

SPECIAL COLLECTOR'S ISSUE

LINUX JOURNAL™

Since 1994: The Original Magazine of the Linux Community

www.linuxjournal.com

THE VOTES ARE IN!



Your Favorite Gadgets, Programs, Tools, Hardware & More!

2000TH ISSUE CELEBRATION!

Parallel Programming with NVIDIA | Control a Fridge with Pogoplug | How-To: PiTiVi Video Editor | 200 Things to Do with Linux | Web Retrospective and Predictions

DECEMBER 2010 | ISSUE 200



REVIEWED

ZOTAC ZBOX HD-ID11 and Barnes & Noble's Nook

More TFLOPS, Fewer WATTS

Microway delivers the fastest and greenest floating point throughput in history

Enhanced GPU Computing with Tesla Fermi

- ▶ 480 Core NVIDIA® Tesla™ Fermi GPUs deliver 1.2 TFLOP single precision & 600 GFLOP double precision performance!
- ▶ New Tesla C2050 adds 3GB ECC protected memory
- ▶ New Tesla C2070 adds 6GB ECC protected memory
- ▶ Tesla Pre-Configured Clusters with S2070 4 GPU servers
- ▶ WhisperStation - PSC with up to 4 Fermi GPUs
- ▶ OctoPuter™ with up to 8 Fermi GPUs and 144GB memory

New Processors

- ▶ 12 Core AMD Opterons with quad channel DDR3 memory
- ▶ 8 Core Intel Xeons with quad channel DDR3 memory
- ▶ Superior bandwidth with faster, wider CPU memory busses
- ▶ Increased efficiency for memory-bound floating point algorithms

Configure your next Cluster today!

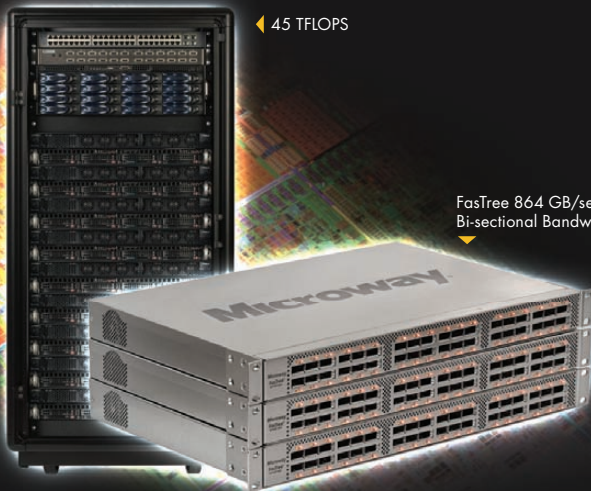
www.microway.com/quickquote
508-746-7341



2.5 TFLOPS

10 TFLOPS

5 TFLOPS



4.5 TFLOPS

FasTree 864 GB/sec
Bi-sectional Bandwidth

FasTree™ QDR InfiniBand Switches and HCAs

- ▶ 36 Port, 40 Gb/s, Low Cost Fabrics
- ▶ Compact, Scalable, Modular Architecture
- ▶ Ideal for Building Expandable Clusters and Fabrics
- ▶ MPI Link-Checker™ and InfiniScope™ Network Diagnostics

Achieve the Optimal Fabric Design for your Specific MPI Application with ProSim™ Fabric Simulator

Now you can observe the real time communication coherency of your algorithms. Use this information to evaluate whether your codes have the potential to suffer from congestion. Feeding observed data into our IB fabric queuing-theory simulator lets you examine latency and bi-sectional bandwidth tradeoffs in fabric topologies.



GSA Schedule
Contract Number:
GS-35F-0431N

Microway
Technology you can count on™

WEBSITE PLANS ON SALE!

**ALL WEB HOSTING
PACKAGES JUST:**

\$3.99
per month*
For the first 3 months!



Whether you're a beginner or a professional, 1&1 offers a full range of website solutions to suit your needs. For a limited time, we're offering all web hosting packages at one incredible low price. Website building tools, unlimited traffic, and search engine marketing dollars are included with all packages.

Go to www.1and1.com to choose your package!

DOMAIN OFFERS:

.info only \$0.99 first year* **.com** only \$7.99 first year*



Free Web
Marketing Tools



90-Day Money
Back Guarantee



24/7 Toll-Free
Support



Call 1-877-GO-1AND1 or visit us now

www.1and1.com

*Offers begin November 1, 2010. 12 month minimum contract term applies for web hosting offers. Setup fee and other terms and conditions may apply. Domain offers valid first year only. After first year, standard pricing applies. Visit www.1and1.com for full promotional offer details. Program and pricing specifications and availability subject to change without notice. 1&1 and the 1&1 logo are trademarks of 1&1 Internet AG, all other trademarks are the property of their respective owners. © 2010 1&1 Internet, Inc. All rights reserved.

200th ISSUE CELEBRATION

FEATURE

52 Readers' Choice Awards 2010

Your favorite hardware, software, gadgets and more.
James Gray

ON THE COVER

- Readers' Choice Awards 2010, p. 52
- Parallel Programming with NVIDIA, p. 64
- Control a Fridge with Pogoplug, p. 36
- How-To: PiTiVi Video Editor, p. 70
- 200 Things to Do with Linux, p. 18
- Web Retrospective and Predictions, p. 24
- ZOTAC ZBOX HD-ID11, p. 46
- Barnes & Noble's Nook, p. 48



They say work smarter, not harder. They must be using our processor.



Sponsors of Tomorrow.™

The next generation of intelligent server processors

The Intel® Xeon® processor 5600 series automatically regulates power consumption to combine industry-leading energy efficiency with intelligent performance that adapts to your workload. Check out the new intelligent features of the Xeon 5600 at intel.com/itcenter.



SEE WHAT
INTELLIGENCE
CAN DO



Servers from iXsystems feature the Intel® Xeon® processor 5600 series.



Enterprise Servers
for Open Source
www.ixsystems.com
800-820-BSDi

Intel is not responsible for and has not verified any statements or computer system product-specific claims contained herein.

The **Triton Twin-Blade Server** supports up to 120 dual processor servers with 240 Intel Xeon 5600 series processors per 42U rack, achieving an unmatched .35U per dual processor node.

Cleverly disguised as any other 2U server, the **Gemini²** houses 4 highly efficient, extremely powerful RAID 5-capable servers.



Triton Twin-Blade

- Intel® Xeon® processor 5600 series
- 20 server compute nodes in 7U of rack space
- N+1 redundant, high efficiency (94%) 2500W power supplies
- Remote access through SOL, KVM, KVM over IP technologies

Gemini²

- Intel® Xeon® processor 5600 series
- 4 nodes in a 2U form factor
- Up to 192GB DDR3 SDRAM per node
- 3 x 3.5" SAS/SATA drives per node



CONTENTS

DECEMBER 2010

Issue 200

COLUMNS

- 24** Reuven M. Lerner's At the Forge
Issue 200
- 28** Dave Taylor's Work the Shell
Generating Turn-by-Turn Driving Directions
- 30** Mick Bauer's Paranoid Penguin
Building a Transparent Firewall with Linux, Part IV
- 36** Kyle Rankin's Hack and /
Working on My Temper
- 76** Kyle Rankin and Bill Childers'
Point/Counterpoint
Bill and Kyle vs. *LJ* Readers
- 80** Doc Searls' EOF
Remembering the Future

REVIEWS

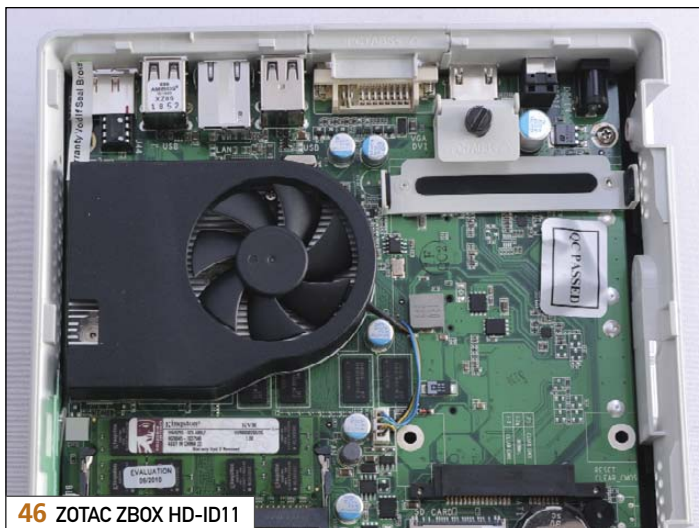
- 46** ZOTAC ZBOX HD-ID11
Steven Evatt
- 48** Barnes & Noble's Nook
Bill Childers

INDEPTH

- 64** Parallel Programming with NVIDIA CUDA
Use your GPU to speed up your algorithms by
ten-fold or more.
Alejandro Segovia
- 70** Getting Started with PiTiVi
Video editing for the newcomer.
Jono Bacon

IN EVERY ISSUE

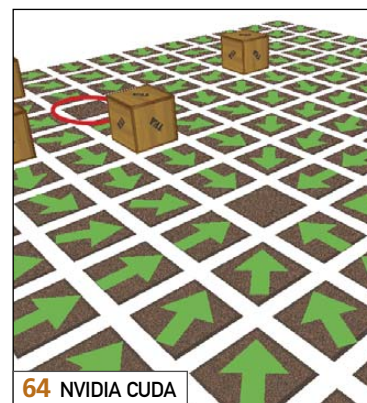
- 8** Current_Issue.tar.gz
- 10** Letters
- 14** UPFRONT
- 40** New Products
- 42** New Projects
- 65** Advertisers Index
- 74** Tech Tips
- 79** Marketplace



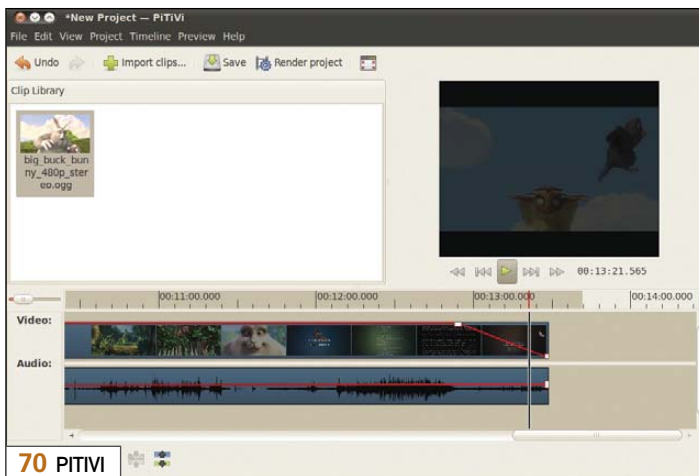
46 ZOTAC ZBOX HD-ID11



48 BARNES & NOBLE'S NOOK



64 NVIDIA CUDA



70 PITIVI

USPS *LINUX JOURNAL* (ISSN 1075-3583) (USPS 12854) is published monthly by Belltown Media, Inc., 2121 Sage Road, Ste. 310, Houston, TX 77056 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 16476, North Hollywood, CA 91615. Subscriptions start with the next issue. Canada Post: Publications Mail Agreement #41549519. Canada Returns to be sent to Pitney Bowes, P.O. Box 25542, London, ON N6C 6B2

STORAGE WITHOUT BOUNDARIES

Imagine what you can achieve with Aberdeen.

