of a black box. It doesn't matter if, when or how it finishes executing. It's all the same to you as the application programmer.

Just so you're aware, sending data to the server with another HTTP method, such as POST or PUT, entails using the very same kind of pattern and works just as predictably with the dojo.xhrPost function. You even can provide a form node so that an entire form is POSTed to the server in one fell swoop or pass in raw data for those times when you need to transfer some information to the server as part of a RESTful (Representational State Transfer-based) architecture. The dojo.toJson function may be especially helpful in serializing JavaScript objects into a properly escaped JSON string, if the protocol is something along the lines of a JSON-RPC system in which the envelope is expected to be JSON in both directions.

SPECIAL EFFECTS

Simple animations are generally a crowd-pleaser, and Base includes a few easy-to-use functions that make animating content a snap. For starters, consider the dojo.fadeIn and dojo.fadeOut functions:

```
dojo.fadeOut({node : "foo"}).play();


// then sometime later...
dojo.fadeIn({node : "foo"}).play();
```

Hopefully, that seemed as simple as it should be: point to a node and fade it. It won't be long though before you'll find the desire to do some animations that involve arbitrary CSS properties, and that's when the dojo.animateProperty function comes into play. The basic pattern is that you pass it a node, a map of properties and a duration, and it handles the rest. Here's a simple example that relates the pattern via the dojo.anim function by providing functions for imploding and exploding a node:

```
//implode a node...
dojo.anim("foo", properties : {width : 0, height : 0}, 500);
//implode over 500ms

/* ... Later ... */

//then explode it back out
dojo.anim("foo", properties : {width : 300, height : 300}, 500);
//explode back out over 500ms...
```

A number of other useful animation functions exist in the dojo.fx module.

HIGHLIGHTS FROM CORE

Although there is a ton of functionality jam-packed into Base, there are a number of other highly useful modules that you can get from Core at the expense of a dojo.require statement, which acts like #include from C++ or an import statement from Python or Java. Before providing an overview of what's available in Core, however, it's worth briefly summarizing how the dojo.require statement works, because it is a staple in the toolkit.

REQUIRING RESOURCES

In Dojo parlance, a dojo.require statement generally fetches an entire module or a resource that is part of a module, and a module is just a JavaScript file arranged according to a particular convention. For example, if you were to download a source distribution of Dojo and browse the contents of the dojo/io folder, you'd see that an iframe.js file and a script.js file are present. The first statement in each of these files is dojo.provide("dojo.io.iframe") and dojo.provide("dojo.io.script"), respectively. In this case, you'd say that the dojo.io module provides the iframe and script resources. The basic trade-off when designing modules is the balance between minimizing HTTP requests that incur a lot of latency and not downloading more content than you actually need. (The build tools included in Util, however, can consolidate multiple resource files into a single minified JavaScript file that nearly obliterates any concern surrounding this particular issue for many cases.)

Let's put dojo.require to work by having it retrieve the dojo.io.script resource that we'll use to fetch some public data using Flickr's JSON with padding (JSONP) API. Like

ubuntu LIVE

Take Ubuntu Further

Ubuntu Live is the official, trusted event for the Ubuntu community, welcoming all the different segments within the Ubuntu ecosystem—experts, long-time users, established companies, and newcomers alike in a passionate, energizing environment.

Tracks at Ubuntu Live

- **Ubuntu in Action:** Find Ubuntu in the community and in some unexpected places
- **Ubuntu Means Business:** How Ubuntu works for, and in, business settings
- **The Tech View:** Cool technology inside and on top of Ubuntu
- **Public Sector:** Learn how Ubuntu is changing the world, or at least the public sector and NGOs
- **Management Systems:** Tips and tools to deploy and manage Ubuntu systems

Register Now and Save 15%

Use discount code **ubu08ljr**

www.ubuntulive.com

July 21 – 22, 2008 • Oregon Convention Center • Portland, OR

Co-presented by CANONICAL O'REILLY

THE OVERARCHING CONCEPT BEHIND A DEFERRED IS THAT IT PROVIDES A UNIFORM INTERFACE THAT DRASTICALLY SIMPLIFIES I/O BY ALLOWING YOU TO HANDLE ASYNCHRONOUS AND SYNCHRONOUS REQUESTS THE VERY SAME WAY.

almost everything else in the toolkit, the dojo.io.script.get function that we'll use abstracts most of the dirty work away, so you don't have to write or maintain any of that brittle boilerplate:

```
//Require what you need...
dojo.require("dojo.io.script");


//...but don't reference it outside of the dojo.addOnLoad
//block or you'll create a race condition since dojo.require
//statements are satisfied asynchronously over the CDN...
dojo.addOnLoad(function() {
  dojo.io.script.get({
     callbackParamName : "jsoncallback", //provided by Flickr
     url: "http://www.flickr.com/services/feeds/photos_public.gne",
     load : function(response, ioArgs) {
      /* response is a JSON object with data about public photos */
      return response;
    },
    error : function(response, ioArgs) {
      /* ... handle the error ... */
      return response;
    }
  });

}
```

MORE CORE

Although there's not time to gloss over Core systematically the same way we did with Base, it's well worth the time to explore it, and you're now equipped with enough fundamental knowledge to go do some digging on your own. A few of the resources you'll find in Core include:

- Internationalization facilities and functions for computing dates, times and formatting currency.
- Additional animation capabilities.
- The IFRAME transport (useful for uploading files to the server).
- Functions for handling cookies.
- Powerful data APIs that abstract cumbersome server-side I/O.
- Drag-and-drop machinery.

MORE TOOLKIT

And even though Base and Core are two substantial components of the toolkit, there's also Dijit, DojoX and Util, which easily could span entire articles and books of their own. In short, Dijit is a huge infrastructure of DHTML-based turn-key widgets. DojoX is a collection of specialized subprojects ranging from wrappers around Google Gears to graphics functions that can use scalable vector graphics (SVG) to draw in the browser natively. Util provides DOH, a unit-testing framework and a collection of build tools for squeezing as much performance as possible out of your JavaScript. Although not covered in this article, these topics are well worth investigating and are just as important to your development efforts as Base and Core.■

Matthew Russell is a computer scientist and technologist whose latest pastime entailed authoring *Dojo: The Definitive Guide*. With what little time is left over from writing, hacking on Web technology and noodling on hard computer science problems, he enjoys getting as far away from technology as possible by escaping for long bike rides near his home in Franklin, Tennessee.

Resources

The Dojo Toolkit: **dojotoolkit.org**

The Official Dojo API: **redesign.dojotoolkit.org**

Firebug: **getfirebug.com**

Mozilla's Developer Documentation on Arrays: **developer.mozilla.org/en/docs/Core_JavaScript_1.5_Reference:Global_Objects:Array**

CSS3 Selectors: **www.w3.org/TR/css3-selectors/#selectors**

W3C Documentation on DOM Events: **www.w3.org/TR/DOM-Level-2-Events/events.html**

Python's Twisted Matrix: **twistedmatrix.com**

MochiKit Deferreds: **www.mochikit.com/doc/html/MochiKit/Async.html#fn-deferred**

REST: **en.wikipedia.org/wiki/Representational_State_Transfer**

JSON-RPC: **json-rpc.org**

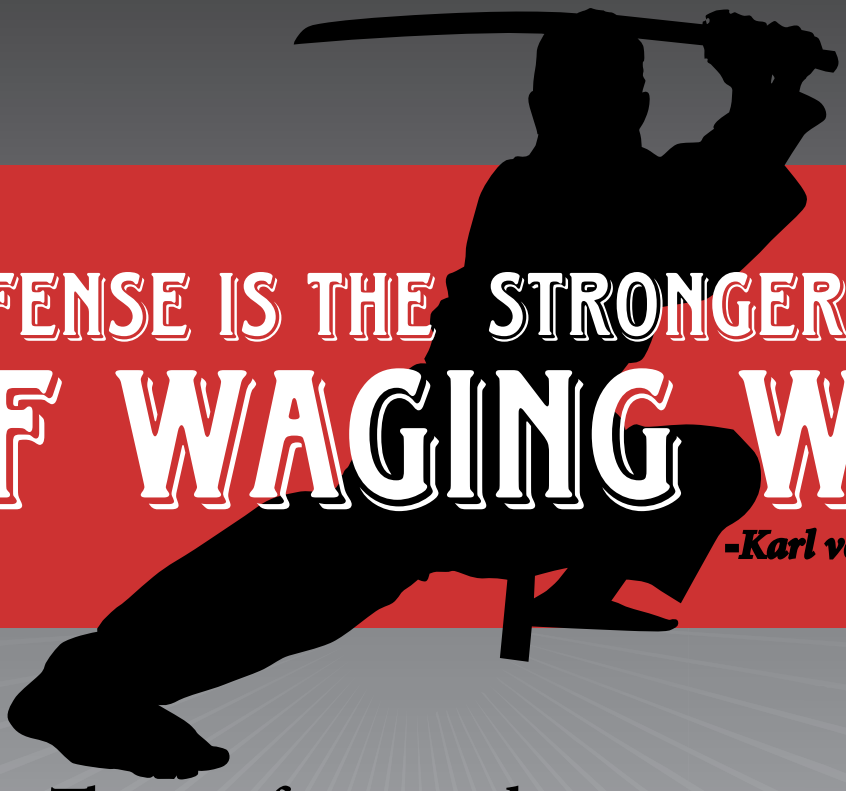
JSONP: **ajaxian.com/archives/jsonp-json-with-padding**

Flickr API: **www.flickr.com/services/api**

Clean Licensing: Why You Should Care (If You Don't Already): **alex.dojotoolkit.org/?p=654**

Dojo: The Definitive Guide by Matthew Russell: **www.amazon.com/Dojo-Definitive-Guide-Matthew-Russell/dp/0596516487**

Black Hat USA 2008



"DEFENSE IS THE STRONGER FORM OF WAGING WAR"

-Karl von Clausewitz

The war for your data rages on.
Be certain your defenses are up to the job.

Black Hat USA convenes the best infosec minds on the planet for six days of intense, hands-on security education and peer-to-peer networking. Our speakers and trainers are the world's leading voices from academia, research and the underground. The breadth and depth of topics is unmatched. You will gain actionable knowledge, discover new tools, and learn expert techniques for digital self defense.

12 tracks **80** presentations **40** training sessions

August 2-7 2008
Caesars Palace

Black Hat
BRIEFINGS AND TRAINING
www.blackhat.com

Las Vegas
Nevada, USA

Diamond Sponsor: Microsoft

Platinum Sponsors: Cisco, Qualys

Gold Sponsors: Configuresoft, CORE, Google, IOActive, NORMAN

Silver Sponsors: ArcSight, Breach, BigFix, Cenzic, Fortify, HP, nCircle, Ounce Labs, radware, SAINT, SecureWorks, Soteria, StillSecure, WhiteHat

Introducing OpenLaszlo 4

Toward desktop-like rich Internet applications with OpenLaszlo 4.

PAUL BARRY

Users' expectations on the Web are changing. No longer content to fill in forms and wait for pages to refresh, modern Web users want to interact with the browser in the same way they do with any other desktop application—that is, interactively and instantly. Recently, this has been made easier by the emergence of AJAX and its JavaScript-centred programming paradigm. But, AJAX is not the only technology that can help here. There is, of course, the ubiquitous Flash technology, but its closed-source heritage puts off the community. And, then there's OpenLaszlo.

Originally a Flash-centric technology, OpenLaszlo Release 4 (hereafter referred to as OpenLaszlo) breaks free from its Flash heritage and supports DHTML as an additional deployment platform. This means that applications written in OpenLaszlo can execute in any browser that supports DHTML or Flash's SWF, which practically covers every major browser on every operating system. As the name suggests, OpenLaszlo is an open-source product, released under the Common Public License, and OpenLaszlo's creators, Laszlo Systems, are keen to see a strong open-source developer community form around this main product offering.

OpenLaszlo is billed as a Rich Internet Application (RIA) development platform. Its goal in life is to add desktop-like functionality to browser-based applications, and it accomplishes this in a non-conventional, yet highly productive way. In this article, I explain how to install and configure OpenLaszlo, and then I present a few small example applications showcasing some of what OpenLaszlo has to offer.

Installing and Configuring OpenLaszlo

OpenLaszlo is a Web development platform built on top of release 1.4 of the Java SDK. Packaged as a Java Servlet, OpenLaszlo can be dropped into any compatible Java Servlet container. The Apache Tomcat server comes with the OpenLaszlo distribution and is already configured and ready to go, so I use it in this article. Whether or not you use Tomcat, it still is necessary to install the Java SDK before attempting to install OpenLaszlo (assuming it's not already installed). If you are on a Debian-derived version of GNU/Linux (like me), installing release 1.4 of the SDK is a breeze:

```
sudo apt-get install j2sdk1.4
```

Users of non-Debian distributions should check their package repositories for the Java SDK and install appropriately. Once the Java SDK install is complete, edit the /etc/bash.bashrc file as root, adding the following lines to the end of the file:

```
export JAVA_HOME="/usr/lib/j2se/1.4"
export PATH=$JAVA_HOME/bin:$PATH
```

These lines effectively allow Java programs to find the Java runtime environment. Be sure to set these environment variables, as without them, nothing works. With the Java SDK configured, it's time to get OpenLaszlo. Download the latest compressed tarball from the OpenLaszlo site (see Resources), then copy it to your /usr/local directory:

```
sudo cp openlaszlo-4.0.10-unix.tar.gz /usr/local
```

At the time of this writing, the latest and greatest OpenLaszlo is release 4.0.10. Be sure to adjust the release number within these instructions if you're using a newer release. Change directory to /usr/local, and unpack the distribution:

```
cd /usr/local
sudo tar zxvf openlaszlo-4.0.10-unix.tar.gz
```

This creates an lps-4.0.10 directory under /usr/local with all the OpenLaszlo goodies unpacked in place. Of importance is the existence of the Tomcat server under the newly created Server directory at lps-4.0.10/Server/tomcat-5.0.24/. To start the server with the OpenLaszlo servlet preconfigured, type:

```
sudo /usr/local/lps-4.0.10/Server/tomcat-5.0.24/bin/startup.sh
```

which results in the following output:

```
Using CATALINA_BASE:   /usr/local/lps-4.0.10/Server/tomcat-5.0.24
Using CATALINA_HOME:   /usr/local/lps-4.0.10/Server/tomcat-5.0.24
Using CATALINA_TMPDIR: /usr/local/lps-4.0.10/Server/tomcat-5.0.24/temp
Using JAVA_HOME:       /usr/lib/j2se/1.4
```

Tomcat and Openlaszlo are now up and running on port 8080.

Testing OpenLaszlo

An OpenLaszlo test page is provided, and you can access it by typing the following URL into the browser: http://localhost:8080/lps-4.0.10/examples/hello.lzx.

This results in the string "Hello Laszlo!" appearing within the browser after a few seconds. (The first time, OpenLaszlo takes a while to load, but subsequent reloads are as quick as a flash.) Ask your browser to view the HTML source, and a perfectly formed page of HTML is displayed, albeit missing a little human-readable white-space.

The output produced is created by an OpenLaszlo application, written in a declarative, XML-based programming language called LZX. Here's the source code to hello.lzx, which is pretty much run-of-the-mill XML:

```
<canvas>
    <text>Hello to Linux Journal from Laszlo!</text>
</canvas>
```

This simple example illustrates an important point about OpenLaszlo. Openlaszlo's programming language is declarative in nature, not procedural. What this means is that you specify what you want OpenLaszlo to do as opposed to specifying how OpenLaszlo is to go about performing what you want done. OpenLaszlo then works out the series of steps that need to be performed and performs them for you. (In a way, this is exactly like how regular expressions work, in that you specify the pattern you are looking for, not how to find it.) So, when you program OpenLaszlo, you declare the behaviour you require in LZX, and you write LZX in XML. Hard-core programming types might think that writing code in XML is far too unwieldy. But, it's not code per se; it's a declaration of the desired behaviour. Once you get your head around this idea, LZX and OpenLaszlo make quite a bit of sense.

Building an OpenLaszlo Application

The root XML node in all OpenLaszlo applications is the <canvas> tag, which contains the declarative code that describes the application's behaviour. To see how all this works, let's play with some LZX code, building on the simple test application above. Create a file called ljhello.lzx, and put the following LZX code in it:

```
<canvas>
  <window title="My First App"
          x="50" y="50"
          height="200" width="500"
          realizable="true">
    <text id="message"
          text="Hello from Linux Journal!"/>
  </window>
</canvas>
```

Save the file, then copy it to a location where OpenLaszlo and the Tomcat server can find it:

```
sudo cp ljhello.lzx /usr/local/lps-4.0.10/Server/lps-4.0.10/
```

Typing http://localhost:8080/lps-4.0.10/ljhello.lzx into a browser results in the creation of a movable, realizable window, as shown in Figure 1.

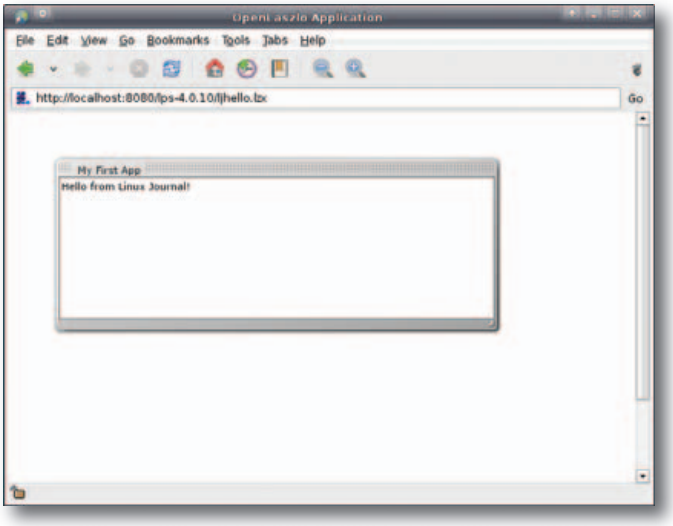


Figure 1. Our First OpenLaszlo Application: ljhello.lzx

Referring to the LZX code, it is not too difficult to work out what's specified for this application. We start with a blank canvas, then create a window that has a title, an x/y position, height/width values and the realizable property switched on. Within the window, we ask for some text, give the text an identifier and an initial value. Note how the use of indentation within the LZX code helps to describe which components of the application are related to which other components rather naturally. Within the browser, the resulting window can be grabbed and dragged, as well as resized.

OpenLaszlo and Data

For the purposes of demonstration, let's imagine we have a small on-line store that wants to provide access to its client list via a nice, modern Web-based interface. To provide the required functionality, let's put the data into a MySQL database and provide access to the list via an OpenLaszlo application.

To begin, log in to the MySQL client as root, then create a database called store and a MySQL user called store_manager:

```
mysql> create database store;
mysql> use mysql;
mysql> grant all on store.* to store_manager identified by 'passwordhere';
```

Log in to MySQL as this new user, and create a table to hold the client list:

```
mysql -u store_manager -p store

mysql> create table client_details
       (
               id int not null auto_increment primary key,
               name varchar (64) not null,
               address varchar (255),
               contact_tel_no varchar (64),
               email_address varchar (64)
        );
```

LINUX HAS NEVER LOOKED SO GOOD

Today's real-world open source solutions are powerful, dynamic, and green enough to deliver proven business value in every part of the enterprise: Business-critical systems. Mobile devices. Desktop apps. The next-generation data center.

Don't just dream about what open source could do for your business. Come to the industry's definitive resource for Linux and open source products, training, and insight—and bring back the solutions your business has been waiting for.

REGISTER TODAY AT
www.linuxworldexpo.com

LINUXWORLD®
CONFERENCE & EXPO

AN IDG WORLD EXPO EVENT

August 4-7, 2008
The Moscone Center
San Francisco, CA

LinuxWorld is open to business professionals only. No one under 18 years of age will be admitted.

A small collection of SQL insert statements (formatted to fit this page) provides us with some data to play with:

```
mysql> insert into client_details value ( 0,        \
'Joe Bloggs', '25 Somewhere Street, Anytown, USA', \
'00-1-415-555-3226', 'joe@bloggs.com' );
mysql> insert into client_details value ( 0,        \
'Jane Doe', 'Apt. 2a, 16 Treatsville, Canada',      \
'00-1-416-555-1222', 'jane@idontknow.ca' );
mysql> insert into client_details value ( 0,        \
'Harry Smith', 'P.O. Box 46, Streetstown, USA',     \
'00-1-581-555-9823', 'harry@harrysmith.com' );
mysql> insert into client_details value ( 0,        \
'Julie Jones','CharmsRus, BT Tower, London, UK',    \
'00-44-081-555-2398', 'julie@charmsrus.co.uk' );
```

With the database table ready, and some sample data inserted, we next need to get the data into a format that OpenLaszlo can understand. It shouldn't surprise you to learn that the best format for your data when communicating with OpenLaszlo is XML. OpenLaszlo has some rather neat, built-in functionality for working with XML data. To demonstrate this, we first have to arrange for MySQL to produce some XML output.

There are a number of ways to do this, and I'm going to write a simple CGI in Ruby that connects to the database, selects all the data from the required table and turns it into

OpenLaszlo has some rather neat, built-in functionality for working with XML data.

XML. My program, called get_data.rb, will execute from Apache's CGI directory, which is /usr/lib/cgi-bin on my system. Here's the Ruby code I wrote:

```
#! /usr/bin/ruby

require 'cgi'
require 'dbi'

resp = CGI.new

puts resp.header( "text/xml" )

dsn =  "DBI:Mysql:store"
user = "store_manager"
pass = "passwordhere"

sql =  "SELECT * FROM client_details"

DBI.connect( dsn, user, pass ) do |dbh|
    rows = dbh.select_all( sql )
    DBI::Utils::XMLFormatter.table( rows,
                                    "clients",
                                    "client" )
end
```

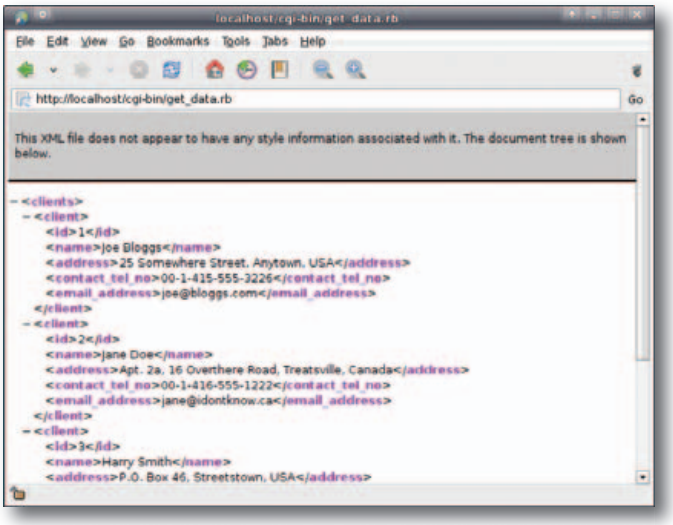


Figure 2. The XML Output Produced by the get_data.rb CGI Script

This code is straightforward. The key line is the call to DBI::Utils::XMLFormatter, which takes the result of the SQL query and produces correctly formatted XML. To see the results, install get_data.rb into Apache's cgi-bin directory (setting get_data.rb to be executable), and then type the following into a browser: http://localhost/cgi-bin/get_data.rb. Figure 2 shows the XML produced by the get_data.rb CGI script.

To access this data from within an OpenLaszlo application, all that's required is the appropriate declaration using the LZX dataset tag. Here's another file, called clients.lzx, which displays the name of each of the store's clients in an OpenLaszlo window:

```
<canvas>
  <dataset    src="http://localhost/cgi-bin/get_data.rb"
              name="dataClients"
              request="true" />

  <window     title="Client Listing"
              name="top"
              height="300" width="200"
              x="50" y="50"
              realizable="true">
    <view>
      <text>
        <datapath xpath="dataClients:/clients/client/name/text()"/>
      </text>
      <simplelayout/>
    </view>

    <scrollbar/>

  </window>

</canvas>
```

As in the previous example, there's a window with some text in it. Note that the text is contained within an LZX

CALL FOR PAPERS NOW OPEN

JOIN US

ONTARIO LINUX FEST

OCTOBER 25, 2008
TORONTO, ONTARIO CANADA

Get all of the details at http://onlinux.ca

Phone: 877-TUX-FEST 866-883-1172 x5102 info@onlinux.ca

LINUX Pro MAGAZINE

waterloodatafortress

netdirect open solutions

LINUX JOURNAL
Since 1994: The Original Magazine of the Linux Community

view, which combines the text with something called simplelayout, an in-built OpenLaszlo style that stacks text one line on top of another. The window, called top, also has a scrollbar associated with it. The dataset LZX tag informs the OpenLaszlo application where to get the data from (src), what to call the dataset (name) and instructs the application to go and get the data as soon as it is loaded (request). The datapath tag is a standard XML XPath specification pointing to the dataset that we want to use. In this case, we want to retrieve the text of the name tag, which is contained within the inner-enclosing client tag, which is itself contained within the outer-enclosing clients tag. Referring back to Figure 2, it is easy to see the data that we are referring to within this XPath specifier.

To try out this application, copy the LZX file to the appropriate directory on the server (using the same destination directory as for the ljhello.lzx file), then start the application running within your browser using the following URL: http://localhost:8080/lps-4.0.10/clients.lzx.

This produces an OpenLaszlo window with the names of the four clients displayed within it, as shown in Figure 3.