Aberdeen's AberSAN Z-Series scalable storage platform brings the simplicity of network attached storage (NAS) to the SAN environment, by utilizing the innovative ZFS file system.

Featuring the high performance of the Intel Xeon processor 5600 series, the AberSAN Z-Series offers enterprise level benefits at entry level pricing, delivering the best bang for the buck.

Who gives you the best bang for the buck?

	NetApp FAS 2050	EMC NS120	Aberdeen AberSAN Z20
Deduplication	✓	✓	✓
Thin Provisioning	✓	✓	✓
VMware® Ready Certified	✓	✓	✓
Citrix® StorageLink™ Support	✓	✓	✓
Optional HA Clustering	✓	✓	✓
Optional Virtualized SAN	✓	✓	✓
Optional Synchronous Replication	✓	✓	✓
Optional Fibre Channel Target	✓	✓	✓
Unified Storage: NFS, CIFS, iSCSI	✓	✓	✓
Unlimited Snapshots	✗	✗	✓
Unlimited Array Size	✗	✗	✓
Hybrid Storage Pooling	✗	✗	✓
Integrated Search	✗	✗	✓
File System	WAFL 64-bit	UxFS 64-bit	ZFS 128-bit
RAID Level Support	4 and DP	5 and 6	5, 6 and Z
Maximum Storage Capacity	104TB	240TB	Over 1,000TB
Warranty	3 Years	3 Years	5 Years
Published Starting MSRP	Not Available	Not Available	\$8,995



Above specific configurations obtained from the respective websites on Sept. 13, 2010. 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. All trademarks are the property of their respective owners. All rights reserved. For terms and conditions, please see www.aberdeenninc.com/abopoly/abterms.htm. IJ036

888-297-7409
www.aberdeenninc.com/lj036

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/DLISSUE



LINUX JOURNAL

Executive Editor Jill Franklin
jill@linuxjournal.com

Senior Editor Doc Searis
doc@linuxjournal.com

Associate Editor Shawn Powers
shawn@linuxjournal.com

Associate Editor Mitch Frazier
mitch@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

Security Editor Mick Bauer
mick@visi.com

Hack Editor Kyle Rankin
lj@greenfly.net

Virtual Editor Bill Childers
bill.childers@linuxjournal.com

Contributing Editors

Ibrahim Haddad • Robert Love • Zack Brown • Dave Phillips • Marco Fioretti • Ludovic Marcotte
Paul Barry • Paul McKenney • Dave Taylor • Dirk Elmendorf • Justin Ryan

Proofreader Geri Gale

Publisher Carlie Fairchild
publisher@linuxjournal.com

General Manager Rebecca Cassidy
rebecca@linuxjournal.com

Senior Sales Manager Joseph Krack
joseph@linuxjournal.com

Associate Publisher Mark Irgang
mark@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

Editorial Advisory Panel

Brad Abram Baillio • Nick Baronian • Hari Boukis • Steve Case
Kalyana Krishna Chadalavada • Brian Conner • Caleb S. Cullen • Keir Davis
Michael Eager • Nick Faltys • Dennis Franklin Frey • Alicia Gibb
Victor Gregorio • Philip Jacob • Jay Kruizenga • David A. Lane
Steve Marquez • Dave McAllister • Carson McDonald • Craig Oda
Jeffrey D. Parent • Charnell Pugsley • Thomas Quinlan • Mike Roberts
Kristin Shoemaker • Chris D. Stark • Patrick Swartz • James Walker

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: +1 818-487-2089
FAX: +1 818-487-4550
TOLL-FREE: 1-888-66-LINUX
MAIL: PO Box 16476, North Hollywood, CA 91615-9911 USA
Please allow 4-6 weeks for processing address changes and orders
PRINTED IN USA

LINUX is a registered trademark of Linus Torvalds.



We Speak Your Language

```
/*
 * Check if user is interested in a great deal on a
 * dedicated server with Codero
 */

switch(ucwords(strtolower($what_are_you_looking_for))) {
  case 'A Great Server Deal':
    header("Location: http://www.codero.com/specials/");
    break;
  case 'An Infrastructure Ally':
  default:
    header("Location: http://www.codero.com/");
    break;
}
```

1.877.999.2701
www.codero.com

Server Solutions





SHAWN POWERS

Issue Number 11001000!

That's binary for 200, of course. I mean, sure, we've been around for a long time, but not since 914,740 BC! Back then, the latest technology was "Wheel 2.0: Round Edition", and although it was great at the time, it was a hard sell on newsstands—especially since newsstands hadn't been invented yet.

This is an exciting issue, and we thought it would be great to include our readers in it as much as possible. If you check out the UpFront section, you'll see a couple articles that are reader contributions. We've got a list of 200 things you like to do with Linux and some reader profiles for you as well. Do you ever wonder who else is a *Linux Journal* fan? Now you can meet a bunch of them!

This is also our annual Readers' Choice issue, where you all vote on your favorite applications, gadgets, columns and so on. To find where you skew along the average readership line, be sure to read the results. A few of them surprised us.

Sure, all this reader contribution is great, but to be honest, we didn't want to be outsourced completely. So we included a hearty helping of articles as well. Reuven M. Lerner reminisces about the early years of Web development and guides us through its progression. Some things certainly have changed, and some things really haven't! Add to that Dave Taylor's column on creating your own turn-by-turn directions from the command line, and you've got a bunch of programming information in just two articles.

Mick Bauer is back this month with the next part in his series on transparent firewalls. Firewalls obviously are great for security and protecting your network from snoops. Kyle Rankin is probably happy about that, as he might get teased if anyone snooped in and saw his pretty pink server. He says it's part of his beer-making rig, but all we can see is pink. Check out his Hack and / column and decide for yourself.

If you'd rather celebrate the 200th issue by buying tech toys, we totally understand. In fact, we have some great reviews on products we'd like to own ourselves. The ZOTAC ZBOX, reviewed by Steven Evatt, is a tiny little computer designed for (or at least often repurposed for) making home entertainment computers. With its onboard ION chipset, it packs a punch in its tiny enclosure. Bill Childers reviews the Barnes & Noble's Nook.

Bill and I both bought one of these Linux-based devices at about the same time. He loves his, and I hated mine. Check out his review to see where you stand on the highly hackable e-reader.

If all that seems a bit too fluffy for your technology taste, you might just love Alejandro Segovia's in-depth piece on parallel computing with NVIDIA's CUDA technology. Video cards are great for gaming, but it's amazing how powerful they can be when you use them for straight up mathematical processing. Alejandro shows how to take advantage of the little powerhouse sitting inside your computer case.

Video editing in Linux is a really hot topic, especially with everyone desiring to upload the next great viral video to YouTube. Several great video editors exist, and although they all are good at what they do, each one seems lacking in one area or another. Canonical has decided to put its faith in PiTiVi, a simple video editor installed by default in new Ubuntu systems. As with any video editing software, PiTiVi has a learning curve. Jono Bacon shows us how to get started quickly with PiTiVi, and he demonstrates that although it may not be the most complex video editor available, it's very usable.

We always argue here at *Linux Journal* about which issue is our favorite. For some of us it's the Cool Projects issue; others prefer the Web Development issue. One thing we can all agree on is that the Readers' Choice issue is always fun. Even Bill Childers and Kyle Rankin get along fairly well in this month's Point/Counterpoint column. Of course, they tend to argue with the readers this time, so I'm not sure it counts as peaceful. (And for the record, I'm with the readers. Pidgin is an awesome IRC client, no matter what Bill and Kyle say!)

So whether you frame this issue of *Linux Journal* because you're in one of the UpFront pieces or roll it up to swat a fly before reading it, we hope you enjoy the 200th issue. Hopefully, we'll be talking about world domination 200 issues from now, but until then, we hope you all party like it's 914,740 BC! ■

Shawn Powers is the Associate Editor for *Linux Journal*. He's also the Gadget Guy for LinuxJournal.com, and he has an interesting collection of vintage Garfield coffee mugs. Don't let his silly hairdo fool you, he's a pretty ordinary guy and can be reached via e-mail at shawn@linuxjournal.com. Or, swing by the #linuxjournal IRC channel on Freenode.net.

WANT YOUR BUSINESS TO BE MORE PRODUCTIVE?

The ServersDirect[®] Systems powered by Intel[®] Xeon[®] Processor provides the quality and dependability to keep up with your growing business.



HIGH EFFICIENCY POWER & HIGH STORAGE CAPACITY SERVER SDR-S4314-T36

ENTERPRISE-LEVEL HIGH CAPACITY STORAGE SERVER

- + Maximum 3.5" hot-swap drives density
 - 36x (24 front + 12 rear) Hard Drive Bays
 - E1: Expander supports SAS
- + Redundant (1+1) 1400W Gold Level power supply with PMBus function
- + Redundant, Hot-pluggable cooling system, Power Supplies, Hot-swap drives
- + Support up to 192GB DDR3 1333/ 1066/ 800MHz ECC Reg DIMM
- + Intel[®] 82576 Dual-Port Gigabit Ethernet Controller



SDR-S1343-T04

1U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 4X 3.5" HOT-SWAP SATA DRIVE BAYS

- Supermicro 1U Rackmount Server with HE 560W Power Supply
- Supermicro Server Board w/Intel[®] 5520 Chipset
- Support up to Dual Intel[®] 5500 series Xeon[®] Quad/ Dual-Core, with QPI up to 6.4 GT/s
- Support up to 96GB DDR3 1333/ 1066/ 800MHz ECC Reg. DIMM
- 4x 3.5" Hot-swap SATA Drive Bays
- Intel[®] 82576 Dual-Port Gigabit Ethernet Controller



SDR-S2318-T06

2U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 8X 3.5" HOT-SWAP SAS/SATA BAYS

- Supermicro 2U Rackmount Server w/650W HE Power Supply
- Supermicro Server Board w/Intel[®] 5520 Chipset
- Support up to Dual Intel[®] 5500 series Xeon[®] Quad/ Dual-Core, with QPI up to 6.4 GT/s
- Support up to 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 6x 3.5" Hot-swap SAS / SATA Drive Bays
- Dual Intel[®] 82574L Gigabit Ethernet Controller



SDR-S3308-T08

3U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 8 X 3.5" HOT-SWAP DRIVES TRAYS

- Supermicro 3U Rackmount Server w/650W HE Power Supply
- Supermicro Server Board w/Intel[®] 5520 Chipsets
- Support up to Dual Intel[®] 5500 series Xeon[®] Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 8 x 3.5" Hot-swap Drives Trays 6 x SATA Hard Drives Supported
- Dual Intel[®] 82574L Gigabit Ethernet



SDR-S4309-T08

TOWER/4U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 8 X 3.5" HOT-SWAP DRIVES TRAYS

- Supermicro Tower 800W Redundant Power Supply
- Supermicro Server Board w/ Dual Intel[®] 5520 Chipsets
- Support up to Dual Intel[®] 5500 series Xeon[®] Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- 24x 3.5" Hot-swap SATA Drive Bay
- Intel[®] 82576 Dual-port Gigabit Ethernet Controller



SDR-S4314-T08

TOWER/4U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 8X 3.5" HOT-SWAP SATA BAYS (SUPPORT UP TO 4 DOUBLE-WIDTH GPU)

- 4U Rackmountable / Tower with 1400W Gold Level Redundant Power Supply Optional Rackmount Kit
- Supermicro Server Board w/ Dual Intel[®] Dual Chipset ICH10R
- Support up to Dual Intel[®] 5500 series Xeon[®] Quad/Dual-Core, with QPI up to 6.4 GT/s
- Support up to 192GB DDR3 1333/ 1066 / 800MHz ECC Reg DIMM
- " 8 x 3.5" Hot-swap Drives Trays Supports up to 6 SATA Drives"
- Intel[®] Dual 82574L Gigabit Ethernet
- 4x NVIDIA Tesla C1060 GPU Cards



SDB-S7002-T00

9U INTEL[®] XEON[®] PROCESSORS 5500 SERIES SERVER W/ 50X HOT-SWAP SATA II / SAS BAYS

- Supermicro SBE-714D-D28 Enclosure chassis with two 1400W power supplies. Up to 14 hot-plug processor blades
- Support Intel based blades
- One Management modules comes standard with each enclosure
- One hot-plug management modules providing remote KVM and IPMI 2.0 functionalities
- Support one hot-plug Gigabit Ethernet switch, Pass - Through Module



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 to assemble systems that require a little extra. Servers Direct - your direct source for scalable, cost effective solutions.

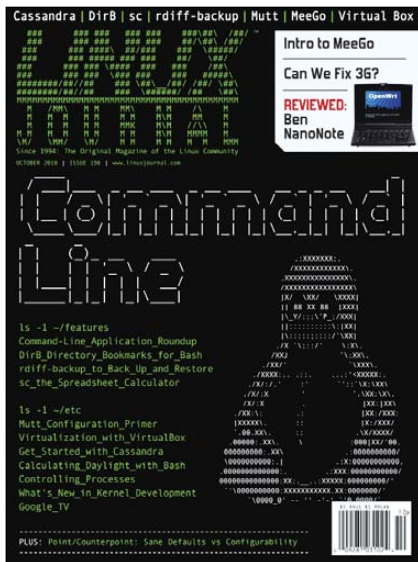
1.877.727.7886 / www.ServersDirect.com

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.



Powerful. Intelligent.

Letters



Facebook

As an inhabitant of the Great Outdoors (Gisborne, New Zealand, aka “First City of the Sun”), I most vigorously applaud Doc Searls’ insightful debunk of that facile phenomenon, Facebook [see Doc’s “Waving Goodbye to Facebook” in the August 2010 issue]. Zuckerberg’s only claim to glory, that I can see, is that he used the Internet to organise the mighty sales predator and manipulate the great American unwashed into making him a billionaire at age 26. The idea that this young man and his cohort should “seize control of the Internet” is enough to make hair grow in unspeakable places. Let the kids go to the mall if they wish. Me? I’ll make sail for discovery bay.

It is my pleasure to express the gratitude felt to Mr Searls and his ilk for their care and compassion in steering true through inevitable turbulence.

--
John McClatchie

UDEV

I would like to see an article that evaluates which distro has the easiest UDEV rules to work with. I have spent hours trying to write a rule on openSUSE and have not gotten the desired results. Is Ubuntu or Fedora easier to work with?

--
Robert Finlon

Honestly, UDEV rules are confusing regardless of the distribution. Perhaps it's something we should cover. Thanks for sharing your misery!—Ed.

3G Hell or Twitter Client Heaven?

I just read Doc Searls’ EOF “3G Hell” in the October 2010 issue, and I’m shocked—but, not about the national-fenced data access. I live in Europe; it’s a well-known nightmare and quite a long matter. Let’s just say that it’s kind of an indirect battle and not strictly Linux-related.

The shocking bit was to learn that not even *Linux Journal’s* Senior Editor is completely in control of what’s running on his (Linux-based)

phone (that is, the Twitter client). I thought these kind of things affected only ignorant and mortal users like myself.

That is a kind of battle much more direct and related to the Linux community. Nobody else has an interest in giving users that kind of control (data access “accidents” are worth millions). Moreover, I think it would be healthy for a community where real-men-use-vim-and-a-zippo-lighter (aka GUIs-are-for-fussies) start thinking about simpler ways to give that kind of low-level control—say, an IASADCFG button (as in “I’m abroad, shut all data communication for god’s sake”).

--
Daniele Branchini

Love sc

As always, *Linux Journal* is great. I’ve read every issue since the earliest issues, and I can’t thank you enough for the articles. I was very pleased to find an article on `sc` [see Serge Hallyn’s “`sc`: the Venerable Spreadsheet Calculator” in the October 2010 issue]. There was an article in 1998 that used `sc` as an example of free software that is easy to embed software in [see “Perl Embedding” by John Quillan in the November 1998 issue]. I used `sc` and Linux to do huge spreadsheet calculations for a Master’s degree project. The other operating systems and spreadsheets kept running out of memory and resources. I use `sc` to make and review hundreds of spreadsheets, sometimes in only a few hours. Here’s a code snippet of mine:

```
psc -d , < /w/Top100Stocks.csv > /w/Top100Stocks.sc
echo "format A 8 0 0" >> /w/Top100Stocks.sc
echo "format C 7 0 0" >> /w/Top100Stocks.sc
echo "format G 12 0 0" >> /w/Top100Stocks.sc
echo "leftstring A1 = \"q\"" >> /w/Top100Stocks.sc
echo "leftstring B1 = \"Corporation\"" >> /w/Top100Stocks.sc
echo "leftstring C1 = \"Alpha\"" >> /w/Top100Stocks.sc
echo "leftstring D1 = \"Last\"" >> /w/Top100Stocks.sc
echo "leftstring E1 = \"RStr%\"" >> /w/Top100Stocks.sc
echo "leftstring F1 = \"HVol%\"" >> /w/Top100Stocks.sc
echo "leftstring G1 = \"AvgVol\"" >> /w/Top100Stocks.sc
echo "leftstring H1 = \"YTD%\"" >> /w/Top100Stocks.sc
echo "leftstring I1 = \"52WH\"" >> /w/Top100Stocks.sc
echo "leftstring J1 = \"52WL\"" >> /w/Top100Stocks.sc
...
```

Now, this brings me to the main reason I’m typing this up. The article on `sc` included some Python code, which is great and very useful. But, `psc` can be used to put a `.sc` into `.sc` form (as shown above)—`psc` comes with `sc`. Just do `man psc` for more information. Again, thanks for the great article; obviously, `sc` is one of my favorite apps. I remember kibbitzing with members of the local DCLUG, and they very much agreed it was one of their favorites too.

--
Brian G. Powell

Doc Searls replies: *That's a great idea, Daniele. I still think opening up mobile data markets is a worthy effort. The Internet is a rising tide that lifts all economic boats. I believe mobile-phone companies can make a lot more money in the open market than by punishing user mistakes in their closed ones.*

A Tribute to Minimalism

When it comes to Linux, although I've been using it only for about five years (I'm 14), I'm already sure of my taste in minimalism when it comes to computing. I'm a firm believer in minimalism, with Tiny-core Linux as my primary distribution. I seem to find any other OS wildly bloated. I would be interested to see more written about it in a world where everything in computers seems to be getting ever more complicated. There is still something to be said for keeping it simple.

--
Jed

Indeed, there is something to be said for keeping it simple! For many of us a command line is all we need. On many days, I manage my network remotely from my Droid. If you ever meet Kyle Rankin, you'll have to ask him about fixing his data center from the top of a ski slope. Anyway, thanks for the reminder to cover the small things. We'll do our best.—Ed.

Help a Pain-in-the-Neck Beginner

I am the pain-in-the-neck subscriber who is always asking for more articles for beginners. If there were three levels of Linux users—Beginner, Intermediate and Advanced—I would characterize myself as between Beginner and Intermediate but closer to Beginner. With that in mind, I have a couple requests for articles (information) in the magazine.

I am a Netflix subscriber (for almost ten years now). However, because I use the Ubuntu Linux operating system on my computer (v. 10.04), I cannot watch Netflix on Demand on it. Netflix evidently has no concrete plans to allow this.

Has anyone figured out a way to watch Netflix movies on his/her Ubuntu computer? And, if so, would that person be willing to write an article giving step-by-step instructions suitable for

Mutt Convert

I'm writing to say that I very much enjoyed Kyle Rankin's October 2010 Hack and / column "Take Mutt for a Walk". I've looked at using mutt many times in the past, but I just couldn't overcome the easiness of continuing to use Icedove. Kyle's article finally gave me the kick in the pants I needed to force myself to use mutt seriously—particularly the listing that showed examples of send hooks and reply hooks, so a huge thank you to him for that!

I would like to point out to Kyle that he is partially incorrect in stating that mutt requires a local mail server to function. Recent versions of mutt include basic MTA capabilities, allowing it to speak directly with a remote mail server (smarthost). This is, in fact, how I am using mutt currently. To use a remote SMTP server, simply set the smtp_url variable like so:

```
set smtp_url="smtps://rekkanoryo@rekkanoryo.org:465/"
```

Of course, the URL will need to be modified appropriately for different environments. Hopefully, this information will be useful to my fellow readers.

--
John Bailey

Kyle Rankin replies: *Wow, did you hear that Mr Childers? Mutt has direct SMTP support now. They must have slipped that in when I wasn't paying attention. Thanks for that tip; I'm sure other beginning mutt users will appreciate not having to set up a local mail server just to try it out.*

Neat Shell Scripts

Regarding Dave Taylor's shell column "Function Return Codes and Daylight Calculations" in the October 2010 issue, the article compares the current hour and minute with the local sunset time and uses three tests for this. And, Dave calls it "kind of neat" to boot.

That's not very neat. To compare two times, you really should convert to the lowest unit and do one test. So here, you would do:

```
currenttime=$(( $hour * 60 + $min ))
sunrisetime=$(( $srh * 60 + $srm ))
sunsettime=$(( $ssh * 60 + $ssm ))
```

And the test is:

```
if [ $currenttime -ge $ sunrisetime -a $currenttime -le $sunsettime ];
then
  # daytime ...
fi
```

Isn't that easier to understand and more maintainable? You even could replace the sunrise and sunset vars with their expressions, but that would hamper readability. Remember, people don't want neat. They want readable.

--
Frederic Mora

Dave Taylor replies: *Thanks for your note, Fred. One thing I've found that is true of shell scripts is that you always can write them more succinctly and tweak algorithms to be shorter or more efficient, but at some point, you easily can end up spending more time tweaking a script than the time you'd be saving by it existing in the first place! Oh, and I'll argue the neat (that is, elegant) versus readable topic, but that's another thing entirely.*

someone like me?

My wife and I do not use cell phones very much; consequently, we have Virgin Mobile pay-as-you-go plans. Virgin Mobile offers a pay-as-you-go Internet access, which we would like very much to have. You can buy it on an as-needed basis, so we would buy it to use when we travel.

Needless to say, because our travel computers are also operated via Ubuntu Linux, we are locked out of that too. Virgin Mobile folks say that Linux computer access is coming "in the future", but they've been making that vague promise for more than a year now.

Have any readers figured out a way to operate their Linux computer on this Virgin Mobile network, and would they be willing to write an article, again with step-by-step instructions that I could understand?

The step-by-step instructions I request mean just that. They really would have to be step by step, assuming I know next to nothing (which is really the case). I think articles like this (and these two specifically) would be of great interest to many of your readers, and they certainly would be to me.

--
Lawrence H. Bulk

Thanks for the article ideas. Your frustrations are not yours alone. Although Netflix works on embedded-Linux devices, there seems to be no support for actual Linux users. Hopefully, we'll see that change someday. As to your Virgin Mobile problem, I usually keep a copy of Windows XP in a VM (behind a NAT on my laptop), so I can authenticate to those Windows-only sites and then use the service on Ubuntu. Good luck!—Ed.