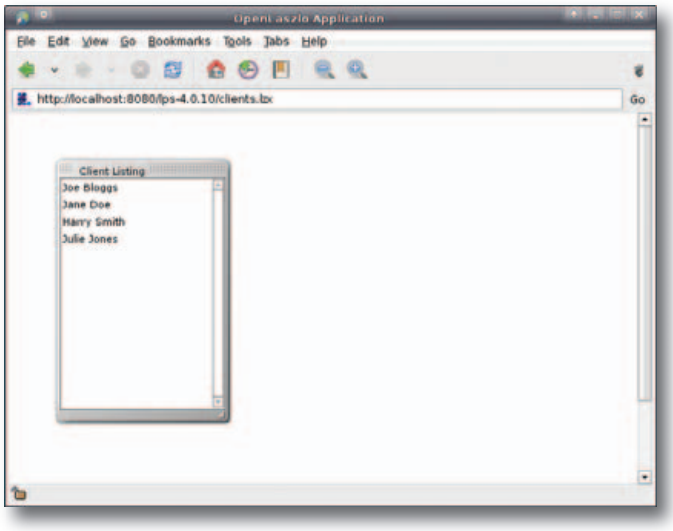


Figure 3. Displaying a List of Names within the OpenLaszlo Application

Adding Interactivity

Although of interest, this list would be made more useful if a single click on the client name produced another window within the browser containing the rest of the client's data. Arranging for this behaviour is not difficult. The first thing we need to do is provide some visual feedback to our users as they select a client name from the first window. Add this code to the window's <text> element:

```
<handler name="onclick">
   client_info.datapath.setFromPointer( this.datapath );
</handler>

<handler name="onmouseover">
   this.setBGColor( 0xBBBBFF );
</handler>

<handler name="onmouseout">
   this.setBGColor( null );
</handler>
```

This snippet of LZX highlights OpenLaszlo's ability to embed JavaScript within XML elements. What this code instructs the browser to do is to set the data pointer for something called client_info to the currently selected datapath once users click a name on the list. It also changes the background color as users move their mouse over the client names, providing nice, desktop-like visual feedback. But, what's this client_info thing, and what does it refer to? It's another OpenLaszlo window defined with the following LZX code:

```
<window name="client_info"
        x="300" y="100"
        width="300" height="200"
        title="Client Specifics">

  <datapath/>
    <text datapath="../address/text()"
          width="100%"
          multiline="true" />
    <text datapath="../contact_tel_no/text()"
          fontsize="16"/>
    <text datapath="../email_address/text()"
          fontsize="14"/>
    <simplelayout/>

</window>
```

This window has its own name and title values, as well as x, y, width and height values that position it initially to the right of the client listing window. It also has a datapath tag, together with three text elements that reference (using an appropriate XPath specification) the other data elements within our database table. We've specified that the address uses the entire width of the client_info window and can word wrap, while the other two pieces of

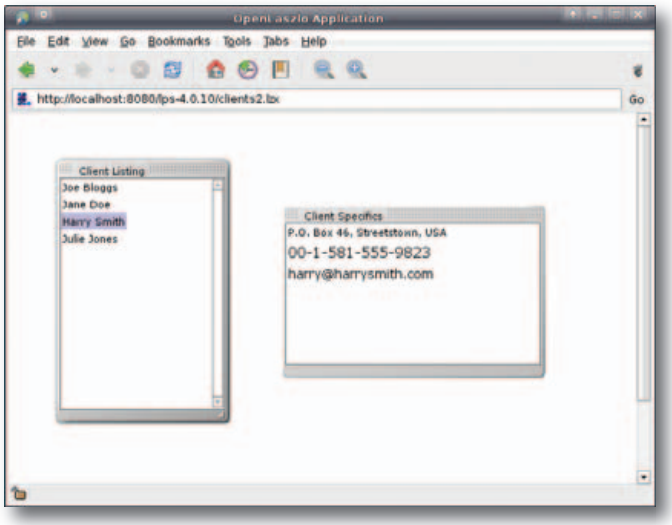


Figure 4. Displaying Specific Details for a Selected Client

data are displayed in differently set font sizes. When this LZX application (called clients2.lzx) is loaded into the browser, the client list appears in the original window, and as each client name is clicked, the second window refreshes to display the address, telephone number and e-mail address of the currently selected client. If you are following along, note how the user receives visual feedback as each client name is clicked. Figure 4 shows an example, with one client name highlighted (clicked) and the associated details appearing in the second window.

Adding Animation

Let's finish this example with a bit of fun by adding some LZX animation effects to our OpenLaszlo application. Specifically, whenever users click on a client name in the first window, in addition to refreshing the data, we want the second window to roll up (shrink), pause, and then roll back down again (grow). To make this work, we need to wrap the onclick handler code with calls to our animators:

```
<handler name="onclick">
  client_info.winShrink.doStart();
  client_info.datapath.setFromPointer( this.datapath );
  client_info.winGrow.doStart();
</handler>
```

Specifying animation with LZX involves writing XML. Here's the shrinking and growing LZX code for this application (which I've called client3.lzx). This code is added to the second window's XML:

```
<animatorgroup name="winShrink"
               start="false"
               duration="0">
    <animator attribute="height" to="50"/>
    <animator attribute="height" to="50"/>
</animatorgroup>

<animatorgroup name="winGrow"
               start="false"
               duration="200">
    <animator attribute="height" to="200"/>
    <animator attribute="height" to="200"/>
</animatorgroup>
```

I define two animatorgroups and give each of them a name. Note how the animatorgroup name is referenced within the onclick handler, above. Within each animatorgroup, I provide some timing data (duration) and new attribute values for the height of the window. When the window shrinks, the height drops to 50 pixels. When the window grows, the height rises to 200 pixels. When combined, the visual effect is that of the window rolling up, pausing, then rolling back down to display the updated client details. Unfortunately, I can't show this in a screenshot, so you'll have to try it to see the effect in action (or take my word for it). The main point, of course, is that the visual effect has been realised without writing code, per se. All I did was define the behaviour I wanted in LZX.

Learning More about OpenLaszlo

Check out the Laszlo Systems Web site for more information on OpenLaszlo (see Resources). Be sure to take 30 minutes to view the rather excellent screencast provided, which has OpenLaszlo guru Adam Wolff stepping through some of the technology's features (note: Adam's video inspired much of the material in this article). To learn most of what there is to know about LZX, check out the Manning Publications book: *Laszlo in Action* (see Resources). All of the LZX and Ruby code presented in this article is available for download from the *Linux Journal* FTP site (see Resources).■

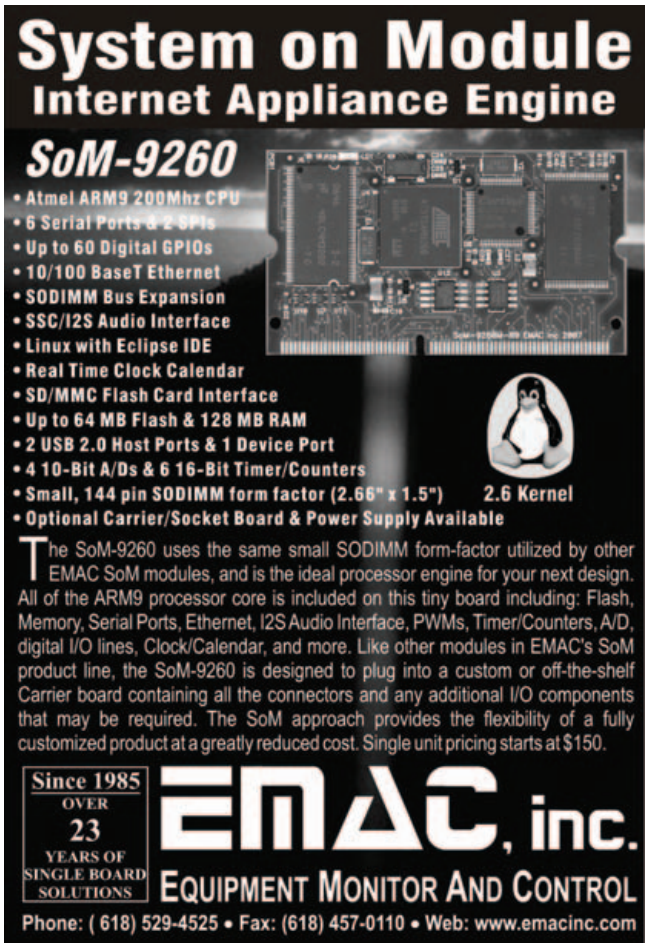
Paul Barry (paul.barry@itcarlow.ie) lectures at the Institute of Technology, Carlow in Ireland. Find out more about the stuff he does at his Web site: glasnost.itcarlow.ie/~barryp.

Resources

The OpenLaszlo Home Page: **www.openlaszlo.org**

Laszlo in Action by Manning Publications: **laszloinaction.com**

Article Source Code:
ftp.linuxjournal.com/pub/lj/issue171/10071.tgz



Semantic Web Publishing with RDFa

How you can use RDFa to embed structured content into your Web page and be part of the Semantic Web.

GOLDA VELEZ

I learned UNIX from a real old-style guru named Jimmy who memorized microchip numbers and used sed as a word processor. Wanting to do well on my first job, I proudly showed him how I was putting detailed comments in my code. My mentor was not impressed. "Why are you doing that?" he shot at me, going on to explain that neither comments nor docs could be trusted. If you wanted to understand what the code was doing, you better read the code. Software project managers might not agree, but Jimmy did have a point. Docs and comments can become out of date or inaccurate, but the code can't. Broken, yes. Inaccurate, no.

A similar issue arises when writing a Web page that is intended to be read by humans and parsed by machines. New sophisticated search engines on the horizon will be hungry for semantic content—that is, for data that can be machine-parsed for meaning. Often the format will be some form of RDF, or Resource Description Framework. If you are publishing Web pages in order to share your data with the world, it follows that you want to make it available to both humans and search engines. Generating two sets of files, one human-readable HTML and another machine-parsable RDF, means that you give up the ability to hand-edit your HTML files to make corrections and sets up your site for likely inconsistency down the road—not to mention that full-on RDF/XML is verbose and ugly.

Enter RDFa, a lightweight relatively new mechanism for embedding structured data into HTML in a simple but fully standards-compliant way. I run a Web site that is generated from templates. To understand how RDFa might fit in to my site, I started with a simple manually created example: an event schedule for the local rodeo. Later in this article, I also briefly cover some of the emerging tools that automate RDF and RDFa and describe how one company has created a large-scale RDF implementation to solve enterprise problems. Now, here's the example.

My original sample code looked like this, in vanilla HTML:

```
<div>
  <h1>Saturday Rodeo Schedule 2/22/08</h1>
  <div>
    2:00PM : Bull Riding
  </div>
</div>
```

It's pretty straightforward and clear to the human reader of the Web page or even someone editing the source, but it's meaningless to a search engine. To make this event clear to an RDFa-parsing engine, my first step was to pick a vocabulary that has well-defined terms for events. Luckily, there is just such a vocabulary, based on the iCalendar standard for calendar data. The vocabulary or vocabularies used in a document are specified right in the <html> tag at the start of the document:

```
<html xmlns="http://www.w3.org/1999/xhtml"
 xmlns:cal="http://www.w3.org/2002/12/cal/ical#"
>
```

The xmlns stands for XML NameSpace, and cal is the shorthand name we'll use to refer to this namespace further down. The http://www.w3.org/2002/12/cal/ical# is the URL to the RDF vocabulary file, and http://www.w3.org/1999/xhtml is the URL for the standard XML namespace that you might already be including in your documents. I explain further on discovering those and deciding which to use in a bit. Applying a bit of RDFa using basic iCal properties, we have this:

```
<div id=RodeoSchedule2008>
  <h1>Saturday Rodeo Schedule 2/22/08</h1>
  <div rel="cal:Vevent">
    <span property="cal:dtstart"
content="20080222T1400-0700">2:00PM</span>
    :
    <span property="cal:summary">Bull Riding</span>
  </div>
</div>
```

From the browser's point of view, the HTML layout is unchanged. If desired, class= properties could be added for CSS formatting and would not impact the RDFa logical structure. This is different from the microformat hCalendar (another popular way of representing calendar data in HTML), in which fixed class names are assigned.

One last step alerts parsers to the presence of RDFa in our document and also specifies the encoding or character set used. We add the following lines at the very beginning of the file, before the <html..> tag:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC
 "-//W3C//DTD XHTML+RDFa 1.0//EN"
 "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
```

Now, any application that understands RDFa can scan your Web page and learn that there is an event called Bull Riding occurring on February 22, 2008 at 2:00 PM PST. In fact, you can verify that you've communicated correctly with the RDFa world by using any of a number of validating/parsing services. Using the Python-based service at **www.w3.org/2007/08/pyRdfa**, called RDFa Distiller, we can see that the above snippet produces the following semantic data, in what is called the N3 format:

```
@prefix cal:<http://www.w3.org/2002/12/cal/ical#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.

 [ a cal:Vevent;
    cal:dtstart "20080222T1400-0700";
    cal:summary "Bull Riding"].
```

N3 is a shorthand that people who work heavily in the RDF world like to use for writing and representing the triples that compose the Semantic Web.

Web 3.0: the Semantic Web

Now, it's time to back up a bit. The term Semantic Web is used in this article to refer to the goal of a machine-parsable Web of structured data, as envisioned by Tim Berners-Lee in his 2001 *Scientific American* article by that name. Although there still is plenty of spirited debate over exactly how Web 3.0 will take shape, the W3 folks and others have been working diligently on a core set of technologies that has started to gain serious traction in the wild. Check out the layercake diagram from the W3C (Figure 1).

RDF, the data model on which the whole thing is based,

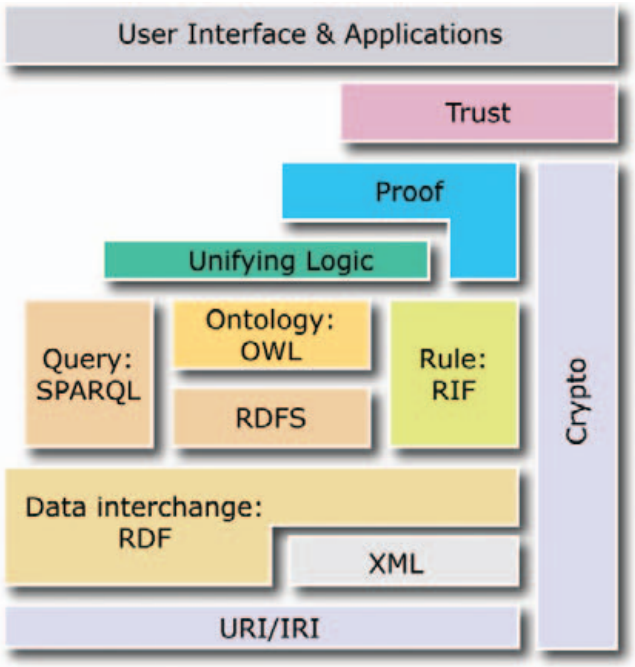
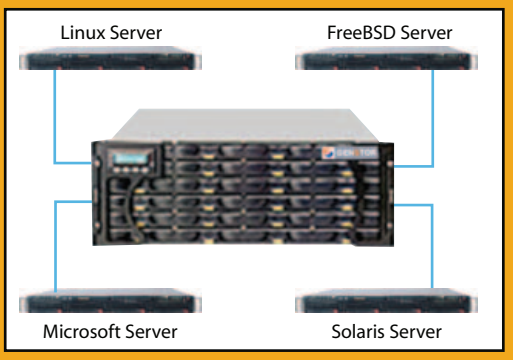


Figure 1. Layercake Diagram from W3C (source: www.w3.org/2001/sw/layerCake.png)

GENSTOR

Linux - FreeBSD - x86 Solaris - MS etc.



GENSTOR STORAGE SOLUTIONS:

- Storage options - FC to SATA/SAS, FC to FC SAS to SAS/SATA, SCSI to SATA, SCSI to SCSI
- Exceptional Performance with Proven Reliability
- 24 TB in 4U with easy upgrade path
- Host Servers and Storage comes Pre-Configured with heterogeneous OS- Linux, * BSD, Solaris Microsoft etc.
- Fully redundant Storage solutions

Proven technology. Proven reliability

When you can't afford to take chances with your business data or productivity, rely on a GS-1245 Server powered by the Intel® Xeon® Processors.

Quad Core Woodcrest

2 Nodes & Up to 16 Cores - in 1U

Ideal for high density clustering in standard 1U form factor. Upto 16 Cores for high CPU needs. Easy to configure failover nodes.

Features:

- 1U rack-optimized chassis (1.75in.)
- Up to 2 Quad Core Intel ® Xeon ® Woodcrest per Node with 1333 MHz system bus
- Up to 16 Woodcrest Cores Per 1U rackspace

Servers : : Storage : : Appliances

Genstor Systems, Inc.
780 Montague Express. #604
San Jose, CA 95131
www.genstor.com
Email: sales@genstor.com
Phone: 1-877-25 SERVER
1-408-383-0120

intel Xeon inside

Intel®, Intel® Xeon® Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries

represents the world as a set of triples: subject, predicate and object. Each item in the triple can be a URI, a literal or a blank node (a kind of temporary variable). In practice, the predicate is likely to be a URI in a namespace created for the purpose, like cal:dtstart or cal:summary.

Vocabularies and ontologies form the backbone of the Semantic Web. You can define your own, and some tools like Semantic MediaWiki create an ontology for you automatically. When defining the terms in a specialized domain, or when creating a private within-enterprise application, creating your own ontology makes sense. For sharing data with the world, I prefer to reuse existing vocabularies as much as possible. (By vocabulary, I mean an RDF file that defines terms and properties; by ontology, I mean a vocabulary that also contains logical rules.) Some widely used vocabs include the following:

- foaf: friend of a friend, for identifying people and other entities (**xmlns.com/foaf/spec/20071002.rdf**).
- ical: based on the iCalendar W3 standard, for calendar and event data (**www.w3.org/2002/12/cal/ical**).
- vcard: intended as an electronic business card, it has simple fields for contact information (**www.w3.org/2001/vcard-rdf/3.0**).
- dc: Dublin Core, defining core properties like title and creator (**purl.org/dc/elements/1.1**).
- cc: for Creative Commons licenses (**creativecommons.org/ns**).
- rss: the RSS 1.0 namespace (**purl.org/rss/1.0**).

Note that in our document, we can choose our own shorthand name for each vocabulary when we list it in the <html> tag. Then, we can use that shorthand to write what is called a CURIE, or Compact URI, like dc:title or foaf:name. In RDFa, those CURIEs are valid URIs and are much easier to read once you get used to them. One of the core ideas of RDF is to be able to use URIs to refer to concepts and things outside cyberspace, and then use them to make logical statements. So, it helps if the URIs are human-readable.

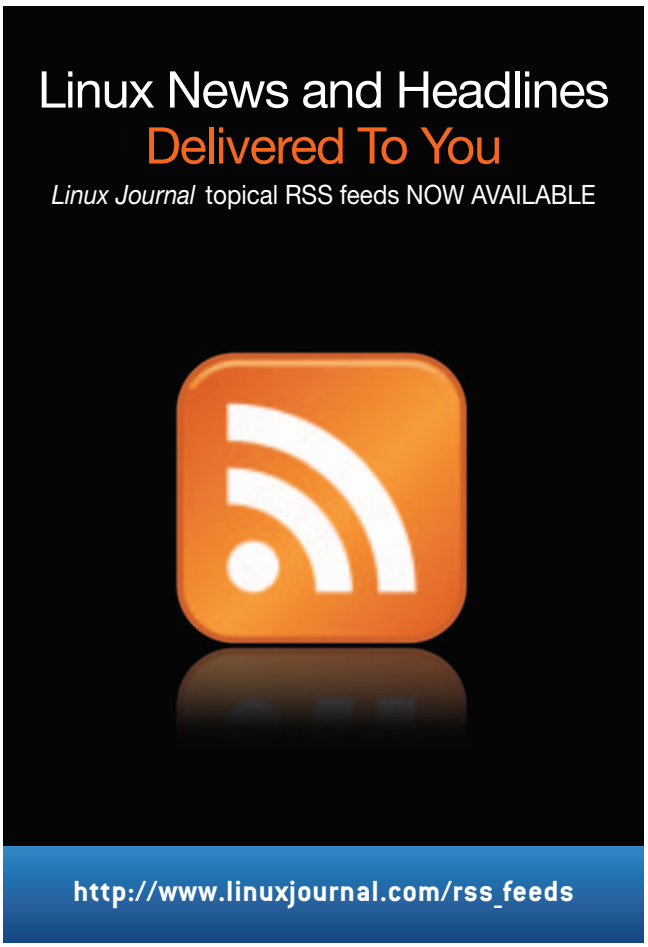
Going back to the rodeo schedule example, suppose we want to list the contestants in each event. Now, we get into the power of RDFa—the ability to connect different types of data together in a logical way right in an HTML file. The first step is to pick or create a vocabulary to describe the contestants. FOAF is the standard for referring to people, but I also want to specify that they are contestants in the rodeo. I did a search on Swoogle for the word contestant, and after a few minutes examining the available ontologies, I decided that http://smartweb.semanticweb.org/ontology/sportevent is the most apt. I also want to add a contact person for the rodeo as a whole, using the vCard vocabulary. So, I added foaf, contact and sportevent vocabularies to the list at the start of the document, which now looks like this:

```
<html xmlns="http://www.w3.org/1999/xhtml"
 xmlns:cal="http://www.w3.org/2002/12/cal/ical#"
 xmlns:sportevent="http://smartweb.semanticweb.org/ontology/sportevent#"
 xmlns:foaf=
 "http://xmlns.com/foaf/spec/20071002.rdf"
 xmlns:contact=
 "http://www.w3.org/2001/vcard-rdf/3.0#"
>
```

Zooming in on just the event itself, we can add some contestants:

```
<div rel="cal:Vevent">
  <span property="cal:dtstart" content="20080222T1400-0700">2:00PM</span>
  :
  <span property="cal:summary">Bull Riding</span>

  <ul>List of Contestants:
  <li rel="sportevent:Contestant" id="Marchi">
    <span property="foaf:name" about="#Marchi"
      >Guilherme Marchi</span><br/>
```



```
    <a rel="foaf:weblog" about="#Marchi"
      href="http://example.com/~Marchi"
    >Marchi's blog</a>
  </li>
  <li rel="sportevent:Contestant" id="Briscoe">
     <span property="foaf:name" about="#Briscoe">Travis Briscoe</span>
  </li>
  </ul>
</div>
```

And, at the bottom of the page, we add a footer with general contact information:

```
<p class="footer" about="/main/page/for/Rodeo">
  For general information or event questions, please call
  <span property="contact:phone">800-555-1212</phone>
  or email
  <a rel="contact:email" href="mailto:rodeo-info@example.com"
  >rodeo-info@example.com</a>
</p>
```

RDFa uses several existing HTML properties and creates a few new ones. Recall that an RDF statement has three parts: subject, predicate and object. The about= or instanceOf= property of a tag can specify the subject. The rel=, rev= or property= property specifies the predicate. Then, the object may be the href=, content= or actual content enclosed by the tag pair. Note that the subject may be in a parent tag and, if missing, defaults to the document itself. Refer to the RDFa Syntax Specification and Primer documents for a detailed explanation of all the ways that RDF can be embedded in HTML.

Re-verifying through the RDFa Distiller returns the necessary @prefix lines to specify the vocabularies, followed by the N3:

```
@prefix cal: <http://www.w3.org/2002/12/cal/ical#>
(...all the other prefixes...)

<http://abra.info/lj/rodeo.xhtml> cal:Vevent
 [ sportevent:Contestant
    <http://abra.info/lj/rodeo.xhtml#Briscoe>,
    <http://abra.info/lj/rodeo.xhtml#Marchi>;

    cal:dtstart "20080222T1400-0700";
    cal:summary "Bull Riding"
 ].

 <http://abra.info/main/page/for/Rodeo>
    contact:email <mailto:rodeo-info@example.com>;
    contact:phone "800-555-1212".

 <http://abra.info/lj/rodeo.xhtml#Briscoe>
  foaf:name "Travis Briscoe".

 <http://abra.info/lj/rodeo.xhtml#Marchi>
  foaf:name "Guilherme Marchi";
  foaf:weblog <http://example.com/~Marchi>.
```


Advertiser Index

For advertising information, please contact our sales department at 1-713-344-1956 ext. 2 or ads@linuxjournal.com. www.linuxjournal.com/advertising

Advertiser	Page #
1&1 Internet Inc. www.oneandone.com	6, 7
Aberdeen, LLC www.aberdeeninc.com	29
ASA Computers www.asacomputers.com	71
Avocent Corporation www.avocent.com	1
Black Hat Briefings (Confex Partners LTD) www.blackhat.com	69
Cari.net www.cari.net	47
Coraid, Inc. www.coraid.com	5
Emac, Inc. www.emacinc.com	77
EmperorLinux www.emperorlinux.com	19
Genstor Systems, Inc. www.genstor.com	79
HPC Systems, Inc. www.hpcsystems.com	C2
Hurricane Electric www.he.net	27
IDG World Expo www.idgworldexpo.com	73
Linux Plumbers Conference linuxplumbersconf.org	85
Logic Supply, Inc. www.logicsupply.com	65
LPI www.lpi.org	95
Microway, Inc. www.microway.com	C4, 59
Mikro Tik www.routerboard.com	3
Ontario Linux Fest onlinux.ca	75
Opengear www.opengear.com	17
O'Reilly OSCON conferences.oreilly.com/oscon	87
O'Reilly Ubuntu Live www.ubuntulive.com	67
The Portland Group www.pgroup.com	21
R1Soft, Inc. www.r1soft.com	49
Rackspace Managed Hosting www.rackspace.com	C3
Servers Direct www.serversdirect.com	9
Silicon Mechanics www.siliconmechanics.com	11, 33
Technologic Systems www.embeddedx86.com	91
USENIX Security Symposium www.usenix.org/security08/lja	83
VersaLogic Corporation www.versalogic.com	89
ZT Group International www.ztgroup.com	43

It's just like that. Well, that's not exactly how it went. The RDFa Distiller fails tersely on less-than-valid XHTML, which means that one mismatched tag or missing quotation mark causes unexplained failure. So, what I really did was use the user-friendly W3 Validator service first, at **validator.w3.org**, which reminded me about some missing tags and also to save my example as .xhtml so it would be returned with the correct MIME type. After passing the validator, I renamed the file and ran it back through the RDFa Distiller to generate the above N3 output. (The Distiller also has some caching issues. It was designed as a check of the syntax specification, not as a user tool. I use it anyway because I like the N3 output format.)

Another useful tool for checking your triple logic is the GetN3 bookmarklet available from **www.w3.org/2006/07/SWD/RDFa/impl/js**. Once you've saved it as a bookmark, you can use it to extract the RDFa quickly as N3 of any page you have in the browser. It's also more forgiving than the Distiller, so you can use it as a quick logic check without worrying about valid XHTML.

Publishing Tools

Tools are emerging from the RDF world at an accelerating pace this year, and you may find what you need without writing a line of code. Not all of them produce RDFa, however. Some, such as Semantic MediaWiki, produce the HTML and RDF side by side, from an internal triple store. It's a fair chance you've already used RSS (which originally stood for RDF Site Summary when it was created at Netscape back in 1999). If you use version 1.0, take a look at the RSS source—it's valid RDF/XML.

Another group to keep an eye on is the Simile Project at MIT. It has an interesting range of tools with the broad purpose of managing and reusing bits of digital data. Not all are RDF-related, but the RDFizer promises to convert a variety of structured formats to RDF for you: mbox, Debian software packages, Subversion and many more.

The most advanced tools probably are not yet in the open-source arena. Metatomix, Inc., has done some heavy lifting in the semantic application field, with major implementations in engineering, finance and integrated justice. I talked to CTO Howard Greenblatt, and he explained the company's technology stack. The key components are first, a set of development tools for creating the ontology, and second, a messaging platform that gathers data from traditional data sources and integrates it into a triple store, along with some business rules logic. For the first component, they have their own plugin for the Eclipse development environment, and for the second, they use Jena from HP Labs plus a bunch of proprietary code. Then, the whole thing can be queried in SPARQL, the query language of the Semantic Web.

That's more than most Web developers are likely to bite off. However, it brings us back to a point from our example above: choosing, or creating, an appropriate vocabulary. To say anything on the Semantic Web, you have to have a namespace in which to speak precisely. Writing your own vocabulary is not too hard (and Semantic MediaWiki helps you do it automatically), but you may want to choose a standard one, at least if you are interested in search engine discovery.

Search Engines and Vocabularies

Yahoo announced in March 2008 that it would start supporting Semantic Web standards, including microformats, RDFa and eRDF. And, it announced specific vocabulary components that would be supported: Dublin Core, Creative Commons, FOAF, GeoRSS and MediaRSS. Using these vocabularies will make your data more portable and easier for search engines to index intelligently.

If you want to see what vocabularies others are using, the GetN3 bookmarklet is helpful. A visit to **digg.com**, run through the GetN3 bookmarklet, shows that Digg is now embedding RDFa using the Dublin Core and Creative Commons vocabularies (prefixes added):