Reader Feedback

I am regular reader of *Linux Journal* and wanted to provide feedback (some a bit overdue). I very much enjoyed the article that Rick Rogers wrote in the September 2010 issue on mobile development. I followed up

with a question or two related to iPhone development and he was quite helpful. Please consider including other articles on mobile development in Linux when planning future issues of *Linux Journal* (from Rick).

Also, this is a bit overdue, but I also really enjoyed the Amateur Radio articles in the January 2010 issue. Because of this special feature, I purchased two copies off the newsstand and gave the second issue to another Ham radio operator. Possibly you could carry this theme every 14–18 months?

Like so many others, I vote with my pocketbook. Because *LJ* has issues such as the January issue featuring Amateur Radio, I purchased some of the special CDs that you offered from the *LJ* store (earlier this year).

--
Kevin

We've gotten lots of good feedback on the Amateur Radio issue, so I wouldn't be surprised to see the theme come up again in one form or another. Thanks for the positive words!—Ed.

Mutt Configuration "alternates" Line

I really liked Kyle Rankin's "Taking Mutt for a Walk" article (October 2010 issue), and I tried it out myself (with the configuration printed). However, upon starting mutt, I got the message "Error in /home/ian/.muttrc, line 17: alternates: unknown variable" (though mutt started anyway). A quick Google told me this was only with Debian and derivatives (I'm running Ubuntu) and to read /usr/share/doc/mutt/NEWS.Debian.gz. It seems that (in the repository version anyway), "alternates" has been changed from a variable to an option. Changing the line set alternates=((foo|bar)|example.org) to alternates ((foo|bar)|example.org) fixes it. Anyway, thanks for a great article!

--
Ian C.

Kyle Rankin replies: *Thanks for the detective work. I admit that my config has been in place for quite some time—probably from before that option was changed.*

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 +1-818-487-2089. E-mail us at subs@linuxjournal.com or reach us via postal mail, Linux Journal, PO Box 16476, North Hollywood, CA 91615-9911 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, PO Box 980985, Houston, TX 77098 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.



Server Solutions for the
Precision-Fit Data Center



The Next Generation of Intelligent Servers

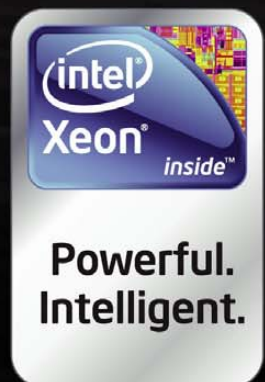
The **ZT 1250Ri Datacenter Server** powered by the **Intel® Xeon® Processor** enables unprecedented cost-effectiveness, energy efficiency and performance.



- One Intel® Xeon® Processor E5507 (dual capable)
- One 1TB 3.5" SATA Enterprise Hard Drive (two bays)
- 8GB DDR3 1333MHz ECC memory
- 400w 1U 80+ Bronze Power Supply
- Three year warranty
- Assembled and serviced in the USA

Now only **\$1345**

Because no two data centers are exactly alike, ZT Systems engineers custom server and delivery solutions to meet your unique technical and business requirements. Our precision-fit approach optimizes platforms, integration, and support for reliable, scalable solutions that deliver the efficiency and performance you need - with the expertise and personal attention you demand. Contact us today to learn more.



Intel, the Intel logo, Xeon and Xeon Inside are trademarks of Intel Corporation in the U.S. and other countries.

866-342-5984 datacentersales@ztsystems.com <http://www.ztsystems.com/inteldatacenter>

diff -u

WHAT'S NEW IN KERNEL DEVELOPMENT

Filesystem hints are attributes that filesystems can pass down to storage devices. The devices then use the hints to make decisions about how to lay down their data most efficiently. **Matthew Wilcox** has expressed interest in implementing this. Specifically, he wanted to implement the **NVMHCI working group's** recommended set of filesystem hints. But, as **James Bottomley** pointed out, the filesystem and the hardware had no real basis for agreement on what any given hint actually meant. So the filesystem would make guesses about what kind of hints to give the device, and the device would make guesses about what those hints actually meant. As James said, one of the most interesting things is that systems using filesystem hints seem to do better than those that don't, in spite of the guesswork involved. But, folks like **Alan Cox** remain unconvinced, saying he'd bet a beer on the fact that filesystem hints would end up not being used, even if they were fully implemented in the kernel. He didn't see enough benefit.

Robert P. J. Day has started offering **kernel programming classes**. See www.crashcourse.ca/wiki/index.php/Online_beginner's_kernel_programming_course for details. Some of the lessons are available for free; others are available at what seems like a pretty low fee. I haven't taken the class myself, and I'm not getting a kickback for mentioning it, but it seems like an interesting way for folks to get started with kernel hacking.

Using **kernel-level encryption** can be slow, but several folks, including **Miloslav Trmac**, recently argued that it would protect user-space applications from certain kinds of malicious attacks. Miloslav submitted a patch implementing a user-space interface to the kernel's encryption routines. This inspired a number of complaints. **Theodore Y. Ts'o** felt the speed issues would be pretty

significant, and he wanted to make sure that potential users were made well aware of the huge slowdown their code would experience if they used this API instead of a user-space implementation of the same basic feature. **Arnd Bergmann** also found Miloslav's code to be overly complex, but this was explained by the fact that so many user requests had come in for extensions to Miloslav's initial implementation. The complexity was necessary to accommodate those requests. In spite of the general complaints against this code, it does seem as though the security reasons do justify it, so none of the critics seem to be objecting too loudly. I'd expect a clean implementation to make it into the kernel.

There was a bit of a scare recently when **Linus Torvalds** received a set of patches that appeared not to have been compiled or tested at all, in spite of the long "Signed-Off-By" chain listed in the patch e-mail messages. One of the main values of the "lieutenant" system is that patches are vetted through a series of trusted people who understand what Linus wants and can give it to him. If that system ever broke down, Linus probably would have to fall back on the "maintainer" system, which would be less good, because maintainers often are selected based solely on their willingness to do that job, and not on their specific reliability as producers of Linus-worthy code. The lieutenant system, in part, helps communicate various requirements to the maintainers. In this particular case, **Len Brown** had done an incorrect merge between some ACPI branches and then fed the wrong branch of his tree into his test suite. It's a very unusual confluence of errors, but the result was that some patches made it to Linus without the proper testing—just one of those things that happens and gets fixed.

—ZACK BROWN

They Said It

Any program is only as good as it is useful.

—**Linus Torvalds**

I like to think that I've been a good manager. That fact has been very instrumental in making Linux a successful product.

—**Linus Torvalds**

Making Linux GPL'd was definitely the best thing I ever did.

—**Linus Torvalds**

Before the commercial ventures, Linux tended to be rather hard to set up, because most of the developers were motivated mainly by their own interests.

—**Linus Torvalds**

Microsoft isn't evil, they just make really crappy operating systems.

—**Linus Torvalds**

When you say "I wrote a program that crashed Windows", people just stare at you blankly and say "Hey, I got those with the system, for free."

—**Linus Torvalds**

The cyberspace earnings I get from Linux come in the format of having a Network of people that know me and trust me, and that I can depend on in return.

—**Linus Torvalds**

People enjoy the interaction on the Internet, and the feeling of belonging to a group that does something interesting: that's how some software projects are born.

—**Linus Torvalds**

Non-technical questions sometimes don't have an answer at all.

—**Linus Torvalds**

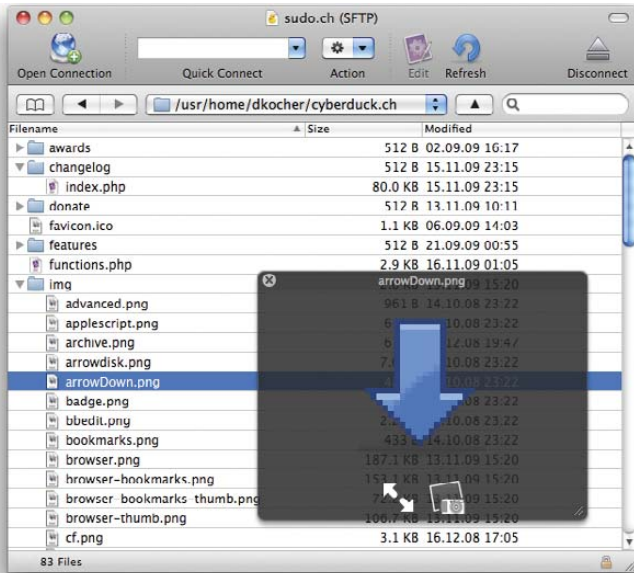
Software is like sex: it's better when it's free.

—**Linus Torvalds**

The memory management on the PowerPC can be used to frighten small children.

—**Linus Torvalds**

NON-LINUX FOSS



Cyberduck QuickLook Preview (from cyberduck.ch)

FTP/TLS, SFTP, WebDAV, Amazon S3, Amazon CloudFront, Google Docs and Rackspace Cloud Files. It integrates with external editors, file viewers, Web browsers, the system Keychain, Spotlight, Bonjour and Growl. It supports synchronizing local and remote directories with a preview of the affected files. Cyberduck can resume interrupted downloads and uploads.

Cyberduck is written mostly in Java. The latest version of Cyberduck is 3.5.1, and it requires Mac OS X 10.5 or later. For the non-Macified out there, the Cyberduck Trac roadmap refers to a port to Windows, and the repository looks like progress is being made toward that end. And, don't worry, Cyberduck can quack in your language, with more than 30 supported translations.

—MITCH FRAZIER

LinuxJournal.com

Because everything is more awesome on-line, you'll find even more great information about our 2010 Readers' Choice Awards at www.linuxjournal.com/rc10. There you'll see the runners-up and get a more in-depth look at the survey results. Save yourself some typing too, as we'll have links to all the winners and runners-up. It was a tough race, and there were some great projects and products represented, so check out all the top vote-getters at LinuxJournal.com.

You'll also notice that this issue is our 200th! That's a lot of Linux over the years, and I encourage you to get nostalgic and check out some goodies from our archives. Our May 1995 "World Wide Web" focused issue is a favorite of mine (www.linuxjournal.com/issue/13), as is a recent Linux troubleshooting series by Kyle Rankin (www.linuxjournal.com/article/10688). With so much information compiled in our 200 issues, you'll see where we've been, where we are, and where we're going in the Open Source community. Here's to issue 300!