```
<http://digg.com/>
 cc:attributionName "Digg users";
 cc:license cclicense:publicdomain/;
...
<http://digg.com/space/Jules_Verne_in_Orbit>
 dc:source <http://apod.nasa.gov...>;
 dc:title "Jules Verne in Orbit";
 dc:abstract "The bright edge of planet Earth..";
 dc:creator <http://digg.com/users/ezentmyer>;
 dc:date "2008-04-05 05:07:38";
```

I think the Semantic Web is finally taking off this year. Semantic applications range from personal desktop productivity (MIT's Piggybank) to new Web search engines (Yahoo) to huge enterprise applications and even military information-sharing. As the social networks grow heavy with data, sharing and structuring that data becomes more important. Eric Miller, an MIT professor who led the Semantic Web initiative for the W3C, sees "a new market space for data aggregation, data integration, and data discovery". And, all you have to do to be a part of that space is add a couple tags into your page.

Acknowledgements

Thanks to the following individuals for their time, assistance and insight: Michael Hausenblas, Ivan Herman, Eric Miller, Howard Greenblatt, Duane Degler, Marwan Sabbouh and Michel Biezunski.■

Golda Velez (goldavelez.info) is a developer, consultant and freelance writer focusing on emerging technologies. She works on developing vocabularies for specialized domains and partnering with domain experts to create vertical sites. She lives with her husband and children in Tucson, Arizona.

Resources

Related links are collected at **abra.info/lj/rdfa.xhtml**.

The example in this article is available at **abra.info/lj/rodeo.xhtml**.

17th USENIX SECURITY SYMPOSIUM

SAN JOSE, CA JULY 28–AUGUST 1, 2008

A five-day tutorial and refereed technical program for security professionals, system and network administrators, and researchers.

Co-Located Workshops

EVT '08
2008 USENIX/ACCURATE Electronic Voting Technology Workshop, July 28–29, 2008

CSET '08
Workshop on Cyber Security Experimentation and Test, July 28, 2008

WOOT '08
2nd USENIX Workshop on Offensive Technologies, July 28, 2008

HotSec '08
3rd USENIX Workshop on Hot Topics in Security, July 29, 2008

MetriCon 3.0
Third Workshop on Security Metrics, July 29, 2008

2 Days of In-Depth Tutorials Taught by Industry Experts, Including:

- Bruce Potter on Botnets: Understanding and Defense
- Simson L. Garfinkel on Computer Forensics

Keynote Address

"Dr. Strangevote or: How I Learned to Stop Worrying and Love the Paper Ballot," by Debra Bowen, *California Secretary of State*

Technical Program

27 refereed papers presenting the best new research in a variety of subject areas, including voting and trusted systems, privacy, and botnet detection

Invited Talks by Experts, Including:

- "Hackernomics," by Hugh Thompson, *Chief Security Strategist, People Security*
- "Political DDoS: Estonia and Beyond," by Jose Nazario, *Senior Security Engineer, Arbor Networks*
- "The Ghost in the Browser and Other Frightening Stories About Web Malware," by Niels Provos, *Google, Inc.*

SAVE!

Register by July 14, 2008, and save!

www.usenix.org/security08/lja

My Move to Solid State

If you are dissatisfied with your laptop hard drive's performance, you should consider moving to a new state—solid state. Read on for head-to-head comparisons between a standard laptop hard drive and a solid state model. KYLE RANKIN

I *love* small laptops. If you ask any of my friends, they will tell you that even a laptop with a 12" screen, no matter how cool, is just too big for me. My very first laptop was a Toshiba Libretto 50CT, which was around the size of a VHS tape (for those of you who remember those), and from there, I have progressed through the Fujitsu P series with a P2110, P7010 and now a P1610—an 8.9" ultra-portable tablet. I use this laptop as my primary machine with few complaints, but when I made the jump to a tiny 8.9" tablet from my old 10.6" sub-notebook, I also had to drop from a 2.5" to a 1.8" hard drive.

For me, especially at first, a 1.8" hard drive wasn't the end of the world. Even though I had upgraded to 5400rpm drives on my other laptops, to me, the decrease in size for the overall laptop was worth any drop in performance. Plus, until recently, it wasn't like I had much choice: 1.8" drives maxed out at 4200rpm. Then I heard about solid state drives. Unlike a traditional hard drive that relies on a head and a spinning platter, solid state drives act more like Flash storage you might use in your camera or on a USB key. Not only are there no moving parts to wear out and much faster seek times, the 1.8" solid state drives I saw touted faster sustained read and write times as well.

Although I have read a number of benchmarks and anecdotes about solid state drives, it always seemed like a mixed bag. Windows users talked about much faster startup times and better overall responsiveness, while the Mac reviews I read seemed to indicate the difference in performance was minimal. I didn't see too many benchmarks about Linux systems, and with the high price tag of solid state drives, I went back and forth on which price point I was willing to pay.

One day I decided to take the plunge and bought a 1.8" Samsung solid state drive for my laptop. In the process, I have taken some comparison benchmarks between my old drive and my new solid state drive. Although statistics can be handy, I decided to take a more tangible approach to my comparisons. I used some standard benchmark tools, but the majority of my comparisons deal with everyday tasks to give you a better idea of what it's really like to have a solid state drive on a Linux system.

The Testing Methodology

First, I should tell you what hardware is being compared. All tests were run on my trusty Fujitsu P1610. It has an Intel 1.2GHz ULV Core Solo processor with 1GB of RAM and is running Ubuntu 7.10. The original hard drive was a 4200rpm Toshiba MK6006GAH, and I am comparing it to a Samsung MCBOE32G8APR solid state drive. When reasonable, I tried to run tests multiple times so I could get an average reading; however, just so you know, most of the tests ended up being pretty consistent between tries. Also, when necessary I rebooted the machine before performing follow-up tests so that any files Linux might have cached into RAM would not affect the results.

Test 1: GRUB to Login

For the first test, I used a stopwatch to time how long it took the system to go from the GRUB boot prompt to my login screen. Depending on how you use your laptop, you may boot it every day, or you may hibernate or suspend between uses. In either case, a slow boot time can be painful when you want to get right to work. The boot process is both disk- and processor-intensive, but even so, when comparing the results, you'll see a significant difference:

- 4200rpm: 50 seconds
- SSD: 34 seconds

Test 2: Login to Desktop

The next logical test is the time it takes from your login to a usable desktop. For my laptop, I use the default desktop environment that comes with Ubuntu (GNOME), but I also have terminals, applets and Firefox all launching at startup. As a result, my numbers might differ a bit from yours, but they give a good sense of the difference between the two drives:

- 4200rpm: 59 seconds
- SSD: 23 seconds

Wow. Although I knew to a degree that it took some time for my desktop to come up with the old hard drive, I didn't realize until this test that it actually took almost an entire minute! By comparison, the SSD took less than half the time, in part due to the increased read speed and the much faster seek times, especially when loading files at random (see the bonnie++ test below to corroborate this). So far, the SSD is looking pretty good. If you combine both tests, the 4200rpm drive took 109 seconds—almost two minutes—to go from the GRUB prompt to a usable desktop, and the SSD took 57 seconds—almost half the time.

Test 3: Untar the Kernel

For the next test, I decided to time how long it took to extract the 2.6.22 kernel bzipped tarball. Now, because this tarball is

bzipped, a good deal of the stress on the system will be on the CPU, not the disk. However, because most tarballs are compressed, and it is a pretty common desktop activity, I thought it was still worth comparing. The results weren't nearly as dramatic as the first two tests (due to the activity being mostly CPU-bound), but the SSD still beats the 4200rpm drive by 13 seconds:

- 4200rpm: 66 seconds
- SSD: 53 seconds

Test 4: Suspend to Disk

Many laptop users (myself included) rarely boot and shut down their systems between uses. Instead, they rely on the hibernation and suspend features to save their current state and resume to it quickly. With hibernation, the laptop writes its current state to disk and powers off. With suspend, the laptop keeps its current state in RAM and stays on in a low-power state. Because the hibernation process is so disk-heavy, I decided it would be a good way to test whether an SSD gave any speed benefit. So, for the first test, I measured the time from enabling hibernation until the system powered off:

- 4200rpm: 75 seconds
- SSD: 50 seconds

Again, before I saw the numbers, I didn't realize it had taken more than one minute 15 seconds to shut down and preserve my 1GB of RAM. Although the SSD still took some time, it beat the old drive by 25 seconds.

Test 5: GRUB to Resume

The follow-up to my hibernate test was to resume from the hibernation state. I started the clock once I pressed Enter at the GRUB prompt and stopped it once I got to the login window for my locked screen:

- 4200rpm: 83 seconds
- SSD: 38 seconds

This result really surprised me. The SSD fared better than the 4200rpm drive when suspending to disk, but it was more than twice as fast when resuming! When you compare the combined tests, the 4200rpm drive takes 158 seconds to suspend and resume, and the SSD shortens the process down to 88 seconds.

Test 6: Traditional Benchmarks with hdparm and bonnie++

Even though the everyday benchmarks were enough to convince me of the speed benefit of an SSD, I knew a lot of you also would want some raw data to compare. So, I also ran hdparm and bonnie++ on both drives with some interesting results. First, I ran hdparm three times in a row:

Please Join us in
Portland, Oregon
September 17-19, 2008
For the 1st Annual

Please visit:

http://linuxplumbersconf.org

for additional information and registration.

4200rpm:

```
/dev/sda3:
 Timing cached reads: 1842 MB in 2.00 seconds = 921.90 MB/sec
 Timing buffered disk reads: 64 MB in 3.08 seconds = 20.79 MB/sec
/dev/sda3:
 Timing cached reads: 1814 MB in 2.00 seconds = 907.56 MB/sec
 Timing buffered disk reads: 64 MB in 3.08 seconds = 20.78 MB/sec
/dev/sda3:
 Timing cached reads: 1794 MB in 2.00 seconds = 897.43 MB/sec
 Timing buffered disk reads: 62 MB in 3.04 seconds = 20.39 MB/sec
```

SSD:

```
/dev/sda:
 Timing cached reads: 1894 MB in 2.00 seconds = 947.80 MB/sec
 Timing buffered disk reads: 80 MB in 3.07 seconds = 26.02 MB/sec
/dev/sda:
 Timing cached reads: 1894 MB in 2.00 seconds = 947.61 MB/sec
 Timing buffered disk reads: 80 MB in 3.08 seconds = 26.00 MB/sec
/dev/sda:
 Timing cached reads: 1886 MB in 2.00 seconds = 943.86 MB/sec
 Timing buffered disk reads: 78 MB in 3.00 seconds = 25.99 MB/sec
```

As you can see, the SSD certainly is faster; however, there is not nearly as large a margin as with some of the other tests. The bonnie++ results show a different story:

4200rpm:

```
------Sequential Output------ --Sequential Input- --Random-
 -Per Chr- --Block-- -Rewrite- -Per Chr- --Block-- --Seeks--
Size K/sec %CP K/sec %CP K/sec %CP K/sec %CP K/sec %CP  /sec %CP
2G 11309  52 11272   3  4921   2 10715  44 11471   2  83.8   0
 ------Sequential Create------ --------Random Create--------
 -Create-- --Read--- -Delete-- -Create-- --Read--- -Delete--
files  /sec %CP  /sec %CP  /sec %CP  /sec %CP  /sec %CP  /sec %CP
 16   190   2 +++++ +++   177   1   196   2 +++++ +++   154   1
minimus,2G,11309,52,11272,3,4921,2,10715,44,11471,2,83.8,0,16,190,
➥2,+++++,+++,177,1,196,2,+++++,+++,154,1
```

SSD:

```
------Sequential Output------ --Sequential Input- --Random-
 -Per Chr- --Block-- -Rewrite- -Per Chr- --Block-- --Seeks--
Size K/sec %CP K/sec %CP K/sec %CP K/sec %CP K/sec %CP  /sec %CP
2G 18155  94 23125   8 12521   8 20818  94 28149   8  1226   5
 ------Sequential Create------ --------Random Create--------
 -Create-- --Read--- -Delete-- -Create-- --Read--- -Delete--
files  /sec %CP  /sec %CP  /sec %CP  /sec %CP  /sec %CP  /sec %CP
16  1128  11 +++++ +++  1101  10  1158  10 +++++ +++   449   4
minimus,2G,18155,94,23125,8,12521,8,20818,94,28149,8,1226.4,5,16,
➥1128,11,+++++,+++,1101,10,1158,10,+++++,+++,449,4
```

Well, that's certainly a lot of data. A few numbers do stand out though. On sequential output and input, the SSD's performance is almost twice that of the 4200rpm drive, except in random seeks where it is actually 14 times faster with 1,226 seeks per second! Because there is no spinning platter, random seeks are one area where a solid state drive really shines. The next level of stats compares the speed of creating files on the system sequentially and at random. It is here that we see another huge advantage for the SSD, as it is five times faster at sequential creates, six times faster at sequential deletes and almost six times faster at random creates.

Is It Worth It?

All of these numbers aside, the question you might be asking is, "Is it worth it?" For me, the answer is a definite yes. Not only is my system back to the snappiness I remember with past laptops, it also no longer seems to get bogged down during disk-heavy operations like when my backup software kicks off. That reminds me of another point—noise. With no moving parts, the SSD is basically silent. The only noise on my laptop now is from the fan. The other day I was using my laptop and noticed that the fan had gone to almost top speed. After some time, I decided to check the system temperature to see whether my laptop was really that hot. It turned out that my network backup job had kicked off and was rsyncing. This is a CPU- and disk-heavy operation, and with my old drive, I instantly would know when it kicked off, because the system would slow down, and I would hear the all-too-familiar clicking and clacking of my hard drive. Now, due to the snappiness of the desktop and silence of the SSD, I wasn't even aware the backup was happening.

Whether the performance of an SSD is worth it to you depends on a variety of things. If you are stuck with a 1.8" 4200rpm drive like I was, there aren't too many other options for you (although a 5400rpm 1.8" drive should be available for purchase soon), but if you have a larger drive with up to 7200rpm spindle speeds and SATA interfaces, you definitely will want to compare the posted speeds of comparable drives—it's possible that the current generation of SSDs won't offer you many speed benefits. There also are other factors to consider, including the potential power savings some SSDs offer. Plus, with the lack of moving parts, you not only get a quieter system, you also potentially get a more durable one. On the downside, even with write-leveling technologies, there still are a finite number of writes you can make to an SSD, although most manufacturers claim that the life of an SSD still exceeds that of traditional drives.

If you do decide to get a solid state drive, be sure to do your homework. There are a number of different laptop hard drive interfaces these days, so if you have a 1.8" drive, be sure to check whether you use a ZIF or non-ZIF connector. And, if you want to use a 1.8" SSD in your 2.5" laptop, be sure that a compatible adapter exists (I've seen some sellers include adapters as a package deal).■

Kyle Rankin is a Senior Systems Administrator in the San Francisco Bay Area and the author of a number of books, including *Knoppix Hacks* and *Ubuntu Hacks* for O'Reilly Media. He is currently the president of the North Bay Linux Users' Group.

O'REILLY®

OSCON™

Open Source Convention

conferences.oreilly.com/oscon

July 21 – 25 , 2008
Oregon Convention Center
Portland, Oregon

OSCON brings you face to face with the bright minds, bold ideas, and open technologies that are powering some of the largest companies and most innovative projects in the world. Explore today's issues and tomorrow's solutions with over 2,500 other open source professionals and influencers at the tenth anniversary of OSCON.

Register now and save 15%
Use discount code os08ljr

OSCON 2008 features over 40 tutorials and 400 sessions that dive deep into:

- **Administration**
- **Business**
- **Databases**
- **Emerging Topics**
- **Fundamentals**
- **Java**
- **Linux**
- **Mobile**
- **People**
- **Perl**
- **PHP**
- **Programming**
- **Python**
- **Ruby**
- **Web Applications**
- **and much more**

©2008 O'Reilly Media, Inc. O'Reilly logo is a registered trademark of O'Reilly Media, Inc. All other trademarks are property of their respective owners. 80194ljr

Linux and the Enterprise Desktop: Where Are We Today?

Desktop Linux is maturing, and many organizations are taking notice. JAMES GRAY

Linux making its way out of the server room and onto the desktop has been "just around the corner" for years now. Prognostications of desktop dominance have not materialized, leaving Linux with a market share in the low single digits. Nevertheless, Linux is maturing as a desktop platform for the enterprise and is gaining converts, with a growing number of companies leveraging Linux to get more features for their money. In this article, I take a closer look at the latest trends in desktop Linux in the enterprise, as well as a number of case studies that illustrate how Linux is fully ready to be a robust desktop platform in many situations.

Big Trends

From talking with several people in the industry who promote desktop Linux to the enterprise for a living, my overall impression is that the Linux desktop wave is indeed building. Although interesting and significant implementations exist, more large-scale projects are in the pipeline than have emerged from it. The people I spoke with pointed to trends, but they generally could back them up with only a single example or weren't able to mention the client's name.

Nevertheless, forward movement is occurring for Linux on the enterprise desktop, and the people on the front lines are bullish. For instance, Mindy Anderson, Business Manager for Client Strategies at Red Hat, states that "the desktop is working itself into being disruptive in many industries, including finance, telecommunications and health care". Meanwhile, Gerry Carr, Marketing Manager at Canonical (commercial supporter of Ubuntu), adds that "we find ourselves at the beginning of a bell curve, where only a minority of potential clients have deployed, and we're engaged in talks with the people who are in the big hump of the curve". Over at Novell, Guy Lunardi, Senior Product Manager of SUSE Linux Enterprise Desktop, asserts that "we're there from a technological standpoint", and the critical factor that prevents Linux from going gangbusters, says co-Novellite Michael Applebaum, Product Marketing Manager for the Desktop, is "simply the awareness that desktop Linux is already a very viable platform".

Barrier Removal

Despite bullishness on the part of Linux vendors, these same companies admit that they remain in barrier-removal mode. Canonical's Carr admits that his sales staff continue to confront objections, such as lack of equivalent Linux-based applications, which often can be resolved conveniently with solutions like virtualization, but sometimes, they can't. Novell's Applebaum notes that his firm must further improve on the interoperability of all ecosystem elements to make them easier to manage, as well as expand hardware and software certification so that customers can acquire complete, preloaded desktop solutions. Although the larger distribution providers, such as Red Hat, Canonical and Novell, have collaborated with Lenovo, Dell and others to preload and certify PCs for Linux, the reality is that the hardware vendors offer fewer options and lack the same hard-sell enthusiasm to hawk Linux. Even today, you can buy a PC from the Lenovo or Dell's on-line stores and never realize that Linux is available.

It also is true that Linux providers at last can say that the OS is intuitive enough for typical office workers who are accustomed to using Windows. This has not always been the case. Novell should be commended for its Better Desktop initiative, which applied a scientific methodology and video capture to examine how real people use Linux, discover its pitfalls and see how its deficiencies can be removed. The investigators captured more than 200 videos of people using Linux and its core applications for everyday tasks. For instance, normal users were examined while doing everything that is fully routine to us geeks—logging on to their system, finding and playing a particular music track, making shortcuts on the desktop, determining available disk space, sending e-mail and more. The reports and videos are fascinating and available on the project's Web site.

Drivers to Adoption of Desktop Linux

Several IT trends are making desktop Linux more attractive to many organizations. One of these is a growing desire to reduce licensing costs. Novell's Lunardi notes how its customer, the automaker Peugeot, decided to cap its number of Microsoft licenses as its workforce grows and offer Linux desktops to new employees. Another trend is the push toward accommodating more types of devices, including mobile and thin clients, as well as allowing users to take their desktops with them wherever they go. Red Hat's Anderson says that "many firms are coming back to a situation where key workloads are centralized", something that Linux does very well and securely. Similarly, Novell's Applebaum says that San Diego Public Schools chose Linux over other operating systems because it offered the most robust way to run its "Always-On Learning Initiative", which included integrating 100,000 student laptops and many other types of devices.

A third trend involves avoiding Windows Vista drawbacks, especially the cost of required hardware upgrades and lack of additional features to justify that cost. Linux, with its smaller footprint,

may find a great deal of growth opportunity from this situation.

Where Linux Is Ready

The distribution providers are finding several types of workers ready for desktop Linux. Novell, for example, classifies its target customers this way:

- Office workers—marketing associates, office managers, operations managers and insurance agents who rely on a robust desktop or notebook platform for e-mail, Web browsing, instant messaging, multimedia applications and office productivity tools. Linux offers cost savings on hardware requirements, as well as for the OS and applications.

- Transactional workers—bank sales/service representatives, call-center representatives and other service personnel who spend most of their time using a few specialized business applications, but who also require collaboration applications, such as an e-mail client and Web browser, and productivity applications, such as a word processor or spreadsheet.

- Thin-client workers—companies with mobile workers in multiple locations who want to keep data consolidated are ideal for thin-client solutions. Unneeded software and hardware costs can be removed from the budget. Users can be provided with the right applications when and where they need them.

- POS workers—front-line sales and service workers who deal with customers that need a reliable and user-friendly technology for enabling transactions. These solutions also could be self-service kiosks. Linux offers advantages, such as reliability, convenient security patches and immunity to viruses.

- Technical workstation—the creative and analytical workers in an organization who design products, create film animation, run mathematical models and so on. In these situations, Linux offers the same reliability as UNIX at a fraction of the cost.

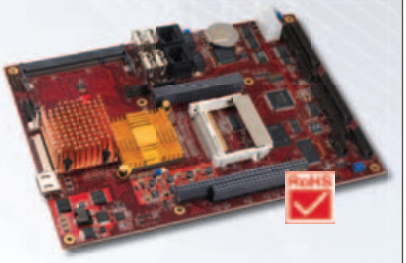
Early Adopters

It's safe to say that the companies already deploying desktop Linux are early adopters. Gerry Carr of Canonical says that those entities who already have moved to Linux tend to be smaller organizations "that offer their savvy technical guys lots of autonomy". Another large share of early-adopter organizations are public entities, which have a strong mandate for cost-cutting. Recall the fanfare back in 2003 when the city of Munich, Germany, snubbed Microsoft in favor of deploying 14,000 Linux desktops. Since then, public entities of all sizes have leveraged Linux for public benefit. For instance, Canonical's most highly touted implementations range from 150,000 Linux desktops in the Macedonian public schools and 70,000 in the French National Police, down to 300 seats in the Howard County (Maryland) Public Library system.

Though many large private companies also have adopted desktop Linux, more case studies are available overseas than here in the US. The distribution providers say this is because US-based firms are more secretive, as they see Linux as a comparative advantage. Red Hat's Anderson told me "I could rattle off names, but you can't print them". And, although other spokespeople also offered company names off the record, they remained much less numerous than the overseas examples. Some larger organizations who openly use desktop Linux include the French automaker Peugeot (20,000 clients), the Australian affiliate of Europcar automobile rental, the American firm ECI Telecom (3,000 clients), Taiwan's Realtek Semiconductor (2,000 clients) and the German insurance company LVM Versicherungen (8,500 clients).

Arguably the most mature sector for desktop Linux is the corporate workstation, where the dynamics of the game are a bit different. Although the absolute number of Linux workstations is not large in real terms, Linux's share is substantial. HP's David Ramsey, Product Marketing Manager for Linux workstation software, notes that Linux is well suited to the task-oriented nature of workstations. Companies with specialized tasks, such as modeling, forecasting and

The beauty of high performance.