—KATHERINE DRUCKMAN

LJ Index December 2010

1. Number of public DNS A-records for the private IP address 192.168.0.200: **63**
2. Number of public DNS A-records for the private IP address 192.168.1.200: **140**
3. Number of public DNS A-records for the private IP address 192.168.254.200: **4**
4. Number of public DNS A-records for the private IP address 192.168.255.200: **0**
5. Number of domains with names matching [a-z]200.com with valid DNS A-records: **23**
6. Number of domains with names matching [a-z]200.net with valid DNS A-records: **20**
7. Number of domains with names matching [a-z]200.org with valid DNS A-records: **13**
8. Number of domains with names matching 200.* with valid DNS A-records out of 20 possible generic TLDs (Top Level Domains): **6**
9. Number of domains with names matching 200.co.* with valid DNS A-records out of 248 possible country code TLDs: **30**
10. Number of domains with names matching [a-z]200.com that lead to "real" Web pages: **4**
11. Number of domains with names matching [a-z]200.com that lead to "real" Web pages: **6**
12. Number of domains with names matching [a-z]200.org that lead to "real" Web pages: **8**
13. Number of domains with names matching 200.* that lead to "real" Web pages: **3**
14. Number of domains with names matching 200.co.* that lead to "real" Web pages: **5**
15. Approximate number of times a byte with the value 200 occurs in the file /dev/mem: **1,900**
16. Approximate number of times a byte with the value 200 appears after reading 1 million bytes from the file /dev/urandom: **3,890**
17. Approximate number of times a byte with the value 200 appears after reading 1 billion bytes from the file /dev/urandom: **38,900**
18. Thousands of Google results for the search for "200th Issue": **381**
19. Thousands of Google results for the search for "200th Anniversary": **590**
20. Number of times the LJ staff has enjoyed creating an issue of *Linux Journal*: **200**

Sources:

- 1: www.robtex.com/ip/192.168.0.200.html
- 2: www.robtex.com/ip/192.168.1.200.html
- 3: www.robtex.com/ip/192.168.254.200.html
- 4: www.robtex.com/ip/192.168.255.200.html
- 5-9: dig/grep | 10-14: Firefox | 15-17: getchar
- 18, 19: Google | 20: 2010 Confidential LJ Staff Psychological Evaluation Report

Networking on the Command Line

Lots of GUI tools exist to help set up and maintain network connections. Two common ones are NetworkManager and wicd. But, because the focus here is doing things on the command line, how can you configure your network connections and be sure they are behaving correctly?

The first utility to cover is `ifconfig`, which lets you learn about and set all kinds of parameters for network interfaces. When you simply run it with no options, you get a list of all the network interfaces on your machine along with details for each. It looks a bit like this:

```
eth0  Link encap:Ethernet  HWaddr 00:1e:8c:71:d4:1f
      UP BROADCAST MULTICAST  MTU:1500  Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:0 errors:0 dropped:0 overruns:0 carrier:1
      collisions:0 txqueuelen:1000
      RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
      Memory:fbfc0000-fc000000

lo    Link encap:Local Loopback
      inet addr:127.0.0.1  Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING  MTU:16436  Metric:1
      RX packets:264 errors:0 dropped:0 overruns:0 frame:0
      TX packets:264 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:19232 (19.2 KB)  TX bytes:19232 (19.2 KB)

wlan0 Link encap:Ethernet  HWaddr 00:15:af:6b:59:ec
      inet addr:192.168.2.101  Bcast:192.168.2.255
      Mask:255.255.255.0
      inet6 addr: fe80::215:aff:fe6b:59ec/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
      RX packets:3228 errors:0 dropped:0 overruns:0 frame:0
      TX packets:1639 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:2994761 (2.9 MB)  TX bytes:205416 (205.4 KB)
```

You can set all kinds of options for your network interfaces with `ifconfig`. These are applied to the interface you use on the command line. In the examples below, I use `eth1`. You can set the usual things, like the netmask:

```
ifconfig eth1 netmask 255.255.255.0
```

Or, the MTU:

```
ifconfig mtu 1500
```

You can set a network device to promiscuous mode so that it receives all packets on the network, not just the ones addressed for your machine:

```
ifconfig promisc
```

Setting the address is as simple as:

```
ifconfig eth1 192.168.4.4
```

Several other more esoteric options are available, but they usually apply only to specific hardware. Check the man page for more details.

Now that you can get your network interfaces configured on the command line, you probably want to be able to have this configuration applied on each reboot. This is where the file `/etc/network/interfaces` comes in. You define each interface and whether each interface should be brought up at boot time. The most basic entry would be for a wired network interface that is using DHCP. In that case, it would look like this:

```
iface eth1 inet dhcp
```

Be sure to replace `eth1` with the label for the specific interface you want to configure. If your interface is static, you can set the address, network, netmask and broadcast values. If you want this interface to come up automatically at boot time, simply add `auto eth1` to the interfaces file. A full example looks like this:

```
auto eth1
iface eth1 inet static
    address 192.168.2.34
    network 192.168.2.0
    netmask 255.255.255.0
    broadcast 192.168.2.255
```

More options are involved when you want to configure a wireless network interface. These extra options begin with `wireless-`. In these cases, you probably want to set the SSID of the wireless network to which you actually want to connect. Also, if you need any security settings to make your connection, you also can add them here by using the option `wireless-key xxxxxxxx`. Here's a basic example, consisting of an unsecured Wi-Fi connection using DHCP:

```
iface wlan0 inet dhcp
    wireless-ssid "mynetwork"
```

If you are a bit more safety-conscious and have chosen to use WPA, you can enter your credentials with `wpa-` options. This tells the network subsystem to start up `wpa_supplicant` in the background to handle these parts. A simple example looks like this:

```
iface wlan1 inet dhcp
    wpa-ssid mynetwork
    wpa-psk mysecretphrase
```

More complex examples, like those using EAP-TLS, can use an external configuration file to handle authentication, for example:

```
auto wlan0
iface wlan0 inet dhcp
    wpa-conf /etc/wpa-supPLICANT/wpa-supPLICANT.conf
```

In the above example, all the extra parameters needed to connect are located in the named file. For more details on what you

can put in this file, see the man page for `wpa_supplicant.conf`.

Once you have all this configured, how can you activate and deactivate the various network interfaces? Use the utilities `ifup` and `ifdown`. These use the network interfaces you defined in the file `/etc/network/interfaces`. When you want to bring up a particular interface, it's as simple as `ifup eth1`. To bring down an interface, do `ifdown wlan0`. If you aren't sure whether a particular interface is up, use `ifstatus eth0` to check the status of the network interface `eth0`. If you get your IP address through DHCP, you may need to renew your lease, which you can do with `ifrenew eth1`. This accomplishes the DHCP renewal without actually cycling your network interface. Now you should be able to get your network up and running without having to resort to a GUI at all.

—JOEY BERNARD

JUST KEEP ROLLING



Linux Mint recently came out with a version of its operating system based on Debian rather than Ubuntu. For the life of me, I couldn't see the advantage over the Ubuntu-

based version. Then, in an IM chat with *Linux Journal* reader "Topher", I finally understood. Rolling releases.

That may not sound significant, but if you are (or ever have been) a Debian user, it's possible you use the "testing" release of the distribution. It's been so many years since I've been a Debian user, I forgot about the beauty of the concept. See, when the Debian folks decide to make a "release" of their distribution, they'll take a snapshot of the "testing" branch and stabilize it from there. The testing branch continues to roll along, never getting finished, and yet never getting long in the tooth.

So although the idea of a rolling distribution isn't new by any means, if you've been lulled into the Ubuntu release schedule but hate upgrading every six months, perhaps a flavor of Linux that is always up to date will appeal to you. If you don't like it, you always can update to something else!

—SHAWN POWERS

Power Supplies

One hundred forty-seven dollars and thirty-nine cents—that is the cost for replacing a power supply for an old MiniITX computer system I found in my office. Mind you, the entire unit cost about \$199, and that was five years ago, but still, the cost for a replacement power supply is absurd.

Thankfully, a quick search on the Internet found a universal power adapter that fit my requirements for about \$18. How can you find inexpensive replacements for your missing power supplies, without frying your vintage arcade cabinet computer? There are a few important things to watch for:

1. Voltage: most universal power adapters have several voltage selectors; make sure that they match your needs. (For example, laptop power supplies generally require more voltage and, unfortunately, are more expensive.) The device should say near the power adapter how much voltage it requires. The network switch in Figure 1 shows a need for 7.5V of DC current. Some devices require AC voltage as opposed to DC, so be sure to look for "DC" on the device.
2. Amperage: your device generally will say near its power port the amperage it requires along with the voltage. Amperage is a little different from voltage, and you want to make sure your power supply supplies *at least* as much amperage (usually measured in milliamps) as your device requires. The device will draw as much amperage as it requires, but there's no concern if the power supply gives more than it requires. The network switch in Figure 1 shows a 1 amp minimum requirement.
3. Polarity: your device most likely will have a drawing that shows whether the tip of the plug is positive and the jacket negative, or vice versa. Most universal power supplies will have a selector that looks similar. Make sure polarity is lined up! It's just like putting batteries in backward if you flip the polarity.
4. Plug: I wish there were a standard for the various types of plugs you might face, but sadly, there's not. Most universal adapters come with a selection



Figure 1. This network switch shows a need for 7.5V of DC current.

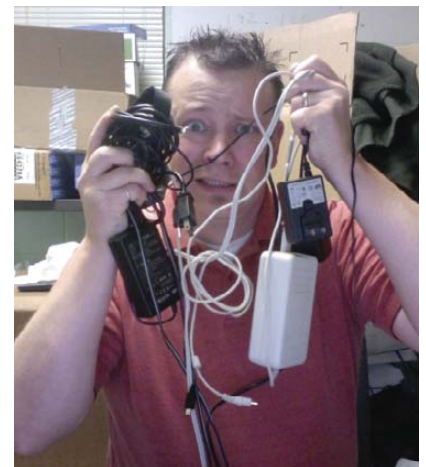


Figure 2. Associate Editor Shawn Powers wrangles with his power supplies.

of plugs that will fit most devices, but unfortunately, not all. It is possible, if you feel a bit adventurous, to cut the end off your old power adapter (assuming you have it) and solder or tape the correct plug on the wire of the universal adapter. Be warned, however, that it's easy to mess up polarity when you do that.

There are some other factors to weigh in as well. Some cheap universal power supplies are not regulated, which means they can vary in voltage depending on the load they're put under. If your device is particularly sensitive, you may want to watch for that. In the end, if you're worried you might mess up and ruin your prized powerless device, you always can shell out the \$147.39 and get a new one. For me though, \$18 was more in my budget.

—SHAWN POWERS

200 Things to Do with Linux

For our 200th issue of *Linux Journal*, we did a virtual “man on the street” interview with our Web site readers, asking what things they do with Linux. Many of the responses were rather lengthy, but we’ve trimmed them down and added some of our own.