The Sidewinder Single Board Computer (SBC) features a host of integrated I/O in a compact, RoHS compliant package. Sidewinder's VIA® Eden 1.2 GHz processor enables power-efficient performance, while its EBX platform hosts a PC/104-*Plus* PCI expansion site for adding critical capabilities without expanding the board footprint. Sidewinder offers multi-function support for larger system requirements without compromising computing power and design flexibility. Its elegant fanless design brings high per-watt performance, on-board Analog and Digital I/O, CompactFlash® socket, SPX™ interface and an overall outstanding cost of ownership. Customization is available on quantities as low as 100 pieces.

Contact us and discover how for more than 30 years we've been perfecting the fine art of extraordinary support and on-time delivery: One customer at a time.

1.800.824.3163
1.541.485.8575

VersaLogic Corporation
www.versalogic.com/sid

animations, which historically have been housed on UNIX workstations, gradually have migrated to the Linux platform. Ramsey says that HP and others have been especially successful in industries, such as oil and gas (geological modeling); finance (real-time data processing); animation studios, such as DreamWorks and Pixar; and government labs. Mechanical CAD, electronic design automation and governmental applications in national security are up-and-coming applications for Linux workstations. Ramsey also added that expansion of this space often is contingent on application providers porting their products to Linux, as well as the cost of migration. "The reward must be substantial to do it", he said.

Case Studies of Desktop Linux in the Enterprise

Replete with context on desktop Linux in the enterprise, let's explore some interesting, representative case studies. Most case studies here were emphasized by distribution and solution providers as their most interesting projects. The implementations are in entities of various sizes in both the private and public sectors.

PSA Peugeot Citroen: Let the User Decide

PSA Peugeot Citroen, Europe's second-largest automobile manufacturer, will one day have at least 20,000 of its 72,000 workers running Novell's SUSE Linux Enterprise Desktop. Novell's Guy Lunardi credits Peugeot's move to Linux on its hard-nosed asset management system that identifies internal resources, leading its IT managers to question conventional wisdom and make the best strategic decision for the company. Lunardi also points out that "Novell was able to not only remove technical barriers that stood in the way of adoption, but it also was able to offer superior solutions in areas such as VPNs."

One unique facet of the Peugeot project is its organic nature, whereby users are allowed to choose which desktop they prefer, Linux or Windows. The firm is finding most users choose Linux and become Linux advocates in the process, which further builds internal support for the OS. Furthermore, most new employees are encouraged to adopt Linux, which has allowed Peugeot to cap its number of Windows licenses and thus save on IT costs, despite the additional growth in IT infrastructure.

Europcar: Save Money and Still Offer Windows Apps

Another convert to desktop Linux, in this case Red Hat Enterprise Linux Desktop, is the auto-rental firm Europcar Asia Pacific. Europcar CIO Scott Allen explains how he originally looked to desktop Linux "to deliver a fit-for-purpose platform at a reduced price point". Allen added that this goal has been achieved through significant reduction in licensing costs and the ability to extend the life of otherwise obsolete hardware. Europcar has implemented Linux on a variety of different hardware in its finance department, national call center and branch network, which spreads across Australia and New Zealand. Allen's team customized the Red Hat desktop to deliver a simplified interface, offering many Microsoft Windows applications to users via the Citrix client, as well as Web browsers natively to the Linux desktop. Using Citrix "makes the solution an overall lower cost option but still a great fit against business needs", said Allen.

A key factor in Europcar's expanding its Linux usage to the desktop was quality support. Because the firm had good, long-term experiences with Red Hat's support for its servers, it felt comfortable diving into the desktop space too.

Thus far, CIO Allen says, his firm's experiences with desktop Linux have been very positive. "The work done to deliver a customized user interface was worthwhile and has meant that very little end-user training has been required." He adds that "the only negative—and it is only a minor one because of our relationship with Red Hat—is the availability of Linux skills in the market. At times it has been difficult to recruit people with in-depth Linux knowledge."

Allen's advice to firms trying to decide whether to adopt desktop Linux is to "start with the needs of the end user and evaluate Linux desktop against these needs". He says if users require Windows applications 100% of the time, Linux desktop probably is not the best solution, even when using something like Citrix. "But if your business applications do not rely on Windows or your users only require part-time access to Windows applications, I would at least include Linux desktop as an option to evaluate", he added.

Howard County Library: Linux Saves Public Money

Though the Howard County Maryland Public Library is not the largest or most sexy desktop Linux implementation, it was the first example to proudly roll off the tongue of Canonical's Gerry Carr. In 2006, the library had 300 aging PCs whose licenses for Windows XP were about to expire. Realizing it didn't need Windows for its staff and public computers, it became the first public library system in Maryland to use open source. When additional machines are needed, the library can purchase used ones for around $100 that offer the needed functionality. The system estimates that it saved more than $300,000 of public money by not having to upgrade licenses and hardware. Monies saved were used to upgrade computers, purchase software customization and expand library collections. Amy Begg De Groff, the library's Technology Services Department Head, stated that "Because open-source software is available free or at a very modest cost, the library can provide public computers at a fraction of the cost using comparable commercially available software."

Mosaic: Savings and Security

As an example from the not-for-profit world, the organization Mosaic is in the process of implementing Ubuntu Linux thin clients running on the NoMachine NX Server for its 39 offices in 14 states nationwide. At the time of this writing, 13 offices serving 1,900 users have been migrated, with the remaining offices to migrate by 2009. More than 5,000 employees are projected to be using the system. Users have access to a full Ubuntu desktop, which offers OpenOffice.org, Firefox, Mosaic's corporate CRM and Web-based e-mail applications. Mosaic's two main reasons for switching to Linux were dramatic

cost-savings and security. The organization has re-utilized older PCs to run as its thin-client terminals, which precluded the need for large investments in new hardware. Regarding security, Mosaic is able to keep sensitive data centralized in its data center to ensure that all data access is controlled and monitored to meet HIPPA regulations. Wayne Victor, Mosaic's Director of IT Infrastructure, said that "the NX Server is able to handle as many as 100 users per server."

San Diego Schools: Addressing the Digital Divide

At 135,000 students, the San Diego Unified School District (SDUSD) is the country's eighth largest. During each school day, around 100,000 of those students in grades 3–12 use a Lenovo R-Series ThinkPad laptop running SUSE Linux Enterprise Desktop provided by the district as a learning tool. Dubbed the "Always-On Learning Initiative", the purpose of the program is to promote academic success by giving all students access to the tools they'll need to learn, live and work successfully in the modern world. The name Always-On comes from the fact that students have access to a wireless network anywhere and at all times throughout the district.

Choosing Linux over other operating systems offers several advantages to SDUSD. First, the lower cost of Linux allows SDUSD to reach more students with fewer resources. By reaching more students, the chronic problem of a digital divide between wealthier and poorer students can be addressed. Second, after comparing its options, SDUSD determined that Linux was easier to scale and support, and more types of devices could be utilized. Finally, SDUSD sees that providing a laptop to everyone has great motivational and learning benefits for both teachers and students. Deputy Superintendent Geno Flores said that "students are more interested in and excited about their classroom work" thanks to the program.

Early Adopters Move Linux Forward

The small sample of case studies above illustrate how Linux is fully ready to start taking over more desktops in companies, nonprofits and government offices worldwide. Although we long have been optimistic that Linux's day of glory would come sooner, our 20/20 hindsight allows us to grasp that usability and features had to be improved to meet the needs of most workers. Fortunately, the distribution providers have realized this fact and invested heavily in removing barriers to Linux implementation. Certainly, the maturation of virtualization has helped as well. In addition, the early adopters, whose stories are told here, have helped all of us by implementing Linux despite some unknowns and moved it forward. Now, more-conservative organizations can observe these examples and learn from their experiences, both positive and negative. So our thanks go out to Peugeot, Europcar, Howard County Library, Mosaic, San Diego Schools and the other desktop Linux pioneers for implementing Linux on a large scale and helping make it better. In a few years, you should be able to look back and be amazed at what you started.■

James Gray is *Linux Journal* Products Editor and a graduate student in environmental science and management at Michigan State University. A Linux enthusiast since the mid-1990s, he currently resides in Lansing, Michigan, with his wife and cats.

Resources

Canonical: **www.canonical.com**

Linux at HP: **www.hp.com/linux**

NoMachine NX Server: **www.nomachine.com**

Novell's Better Desktop Initiative: **www.betterdesktop.org**

Red Hat Enterprise Linux Desktop: **www.redhat.com/rhel/desktop**

SUSE Linux Enterprise Desktop: **www.novell.com/products/desktop**

Ubuntu Desktop Edition: **www.ubuntu.com/products/whatisubuntu/desktopedition**

TS-7800 High-End Performance with Embedded Ruggedness

$269 qty 1 $229 qty 100

500 MHz ARM9

- New unbrickable design- 3x faster
- Backward compatible w/ TS-72xx
- Low power - 4W at 5V
- 128MB DDR RAM
- 512MB high-speed onboard Flash
- 12K LUT user-programmable FPGA
- Internal PCI Bus, PC/104 connector
- 2 USB 2.0 480 Mbps
- Gigabit ethernet
- 2 SD sockets
- 10 serial ports
- 110 GPIO
- 5 10-bit ADC
- 2 SATA ports
- Sleep mode uses 200 microamps
- Boots Linux in < 2 seconds
- Linux 2.6 and Debian by default

Design your solution with one of our engineers

- Over 20 years in business
- Never discontinued a product
- Engineers on Tech Support
- Open Source Vision
- Custom configurations and designs w/ excellent pricing and turn-around time
- Most products stocked and available for next day shipping

See our website for options, peripherals and x86 SBCs

Technologic SYSTEMS

We use our stuff.

visit our TS-7200 powered website at

www.embeddedARM.com

(480) 837-5200

How to Use Quanta Plus, the Web Developer Tool with Everything but the Kitchen Sink

Web developers spend lifetimes looking for the perfect Web development tool. This article is a step-by-step tutorial on how to use Quanta Plus, the open-source app that thinks it's the one. ANDREW MIN

Many Web developers spend their entire coding career looking for the perfect Web development tool. Many Windows and OS X users have found it in Adobe's Dreamweaver. However, Dreamweaver doesn't have a native Linux version and runs spottily on Wine. Additionally, Dreamweaver is closed source and costs about $399. Luckily, the Open Source world has produced a viable alternative: Quanta Plus.

What's Quanta Plus and How Do I Get It?

Quanta Plus originally was created by a Linux corporation called theKompany as an open-source Web development tool. Unfortunately, theKompany abandoned Quanta Plus in favor of its own commercial product, Quanta Gold. Fortunately, Quanta Plus was picked up by the KDE Web Development team and continued its life as an open-source product.

If you're a KDE user, Quanta Plus is most likely installed already, as it comes as a part of the kdewebdev package.

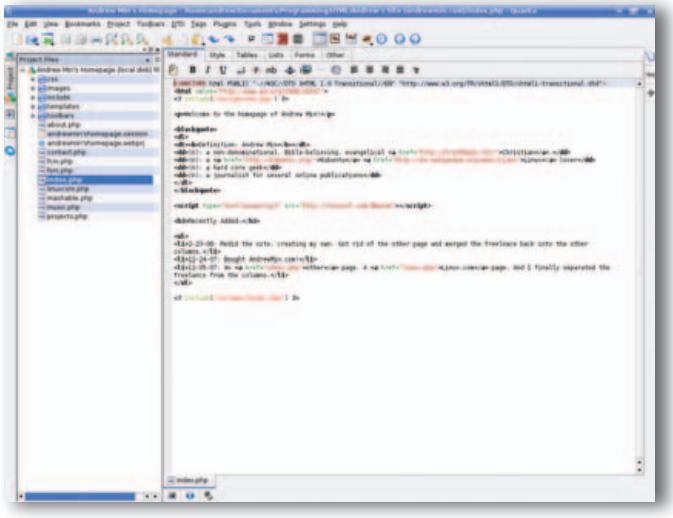


Figure 1. Quanta Plus Editing My Home Page

But if you aren't a KDE user or you don't have it pre-installed, you probably can find it in your operating system's repositories (for example, Ubuntu users can install it by using apt-get to install the package quanta). If you can't find it or your distribution doesn't have a package system, you can compile it from source easily. Simply download the .tar.bz2 release from the Quanta Plus home page, and follow the INSTALL instructions. Note that you'll have to install the whole kdewebdev shebang instead of just Quanta Plus if you're compiling from source.

Basic Features

The first time you open Quanta Plus, it runs a wizard to help you set up your first project. At the first window, you can name your project and choose where the files are stored. If the Web pages are created on your hard drive, use the default, Local. You also can specify FTP if the working files are created on an FTP directory, fish if they are stored over SSH, or any of the other protocols. You also must choose a main directory (your working directory) and a folder for your templates and toolbars. At the next set of screens, you can change the default DTD for the project, the encoding, author details and much more. When you are done, click Finish to go back to the main Quanta Plus interface. If you want, you can configure more advanced properties under Project→Project Properties (or by pressing Shift-F7).

Now that you've created a project, you easily can create, add existing, or edit existing Web pages in your project. All your existing files (if any) in the working folder should be in the project manager already (access it by clicking the Project tab on the sidebar). If they aren't, you can add them by right-clicking on your project in the Project tree and selecting Rescan Project Folder... To create a new file, click File→New (or press Ctrl-N), and save it. Once you do, you will prompted whether to add it to the project. If you do, it will show up nicely in the project manager.

Quanta Plus defaults to the hand-coded view, where it

shows the complete (X)HTML source of the page. At first, you may not think this is much better than using a plain-text editor like Notepad. However, if you're used to using Notepad for your coding, you'll find some nice surprises. First, Quanta Plus offers code highlighting. Different tags in your code are highlighted to create a more readable feel (for example, in a simple link tag, the <a part is in bold, the href= is in green, and the http:// is in red). It's a very helpful way to make your code a lot easier to skim.

Another nice feature is the automatic tag completion. Usually, you have to type all the code by hand and without forgetting to close all the tags. But with Quanta Plus, your end tag is inserted automatically as soon as you close your beginning tag (so when the > in is inserted, a is inserted automatically). Additionally, a lot of the tag itself is generated automatically (when you type <a , the href="" " part of the link tag is inserted).

There's also a terrific tag toolbar that Quanta Plus ships with, located right above the document. Clicking one of the buttons (for example, the Bold button) inserts the appropriate tag set (for example,). You also can highlight text and click the button to change the formatting for the highlighted words (for example, highlighting the word firefox and clicking the Bold button puts firefox between a and a tag). And, you aren't limited only to the bold, italic, underline, new line, paragraph, nbsp, anchor, image, hl, comment or align tags. You also can get into advanced formatting by switching from the default Standard tab to the Style, Table, List, Form or other tabs on the toolbar.

Unfortunately, we're not all HTML-savvy. Sometimes, all you want to do is create a Web page without having to jump through all the HTML/CSS hoops. In Dreamweaver, many people fell in love with the great WYSIWYG (What You See Is What You Get) editor built in to the powerful Web developing tool. Quanta Plus has a similar (if not quite as good) mode called VPL (Virtual Preview Layout).

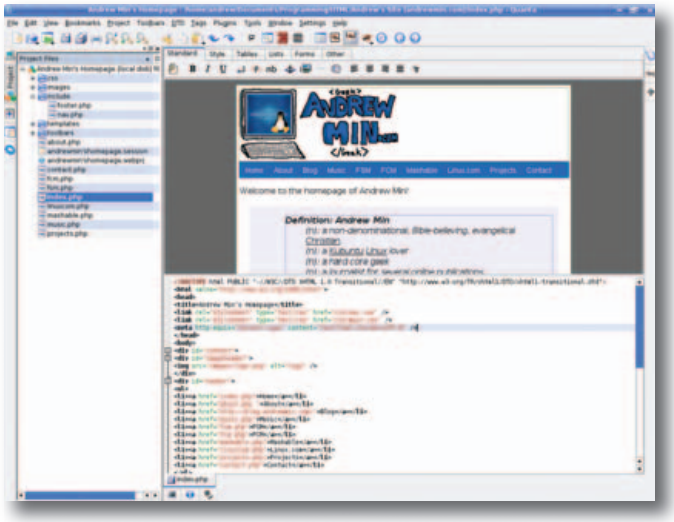


Figure 2. The VPL Split Screen

To activate it, go to View→VPL Editor or press Ctrl-Shift-F9. You'll be presented with a simple WYSIWYG editor with support for basic formatting using the tag toolbar mentioned above. You also can split the screen between the VPL and source by going to View→VPL & Source Editors or by pressing F9. It's a great way to get an instant preview.

Of course, a project that sits on your hard drive is basically useless. Luckily, Quanta Plus offers a handy FTP uploader. To set it up, go to Project→Project Properties (or Shift-F7) and then to the Upload Profiles tab. Click the Edit Profiles button and then the New button. Give it a name (like Dreamhost), a hostname (like andrewmin.com), a user name (like andrew) and a password. You also may want to put in a path, such as /andrewsapps.com/, if the root of your FTP site isn't the root of your Web site. If you're not

Sometimes, all you want to do is create a Web page without having to jump through all the HTML/CSS hoops.

uploading over FTP, change the protocol to the appropriate KIO protocol (like fish for SSH). Then click OK and go back to the main Quanta Plus window. Now, simply click over to Project→Upload Project (or press F8). Make sure your profile name is selected at the top, and then click the Proceed button. Your project will be uploaded and should become live instantly.

Advanced Features

Quanta Plus also has a lot really nice advanced features. For example, there is the nice built-in bookmark feature. To use it, simply select the line you want to bookmark, and then click Bookmarks→Add Bookmark. Every time you want to go to that line again, simply click the bookmark (under the Bookmarks menu), and you'll jump to that location automatically. This is especially useful for those times when you need to debug something but can't do it right away.

Of course, Quanta Plus can't do everything by itself. However, that's where the powerful plugins come in. For example, there is the HTML Tidy plugin that comes with Quanta Plus. To activate it, run Tools→HTML Tidy Syntax Checking (or Ctrl-Alt-T). This essential tool debugs your code, pointing to all those nonstandard-compliant code bits that you just happened to leave out (or leave in). Another powerful plugin I frequently use is KFileReplace (Plugins→KFileReplace). Quanta Plus has a decent Find and Replace program built in, but it searches in only the current document. However, I often want to find and replace a certain word in *every* file in my project. For example, if I change my domain from andrewmin.net to andrewmin.com, I want to replace all instances of andrewmin.net with andrewmin.com. That's when KFileReplace comes in. Just run it, type in what to search for (like andrewmin.net) and what to replace it with (like andrewmin.com), the location and the filter (maybe you don't want to replace

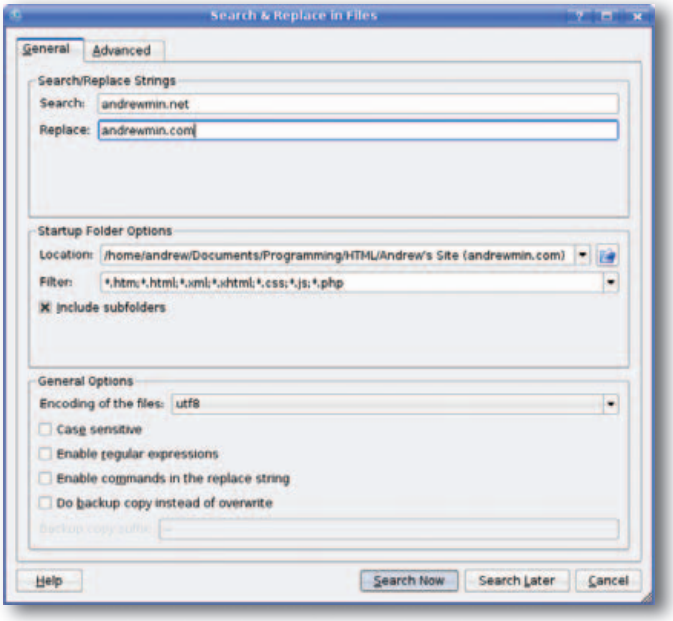


Figure 3. The Find and Replace Dialog

andrewmin.net in your HISTORY.TXT file) and click Search Now. KFileReplace will hunt down every string that includes your search term and replace it with your term. There are many more (look under the Tools menu and Plugins menu to see most of them).

Conclusion

Obviously, no Web development tool is perfect. And one day, perhaps there will be a perfect tool that will do everything you want it to do and then some. When that day comes, I'll be the first to download it. Until then, you can find me right here, using my wonderful Quanta Plus.■

Andrew Min's been a geek since he threw his first tantrum over the Mickey ABC's: A Day at the Fair. He's also a freelance writer who has done work for several other technology publications, an evangelical Christian, a proud American and a New York Yankees fan. You can find out more about him at www.andrewmin.com.

Resources

Quanta Plus Home Page: **quanta.kdewebdev.org**

Mailing List: **https://mail.kde.org/mailman/listinfo/quanta**

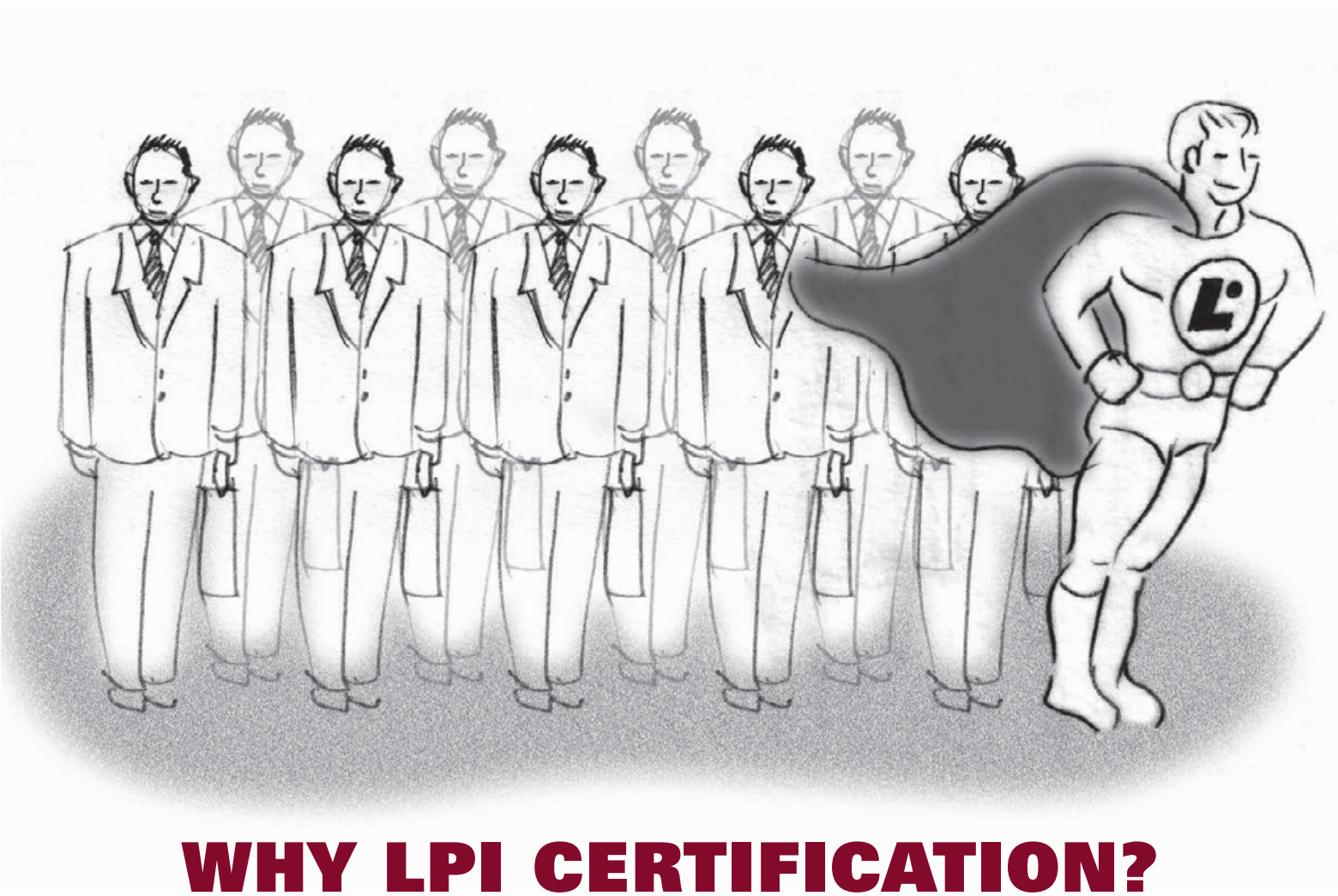
theKompany (Original Developers): **www.thekompany.com**

Quanta Gold: **www.thekompany.com/products/quanta**

***Did you know* Linux Journal *maintains a mailing list where list members discuss all things Linux? Join* LJ's *linux-list today:* http://lists2.linuxjournal.com/mailman/listinfo/linux-list.**

www.LinuxJournal.com/ArchiveCD

The 1994–2007 Archive CD, back issues, and more!



WHY LPI CERTIFICATION?

RELEVANCE

- #1 Linux certification worldwide and growing
- Program framework created from industry needs and input
- Professional "Job Task Analysis"

CREDIBILITY

- Designed by professionals for professionals
- Internationalization through regional involvement
- Endorsed by global leaders in Open Source
- Recognized and accredited psychometric processes

VALUE

- A global standard in Linux professionalism
- Proven demonstration of knowledge and skills for customers and employers
- Provides benchmarks for HR recruitment and promotion
- Access to global network of professionals

Linux Professional Institute LPIC-1

Linux Professional Institute LPIC-2

Linux Professional Institute LPIC-3

OSPRE

Linux Professional Institute

www.lpi.org

EOF

A Tale of Two Futures

Generativity is the hot new word, and nothing fits it better than Linux and the development it supports. But, will we use our generative powers to save the Net? DOC SEARLS

We've long since lost count of free and open-source (FOSS) codebases. Last I heard, the sum was passing half a million. If we were to visualize these as a tree, it would resemble a banyan—wide and flat, a forest in itself, with one main trunk in the middle and smaller ones under its radiating branches. That main trunk would be Linux. The ground would be the Internet.

Why has this vast organism grown so broadly and rapidly, with no end in sight? Many answers may come to mind, but I suggest one that should be new to *Linux Journal* readers—as it was to me, when I first heard it from Jonathan Zittrain. That answer is generativity.

In his new book, *The Future of the Internet—And How to Stop It* (Yale University Press, 2008), Jonathan defines generativity as "a system's capacity to produce unanticipated change through unfiltered contributions from broad and varied audiences". In an earlier research paper, "The Generative Internet", he explained, "The grid of PCs connected by the Internet has developed in such a way that it is consummately generative. From the beginning, the PC has been designed to run almost any program created by the manufacturer, the user, or a remote third party and to make the creation of such programs a relatively easy task."

Linux and the FOSS portfolio fit this description, and so do its developers. In fact, I submit that both are even more generative than the wide-open machines they put to work. But, although it would be nice to see FOSS programmers credited with setting new records for generativity, what I'd rather see is those same programmers playing a leading role in preserving and expanding the Net's generative power.

According to Jonathan, the future does not default to rosy. In fact, he says the Net's generative growth is stalling. "The future unfolding right now is very different from its past", he writes. "The future is not one of generative PCs attached to a generative Internet. It is instead one of sterile appliances tethered to a network of control." Among those appliances, he lists Microsoft's Xbox 360, Apple's iPhone and TiVo DVRs. Thus, we stand at a fork between two futures: one generative, the other applianced—and the fight being won by the latter.

Linux and FOSS programmers are not innocent bystanders in this fight between futures. They contribute to both. As Jonathan puts it:

> ...generative and non-generative models are not mutually exclusive. They can compete and intertwine within a single system. For example, a free operating system such as GNU/Linux can be locked within an information appliance like the TiVo, and classical, profit-maximizing firms like Red Hat and IBM can find it worthwhile to contribute to generative technologies like GNU/Linux.

The generative/applianced divide is one between cultures as well as work, and we have geeks laboring on both sides of it. One side creates code that is both useful and re-usable—whether it's a leaf on the collective FOSS banyan tree, or humus in the networked ground on which that tree grows. The other side does what The Man tells it to do, even if the job is equipping an appliance to do something closed on top of open code.

What's strange is that both are mundane. They are not romantic. They do not supply fodder for partisan arguments. They are not box office. They are simply useful. This enormously productive (and reproductive) practicality is perhaps the most plain yet overlooked fact about FOSS development. Even within our community, we don't think much about how successful, common and purely generative our work is—and how much it has contributed to the growth and success of the Net. We just do good work, have fun and press on.

Yet there are these two sides. One thrives in the open world while the other disappears into machines. One makes stuff that is NEA: Nobody owns it, Everybody can use it, and Anybody can improve it. The other makes stuff that is OOO: One company owns it, Only its customers can use it, and Only the company and its captive partners can improve it.

Perhaps both will win, but maturing markets preponderate toward the simple and the predictable, rather than the complicated and the chaotic. For technology, that favors the applianced over the generative.

I've always been an optimist about generativity, even though I didn't know the word until a few months ago. But I see Jonathan's case, and it has me worried. There is no shortage of closed appliances that run Linux. Sometimes we don't even know they're around. Both my Sony Bravia 1080p flat-screen and the Dish Network set-top box that feeds it have Linux operating systems. And, both are built to prevent far more generativity than they enable.

Back in 2002, I wrote a piece titled "A Tale of Three Cultures" (**www.linuxjournal.com/article/5912**). One culture was FOSS hackers. One was embedded systems programmers. And the third was Hollywood, feeding popular culture. Toward the end of that piece, I offered a challenge: "And if we are asked by our employers and our government to replace the people's Net with a corporate digital rights management system, will we go about it as heads-down technologists? Or will we refuse to build it?"

That challenge still stands.■

Doc Searls is Senior Editor of *Linux Journal*. He is also a Visiting Scholar at the University of California at Santa Barbara and a Fellow with the Berkman Center for Internet and Society at Harvard University.

NEW LOOK. More **IT HOSTING**.
All the **FANATICAL SUPPORT**®
You'll Ever Need.

experience
fanatical support®

WWW.RACKSPACE.COM

Experience Lightning Without The Thunder℠

WhisperStation-Pro™

1 TERAFLOP IN A COOL, FAST, RELIABLE PLATFORM!

Whether it's Wall Street, Main Street or Your Street, Microway's new Nvidia-powered WhisperStation-Pro is energy-efficient, designed for superior performance, and best of all - QUIET.

Originally designed for a group of power hungry, demanding engineers in the automotive industry, WhisperStation-Pro incorporates two AMD® Opteron™ or Intel® Xeon® quad-core processors and high-efficiency power supplies. Ultra-quiet fans and internal sound-proofing produce a powerful, but silent, computational platform.

WhisperStation-Pro configured with one Quad core processor, 4 GB high speed memory, 250 GB drive, dual-GigE, NVIDIA® Quadro™ FX570 graphics and 20" LCD – **starts at $1995.**

You can have it configured to your exact needs with NVIDIA GeForce® or Quadro graphics adapters (including SLI®), NVIDIA Tesla™ GPU, any Linux distribution, or Windows® dual-boot. Also, there is plenty of room for RAID storage expansion. From a home based workstation for financial wizards, to a superior gaming or design station, WhisperStation-Pro fits the bill and your budget.

Visit www.microway.com for more technical information.

WhisperStation™ 3D Elite SLI For Gamers

Hear Yourself Think Again!

Call our technical sales team at 508-746-7341 and customize your WhisperStation-Pro today.

AMD Opteron 64 | NVIDIA PARTNERFORCE PREMIER | intel Xeon inside™

Microway®

Technology you can count on℠