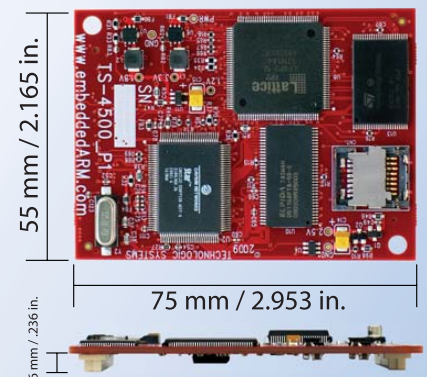
1. Actually work instead of waiting for reboots.—Tim Chase
2. Add extra monitors.—LJ Staff
3. Analyse water level and precipitation data.—Keith Nunn
4. Analysis of remote sensing imagery.—Micha Silver
5. Antagonize Windows users.—John Abbott
6. Anything I need, since 1994.—Manuel Trujillo
7. As the basis for FOSS conferences.—moose
8. Audio chat.—LJ Staff
9. Automate tasks with bash.—Dusty Roberson
10. Avoid using Microsoft Windows!—Simon Quantrill, Chris Szilagyi
11. Be a freelance writer.—Carl Fink
12. Be part of a revolution.—max
13. Be part of the Linux community.—Clifford Garwood II, Rodney Shinkfield
14. Be productive.—Petros Koutoupis
15. Block Web sites.—LJ Staff
16. Blog.—LJ Staff
17. Blow people’s minds.—djstn brimr
18. Bond Ethernet channels.—LJ Staff
19. Boot a live CD.—Tim Kissane
20. Browse the Internet virus-free.—ali
21. Bubble sort.—LJ Staff
22. Build an arcade center.—Kris Occhipinti
23. Build a robot.—LJ Staff
24. Build Asterisk telephone switches.—Mike Synnott
25. Build self-assembling/healing wireless mesh networks.—Ivan Ivanov
26. Build smart appliances.—Tom Gilley
27. Build solutions.—Wilhem Gonzalez
28. Burn CDs and DVDs.—LJ Staff
29. Carry it in my pocket.—Sean Pratz
30. cat stuff to /dev/audio.—Michael Hadam
31. Check e-mail from the command line.—LJ Staff
32. Code, code and code.—Jeff Boschee
33. Combine the power of xargs and MPlayer.—Javier Rojas Balderrama
34. Communicate with other consciences.—Angela Kahealani
35. Compile a kernel.—LJ Staff
36. Compile Windows programs.—LJ Staff
37. Compose music.—LJ Staff
38. Compress data.—LJ Staff
39. Conduct penetration testing.—Anthony Moore
40. Control embedded systems.—Mike Lerley
41. Control my data.—Dieter Plaetinck
42. Control servers from my N900.—Gunder johansen
43. Control space ground network for satellite communications.—Vidar Tyldum
44. Control XBMC from another room and freak out your kids by changing the video that’s playing.—LJ Staff
45. Convert units of measure.—LJ Staff
46. Convert video.—LJ Staff
47. Create and edit videos.—Elmer Perry
48. Create your own PBX.—LJ Staff
49. Customize with compiz.—okiwan
50. Debug ncurses code.—Alexander Cox
51. Delete all the GPS location data from images.—Stuart
52. Dent.—LJ Staff
53. Develop Arduino gadgets.—Eric Schug
54. Do development work for the pike language.—Lance Dillon
55. Do multilingual work.—Jonathan Abolins
56. Download back episodes.—john bosco
57. Dual-boot.—LJ Staff
58. Edit photographs.—Tarek Ahmed, Jim Peterson, DANiel Asselin
59. Edit the programing environment.—bhanupriya jena
60. Enjoy 1,000 days of uptime!—Ted Behling
61. Everyday tasks.—Patrick Dunn
62. Everything.—Philippe Godin, Lucas Westermann
63. Explore all the open-source apps.—Magesh
64. Explore source code.—Yash Datta
65. Explore various tools.—Bhupesh Chawda
66. Explore what Linux is made of.—Sriharsha
67. Feel the freedom.—hasintha, Risman
68. Filter spam.—LJ Staf
69. Fix Windows machines.—Scott Boucher, Detron Phillips, Stan Hearn
70. Geocache.—Buster Stone
71. Gloat when colleagues reboot Windows.—Kanwar Plaha
72. Grep the heck out of everything!—mixtape
73. Hack a Gibson.—LJ Staff
74. Hack an e-book reader.—LJ Staff
75. Hack everything.—Bart Friederichs
76. Hack your phone.—LJ Staff
77. Hang around various IRC networks.—dewey
78. Hijack Facebook on my wife.—Jon Elofson
79. Home music studio.—David Trombly
80. Home server.—Eric Gamache
81. Host your own blog.—BaloneyGeek
82. Impress girls with the command line.—Tim Kissane
83. Install apps from terminal.—M. Taylor
84. Install a RADIUS server.—LJ Staff
85. Install Boxee.—LJ Staff
86. Install on exotic hardware.—Jed Dale
87. Instant message/chat.—Josh
88. Launch a (USB) missile.—LJ Staff
89. Learn.—Andrew Frame
90. Learn C, C++, PHP, Python, Tcl/Tk, etc.—LJ Staff
91. Learn new technologies.—cga
92. Learn operating systems.—Alex Link
93. Link VHF radios using Internet.—Gustavo Conrad
94. Listen to music.—LJ Staff
95. Listen to podcasts.—LJ Staff
96. Load balance with round-robin DNS.—LJ Staff
97. Log on to Windows and remove IE.—Kartik Mistry
98. Make affordable technology solutions.—nettie feldman
99. Make a living.—Doug Roberts, cbleslie, Woody
100. Make free phone calls.—LJ Staff
101. Make my terminal window transparent.—Josiah Ritchie
102. Make non-Linux users jealous.—T.J. Domingue
103. Make videos of my desktop.—Praveen Kumar Singh
104. Make your computer look like Windows or OS X.—LJ Staff
105. Manipulate data with Python and shell.—Darrell Collins
106. Multitask.—Samuel Huang
107. Not waste my time rebuilding systems.—Jim Wallace
108. Parse weather data.—Xiao Haozi
109. Partition and format my hard drive.—Samsuddin Wira
110. Pay my bills securely on-line.—J. E. Aneiros
111. Photo management system with digiKam.—Fri13
112. Play a game.—LJ Staff
113. Play *Commander Keen*.—Terry Letsche
114. Play console emulators.—LJ Staff
115. Play SCummVM games.—LJ Staff

116. Play with Compiz Fusion.—Oleg Shmelyov
117. Play with OSes in VirtualBox.—Kousik Maiti
118. Pretend to be a Windows server.—LJ Staff
119. Provide services for Windows.—Gene Liverman
120. Proxy through SSH tunnel.—Scott Schafer
121. PXE boot GeeXboX.—Jeremy Kepler
122. Read a book.—LJ Staff
123. Read comics.—Neal Murphy
124. Read the boot sequence.—José Filipe
125. Read the digital edition of *Linux Journal*.—John Abbott
126. Record and watch TV.—Cory Lievers
127. Record, edit and publish a podcast about Linux.—Larry Bushey
128. Record HDTV with MythTV.—David Miller
129. Recover my girlfriend's data.—Arun SAG
130. Rejuvenate a sluggish computer.—Andrea Zygmunt
131. Render fractals.—LJ Staff
132. Render video content.—Erin Bournival
133. Research and analyze baseball.—Sid Finch
134. Revolutionize healthcare.—Fred Trotter
135. Rip audio from streaming radio.—Galen Gish
136. Rip YouTube videos.—LJ Staff
137. Root around a Windows computer.—Ben Pratt
138. Run a beer fermentation cooler.—LJ Staff
139. Run a feature-rich Web site with Drupal.—Jim Caruso
140. Run an embedded server (where Windows failed).—Ryan Kirkpatrick
141. Run a proxy for my friend in China.—DavidWC
142. Run Lotus Notes version 8.—David Vasta
143. Run mutt and irssi in a screen session.—Matthew Cengia
144. Run my home family network.—Zak_Neutron
145. Run my whole house.—Robert White
146. Run Radiance daylight simulations in Amazon's EC.—Severn Clay-Youman
147. Run the sound system at the chapel I attend.—Irving Risch
148. Run Windows in VirtualBox.—Happy Hacker
149. Run XBMC on your TV.—LJ Staff
150. Run Xen hypervisor.—Joe Cortes
151. Save infected Windows machines.—Paul Bucalo
152. Save people's info with Linux.—Lee Schmid
153. Search for aliens.—LJ Staff
154. Search for Mersenne Primes.—Ted Behling
155. Serve a Web page.—LJ Staff
156. Set up a distro mirror.—LJ Staff
157. Set up a VPN.—LJ Staff
158. Set up my system for perfect productivity.—Justin Christian
159. Set up MythTV.—Patrick Bulteel
160. Share Linux with other people.—Rob Haag
161. Shell scripts.—Hieu, Nghiem Ba
162. Show it to my friends.—Dale Rooney
163. Show off my desktop.—Sum Yung Gai
164. Show people cool software.—Rob Hooft
165. Sniff packets.—LJ Staff
166. Solve for Pi (okay, probably not).—LJ Staff
167. Sort your DVD library.—LJ Staff
168. ssh to remote systems.—Bharathi Subramanian
169. Stream Netflix via Roku.—LJ Staff
170. Surf the Web, text, play silly games on my Motorola Droid!—Todd Blake
171. Talk to Amateur radio operators.—Jeff Hanscom
172. Teach Linux.—shrinivasan
173. Teach operating system concepts.—satyaakam goswami, Esteban Arias
174. Time your tea steeping.—LJ Staff
175. tracepath/traceroute.—Gjorgji Taskovski
176. Transmit audio casts.—carlos gomes
177. Try as many different distros as possible.—Carlo van Rijswijk
178. Try interesting apps.—Abhishek Tiwary
179. Tweet.—LJ Staff
180. Type top and press Enter.—Roshan Baladharvi
181. Use a 9+ year-old computer.—Gumnos
182. Use GnuCash.—Peter Anderton
183. Use Linux as a thin-client server.—Tim Strickland
184. Use Linux to fix computers.—Bob Ivie
185. Use multiple virtual desktops.—LJ Staff
186. Video chat.—LJ Staff
187. Watch HD movies.—Vangelis Nonas
188. Watch *Linux Journal* videos!—LJ Staff
189. Watch TV with MythBuntu.—Todd Fowler
190. Watch video RSS with Miro.—David Crews
191. Web hosting.—Jared Moore
192. We like to have it with some funk!—Hedda, Anna and Maxim
193. Wiggle windows with Compiz.—LJ Staff
194. Work mobile or static.—Divakar Ramachandran
195. Work on my Web site.—charles snider
196. Write poetry in shell scripts.—Hani Saigh
197. Write programs.—tlylinux
198. Write Python code.—svaksha
199. Write Web pages that Internet Explorer can't display.—LJ Staff
200. Write with OpenOffice.org.—Jeremy LaCroix

TS-SOCKET Macrocontrollers


Jump Start Your Embedded Design

Series starts at **\$92** qty100



TS-SOCKET Macrocontrollers are CPU core modules that securely connect to a baseboard using the TS-SOCKET dual connector standard with common pin-out interface. COTS baseboards are available or design your own baseboard for a custom solution with drastically reduced design time and complexity. Start your embedded system around a TS-SOCKET Macrocontroller to reduce your overall project risk and accelerate time to market.

- TS-4200: Atmel ARM9 with super low power
- TS-4300: Cavium ARM11 with dual 600 MHz and FPU
- TS-4500: Cavium ARM9 at very low cost
- TS-4700: Marvell PXA168 with video and 1.2 GHz CPU
- TS-4800: Freescale iMX515 with video and 800 MHz CPU
- Several COTS baseboards for evaluation & development

 Design your solution with one of our engineers

- Over 25 years in business
- Never discontinued a product
- Engineers on Tech Support
- Open Source Vision
- Custom baseboards w/ excellent pricing and turn-around time
- Most products ship next day



We use our stuff.

visit our TS-7800 powered website at

www.embeddedARM.com

(480) 837-5200

Thanks *LJ* Subscribers!

We asked *LJ* subscribers to write in and tell us a bit about themselves, so we could print a special thanks to them in our 200th issue. So many people responded, and we wish we could include them all. We edited down the responses and chose a few to print here, and we hope you enjoy this brief glimpse at some of the folks who've helped keep us going all these years.



Guillermo Giménez de Castro, subscriber 14 years

I think *LJ* captured the real Linux spirit—a bit of “entrepreneur”, a bit of “amateurism”, a lot of joy for innovation, challenge, freedom and companionship. That’s the only reason to continue reading the magazine. I am a scientist working on Solar Physics, teaching at undergraduate and graduate levels, and now that Moore’s Law has a shallow slope, I’m developing parallel systems to take advantage of multicore technologies.



Jonathan Coker, subscriber 12 years

Reading *LJ* is like listening to a group of people discussing a subject that they really love. I do not feel like I am getting a bunch of information shoved at me like a sales pitch. It’s more like casual conversation and I am just being a good listener.

I am an EE by degree but have been writing software for 20+ years using FORTRAN, Ada, C, C++, ASP, PHP, MySQL, etc. I started at a large company then went into the startup/consulting world for about seven years and am back at a larger organization. I have been using Linux since 1995. Here is the story of my entry into “the penguin zone”: I left a big company to work at a startup and moved from using SGI Irix to Sun systems. I was still using Win98 at home but wanted to learn more about *nix internals, admin, etc. A co-worker said “get a stack of diskettes and come with me.” 30+ floppies later, I had Slackware and X ready to load on my system at home.



David Pease, subscriber since issue #2

Although I had already been using Linux for a while before *LJ* started, I didn’t hear about the new magazine until the first issue was sold out. My first “proper” issue was #2, but the *LJ* staff was kind

enough to photocopy issue #1 for me so I would not miss any issues.

I started playing with Linux using a set of Yggdrasil disks with a 0.99 kernel in 1993. By the way, I just looked at my #1 issue and was amused to see articles on the kernel 1.0 code freeze (written by Linus) and on Linux vs. Windows NT and OS/2 on the front page.

Over the years (16+), *Linux Journal* has helped me stay abreast of developments in the Linux community, and at times has helped to keep up my enthusiasm for the platform when work or school pressures forced my attention in other directions.

I have been programming computers for 40 years (just about every kind imaginable). I have a PhD in Computer Engineering from UC Santa Cruz, where I am currently an Adjunct Professor in the CE department, and I am a researcher and manager at IBM’s Almaden Research Center in San Jose, California. I am also pleased to say that I started the first Linux-based research project at IBM in 1996, several years before IBM officially embraced the Linux platform. Some years ago, I converted all of my systems (work and home, servers and laptops) to Linux (and am Microsoft-free!).



CJ Fearnely, subscriber since issue #1

I’ve been a subscriber since issue #1 and the Phil Hughes days. I missed a few issues during a transition in 2001 or so, but I still have copies of almost every issue of *LJ* ever published!

I was an early leader in the adoption and implementation of Linux and free and open-source software in Philadelphia. Through my leadership position in the Philadelphia Area Computer Society (PACS), I began introducing Linux to organizations in the Greater Philadelphia region. At PACS, I organized monthly presentations on Linux and FOSS and wrote 29 columns in the organization’s print periodical, *The Databus*. I then

founded and helped build Philadelphia’s premiere Linux user group, the Philadelphia-area Linux User Group (PLUG), where I continue to facilitate its first Wednesday meetings. After helping establish a community and culture for Linux and FOSS in Philadelphia, I started building my first company, LinuxForce (www.LinuxForce.net), to be the “go-to” firm for organizations wanting to realize the promise and power of Linux. I contribute to a blog on managing FOSS for business results (blog.RemoteResponder.net).



Marianne Popp, subscriber 3 years

When I first subscribed to *LJ*, it was because I was a new Linux user and needed help. *LJ* still helps. It gives me ideas of things I could/can use Linux for, shows me how to build or add on to my original Linux, and it also shows me how to use new programs that come out for Linux distros.

I’ve been in the fix-and-repair computer business for about 20 years. I started out with Microsoft/DOS. I was a business guru, and then went for my certifications and began building computers for a company that made special-order hardware and software. Then, I moved into freelancing with the fix-and-repair part. After that, I moved into a large school district where I took care of the networking, computer repair and software replacement at one of the high schools. These days, I’m retired, but I still do some work for friends and family.



Marco Ciampa, subscriber 10 years

The things I like best about *LJ* are kernel corner (now diff -u), hardware hacks/projects and Web technology. I think a “how to contribute to Linux/FLOSS” and a “LUG/Community corner” are missing.

I live in Trento (Italy) and co-founded

the local LUG in May 1998. I am active as a FLOSS promoter, and I contribute in the form of translations from English to Italian. I'm actually the main Italian GIMP translator (program and user manual).



Tray Torrence,
subscriber 3 years

The breadth of content helps *LJ* appeal to everyone from myself and my fellow systems administrators

to typical home users, and others in between. In my experience, it has the most balanced content of the Linux magazines.

As of last month, I am a Jr Systems Administrator in the Bay Area (recently moved from Baton Rouge, Louisiana), and enjoy various projects, from mild programming/"hacking", to hardware modifications. In my spare time, I enjoy reading, canoeing and camping.



Nathan Stowe,
subscriber 10 years

I have been an *LJ* subscriber on and off since the early 1990s, depending on my work and leisure interests

at the time. *LJ* is the *only* Linux mag that seems to address anyone other than a complete newbie. I especially like the hardware Linux articles that come out every so often.

I am a hard-core DIY'er, mechanical engineer with a slant toward software/automation. A recent accomplishment is converting an RX7 to an all-electric vehicle (www.mysmartev.com)—no Linux in there yet, but there will be if I get around to making my carputer.



Garrett Nievin,
subscriber 16 years

LJ's interesting, fun and treats me with respect, as in: I'm part of it, not just a consumer. I'm smart, but

there's a ton I don't know. I had a part-time job writing educational physics software in high school (1970s) on an Apple II. I joined the Air Force, where I became a system programmer on Sperry Univac systems; the military wasn't afraid to give amazing responsibilities to young punks. I went full-time Linux at home in 1994 with Yggdrasil, started subscribing with issue #3 and got the back issues for 1 and 2. Became a full-time Unix/Linux admin in

the mid-1990s.

I love airplanes, aviation and airplane people. I'm a full-time volunteer on AirVenture Oshkosh flightline. I own and fly an award-winning restored 1946 Luscombe Silvaire. There's no technology in the cockpit whatsoever—it's got one more instrument than the minimum required by law, and I've never used that one.



Stefano Canepa,
subscriber 12 years

I subscribed because I needed to read something about GNU/Linux. There were no magazines in Italy at the

time, so I decided to subscribe to yours. I used it to learn some English too. I must confess that I don't read every issue from cover to cover but most of them. What I find most interesting are your articles about security and programming.

I'm a software developer currently working on multiplatform projects on GNU/Linux. I'm struggling to find enough time to contribute to Debian and to some other free software. I'm a member of a local association that aims to spread knowledge about free software, free data formats and digital rights. With my association, I'm involved in a project to spread the use of GNU/Linux at primary schools. Early in the morning, I like to run just to relax my mind and keep well-trained, as the Latins said: *Mens sana in corpore sano*.



Peter Teuben,
subscriber 16 years

I like a monthly magazine, because despite reading a lot on the Internet, *LJ* always surprises me with

things I have missed. It's great to read away from the keyboard and think about which things to try before you're too close to the keyboard and jump on it. I've been a subscriber since the stapled issue #1 and still have them all, and it's great fun to flip back some pages from time to time.

I do research and programming in astronomy, and we have pretty much 100% migrated to Linux during the past 10 years. I've built a number of boxes and RAID arrays, even tinkered with real-time Linux for an instrument, but for programming work, I have been a laptop user for the longest time.



Sean Kirkpatrick,
subscriber 14 years

I don't really know how long I've been a subscriber, but it'll take more than your disastrous format

change a couple of years ago to get me to unsubscribe. Timely, relevant, interesting, challenging, cutting-edge, it's helped me sell GNU/Linux (I wish you'd use that term, honoring the other, equally important piece of the OS) to my boss and slash our licensing costs.

I've been a programmer since 1984, network administrator and junior college instructor (I wrote the original GNU/Linux curriculum years ago). I started with the MCC distro WWWAAAAAYYYY back when.



Ryan Chiles,
subscriber 6 years

I am a longtime Linux supporter. I don't always have time to hack around with Linux, as I am a

Windows application developer at my job. So *LJ* helps me stay in touch with the Linux/Open Source movement when I get too involved with the Windows world of thinking.

I have been using Linux/open-source software since 1997. It was first introduced to me in college, and I was hooked instantly and still use it today. I am an applications developer in Visual Studios in ASP.NET and C#. I like to hack in my spare time in Python and PHP. I do Web site designs focusing on usability and experience. When I am not in front of a monitor, I am a volunteer firefighter. I love this as much as I do my Linux.



Mark Rustad,
subscriber 12 years

I get good introductions to various topics from *LJ* that I may not know much about and sometimes pick up

useful techniques. I tend to focus a lot on the kernel and drivers, and it is good for me to be aware of more applications and their uses.

I have been programming since 1970. My first paid programming job was in the summer of 1971 at the University of Illinois on the PLATO Project when I was in high school. I have pretty much been programming ever since. I still remember reading

Linus' e-mail about Linux in the Minix newsgroup while I was working for Apple and thinking, "Those guys are going to have a lot of fun. I wish I had time to help." Between work and family, I really didn't have free time to get involved. I also thought that Linux would never really be important. Boy was I wrong about that! My first exposure to Linux came in 1996 when I worked for an employer that used Linux to host the company's main server that also served as the company's main Internet connection. Initially, it was only dial-up access, but by 1997, it was upgraded to ISDN. I don't remember the kernel version it was running, but that server seemed more stable than the ISP it was connecting to. Still programming after all these years, and mainly using Linux and Mac OS X.



Joe Klemmer, subscriber 16 years

I have been involved with Linux since November 1991. The idea of a magazine dedicated to it was, at the time, incredible. It was as exciting as when the first edition of *Running Linux* was published.

I was born in Germany to American parents and grew up living between the US and Europe. I worked 28 years in the IS/IT field. I'm divorced and have one son. I love programming and am fair to ok at it. Physically disabled since '93, and going to university on-line to get a BS degree in programming.



Chris Cox, subscriber 16 years

I subscribe because I believe in the ability of open source and free software to transform the computing world. And, thanks to publications like *Linux Journal*, it is and will continue to do so.

I started using UNIX with Ultrix back in 1983. My first software contributions were made to assembler language programs written for CP/M and were published on RBBS (a popular bulletin-board system). I programmed in everything from Turbo Pascal, to C, LISP and even Cobol. I have many years of experience as a software release manager. I have been an adjunct professor teaching UNIX using Linux for both continuing education and credit side colleges. I currently

serve as a Sr UNIX sysadmin specializing in Linux and Windows integration. I'm also President of the North Texas Linux Users Group. Putting Linux distributions to use in real-world situations is one of my passions.



Johannes Ramm-Ericson, subscriber 14 years

LJ is simply the first and best Linux magazine available!

Every issue still has at least one interesting feature article that includes things I don't know about Linux—and that is despite having worked professionally with Linux for 12 years and provided Linux/FOSS solutions in a very wide range of areas.

I'm 40-something years old and a dedicated father of three as well as a former track and field athlete who grew up as an international student in Vienna, Austria, and I'm now living in Sweden. I've been a Linux/UNIX system administrator (since 1996) in the Telecommunications Industry who felt extremely satisfied (and justified) when the company I work for finally made the shift to a Linux-based OS in the mobile phones it produces. My colleagues and I had been discussing this for years and seeing it finally happen was vindication at an unprecedented level! Now, of course, I have my work cut out for me since there is a huge demand for everything Linux within the company.



Stuart Powell, subscriber 9 years

My favourite aspect of *LJ* is the "Things that make you go hmmm". Sure, the tech articles are very useful, as are the product tests and comparisons. But it's highlighting all those quirky/odd/crazy/utterly bizarre things that people are out there doing, that Linux is a part of, that makes me look forward to the magazine dropping into my mailbox every month. The latest issue always has pride of place in *my* bathroom, so I can peruse it at leisure.

I guess I'm a life-long techie, having started out by learning BASIC on a Sinclair ZX80 in 1980 at the age of 8. I've essentially grown up with the personal computer industry, and have worked with so many different technologies over the years, I can't even remember

them all now. It appears that my brain is prematurely full. I started with Linux in 1993 as a way of keeping up the UNIX knowledge and skills I had acquired while working in Germany. My first distro was Yggdrasil Linux, and I bought myself a video card off their compatibility list and a 2.2X CD-ROM drive so I wouldn't have to load up an insane amount of floppies. I got it all working, but it didn't really do a lot. But it was a start, and it did keep me interested, and while I was never able to make Linuxing a major part of my career, it has certainly been very useful from time to time in a minor capacity.



Desmond Daignault, subscriber 13 years

I like the variety of articles and the letters that the readers send in with their tips, etc. This is the only magazine that I read cover to cover every month.

I am a software developer that has been using Linux since 1995. I am primarily a Perl developer now, but cut my teeth on C. I have written a screensaver module for XLockMore and XScreensaver and have several projects in the works, all of which are open source.



Tony Mansson, subscriber since issue #1

I have been a subscriber since the very beginning in March 1994. I have all the issues saved, and I sometimes bring issue #1 of *LJ* to class when I hold Linux courses. People are always surprised to find an advertisement from a Swedish company in it, and even more surprised to discover that part of the text in this ad is in Swedish (!). My field is System Design and Software Architecture of mobile devices, and I have a long-term engagement with ST-Ericsson. I help asserting that the ST-Ericsson dual-core ARM-based mobile platforms become attractive for use with open-source environments such as MeeGo. I have been a Linux Evangelist just about forever, and my first distro was Trans-Ameritech 4 from 1994. The best Sunday afternoons for me are the ones when I have a fresh *LJ* to read. Other Linux magazines can be interesting too, but none feel as genuine as *LJ*.

we get how geeks think.

PATIENT MRI EXAM

TurboClocked

CPU Cores: 64

Clock Speed: 6.66 GHz

Bandwidth: 100 Gbps

Refresh Rate: 240 Hz

Storage: 32 PB

Latency: 0.002 ms

Packet Loss: 0.00%

Load Avg: 0.01

S. BEACH

GEEK, IMA

023Y MALE

1 800 7419939

23:59:59

R

L

Linux Journal Magazine Exclusive Offer*

15% OFF

Call **1.888.840.9091** | serverbeach.com

Sign up for any dedicated server at ServerBeach and get 15% off*. Use the promo code: **LJ15OFF** when ordering.

* Offer expires December 31st, 2010.

Terms and conditions:

© 2010 ServerBeach, a PEER 1 Company. Not responsible for errors or omissions in typography or photography. This is a limited time offer and is subject to change without notice. Call for details.



REUVEN M. LERNER

Issue 200

Reuven reminisces about the past of the Web, describes its present state and makes some predictions for the future.

So, *Linux Journal* has reached issue 200! As many of you know, I've been writing for this magazine for much of that time. According to my count, this is my 168th monthly column. I started back in 1996, long before I got married, became a father or began my PhD studies. It's hard to remember a time before *Linux Journal* was a standard item on my monthly calendar.

When I look back over the years, it's amazing how many things have changed when it comes to Web technologies. And yet, so many things also have remained the same. This month, I celebrate this issue of the magazine with a bit of nostalgia, reminding you where we've been and describing where we're headed. Along the way, I discuss some of the topics I intend to address in the future in this space.

The Past

At least a few readers of this column presumably remember a time when the Web and Internet weren't

The biggest thing missing from the early Web was the ability to write custom applications.

ubiquitous. My children always are amazed to hear that I was one of the only kids in my grade to have a home computer. It's hard for them to understand that when my mother told us that we should look something up, she meant we should drive to the local public library, find books (in a paper card catalog) on the subject and search through those books to find the answer. Today, the Internet in general and the Web in particular are fixtures in our daily lives. But back in 1988, just after I started college, my friends gave me a funny look when I asked them if they had Internet e-mail addresses. When we put the MIT student newspaper on the Web in 1993, we had to tell people how to install a Web browser on their computers. All of this is clearly a thing of the past. If nothing else, it's hard to find an advertisement without a URL at the bottom inviting you to learn more.

After decades of discussion and development of hypertext systems, it wasn't necessarily obvious that the World Wide Web, the brainchild of Tim Berners-Lee, would become a major hit. And yet, to those of us who used it in those early days, the Web had a

number of clear advantages over its competitors. It was easy to set up a server and site. The protocols were simple to understand, easy to implement and easy to debug (because they were text-based). The addresses were unique, easy to read and easy to write. Clarity, ease of implementation and ease of use were critical in jump-starting the Web revolution. The success of a simple, easy-to-use approach is easy to spot today as well—look no further than Twitter, LinkedIn or Facebook.

The biggest thing missing from the early Web was the ability to write custom applications. It was simple to set up a server that would make HTML (and other) files available to the general public. But it was the invention of CGI—a standard protocol that allowed HTTP servers to communicate with external programs—that made it possible for programmers to write dynamic Web applications. The idea that the Web was a new application platform was a bit hard for many of us to swallow. I remember bristling at my title, "Web application developer", when I worked at Time Warner in 1995, saying it was ridiculous to think that we were developing "real" software applications. Today, of course, Web applications have overtaken their desktop counterparts in many aspects.

The Apache Web server was one of the most important contributors to Web development in a number of ways. It was one of the first well-known open-source projects that was clearly superior to any of its commercial competitors. (Did you even know that there was once a market for commercial HTTP servers?) Apache's power and flexibility convinced many large companies that they should cooperate and communicate with, and even contribute to, open-source projects that did not compete directly with their core businesses. If I remember my history correctly, I believe it was IBM's interest in donating money to Apache's development, but the developers' lack of any formal infrastructure that could accept the money (let alone sign a contract) that led to the development of the Apache Software Foundation, one of the most prominent players in the Open Source community today.

Apache also demonstrated the advantages of modular software design. Because Apache was intended to serve many different populations, its developers created it as a set of modules, and each of which could be included or excluded from the final product, depending on the site's needs.

Finally, Apache made it possible to create custom

Web applications without having to suffer from the performance problems associated with CGI programs or from the development time associated with writing custom HTTP-enabled applications. By writing your own module (in C), you could do just about anything, attaching your custom functionality to one or more of the hooks that Apache posted during an HTTP request's life span. Eventually, it became possible to write custom applications using Perl and Python, rather than just C—and anyone who moved from CGI programs in Perl to mod_perl benefited from a tremendous increase in both speed and flexibility.

By the end of the 1990s, most people were using a relational database behind the scenes to keep track of their data, after discovering that text files just weren't fast or flexible enough to do the trick. Many applications used commercial databases, wishing that someday we could enjoy the power of SQL without having to fork over enormous amounts of money to a large corporation. And indeed, starting in the late 1990s, things began to improve, both in terms of open-source licensing and functionality. MySQL was re-issued under the GNU General Public License and started to move in the direction of ACID compliance,

and PostgreSQL began to improve its usability, shedding such issues as a laughably small maximum tuple width.

The Present

Today, it's easy to create Web applications. Almost any part of the technological infrastructure you might need—including operating systems, databases, programming languages and frameworks—is available under an open-source license. Indeed, the problem is often not a matter of finding something that will be suitable, but rather sorting through the many competing open-source projects, each of which has its own advantages and disadvantages.

Open source is now the norm and even is expected in many places. I recently spoke about Ruby on Rails at a conference for Web developers in Israel, where one of the keynotes was given by a Microsoft employee. Every other sentence he uttered talked about open-source software, getting popular open-source packages to work under Microsoft technologies, and how small- and medium-size sites can get access to Microsoft products for free, until they achieve a certain level of success. In other words, Microsoft understands that the balance is shifting to the Open Source world and is

Small, Portable Devices with Ubuntu Linux

Small Form Factor Intel® Atom™ Platform

No fans, no moving parts. Just quiet, reliable operation. Incredibly compact and full featured; no compromises.



VESA-Mountable NVIDIA® ION/ION2 System

Compact, lightweight system with GeForce® Graphics. Flexible storage options (dual HDD support) and WiFi.

Value only an **Industry Leader** can provide.

Selecting a complete, dedicated platform from Logic Supply is simple: Pre-configured systems perfect for both business & desktop use, Linux development services for greater system customization, and a wealth of online resources all within a few clicks.

[Learn More > www.logicsupply.com/linux](http://www.logicsupply.com/linux)

LOGIC
SUPPLY

competing by offering greater standards compliance and lower prices—something that open-source advocates can claim as a victory of sorts.

Modern Web development often takes place inside a “framework”, a collection of libraries that make the developer’s life easier. Some of the most popular Web frameworks are Rails (Ruby), Django (Python), Symfony (PHP) and Catalyst (Perl), although there are dozens, and maybe hundreds, of others for these languages and others. By using a framework, developers can concentrate on their specific domains, rather than re-inventing the same infrastructure multiple times.

Most of these frameworks use the MVC (model-view-controller) paradigm pioneered more than 20 years ago by languages such as Smalltalk, reflecting not only the increasing complexity and sophistication of Web applications, but also the size and diversity of the teams needed to create such an application. Keeping things separate within an MVC framework ensures that a designer will probably not step on a

As we move into the future, we’re seeing a need for functional and distributed programming, which has made languages such as Scala, Clojure and Erlang more popular.

developer’s toes during the development process. By adopting the “convention over configuration” idea pioneered by Ruby on Rails, developers also can avoid discussions, arguments and consideration of where each file should be located.

Today, the question is not whether you want to use a database for data storage, but rather which one you want to use, whether it will be relational or non-relational (“NoSQL”), and what sort of interface you will use to communicate with it. Most modern frameworks handle relational databases seamlessly, often providing you with an ORM (object-relational mapper) that allows you to ignore the fact that you’re actually using SQL to store information in two-dimensional tables. There also is growing support for non-relational databases in these Web frameworks, making it possible to choose what type of data storage is ideal for your particular application.

Not only have the frameworks changed, but the languages are starting to change too. Perl continues to be popular in some corners, and PHP still is hanging on, but the growth and action appears to be with Ruby and Python, as well as with many other newer languages. Indeed, I often say that Perl was perfectly suited to early Web applications,

because its strengths were in text manipulation, networks and databases—precisely what you need for a Web application. As applications became larger, these strengths were less important than the ability to maintain code, something for which Ruby and Python are (in my opinion) better suited.

As we move into the future, we’re seeing a need for functional and distributed programming, which has made languages such as Scala, Clojure and Erlang more popular. Scala and Clojure, although very different languages, are both built on top of the Java virtual machine (JVM), as is jRuby. The growing use of the JVM as the underlying infrastructure for a non-Java language continues to interest me, and it raises the question of what will happen to Java itself over time, as these languages become even more popular.

Perhaps the biggest surprise, to me at least, has been the growth of JavaScript during the past few years from a language that was barely used to animate some menus, to one that has led to the introduction of radically new JavaScript engines in all of the major browsers and to the creation of several high-quality, cross-platform libraries. I certainly tended to pooh-poooh JavaScript as a language. In many ways, the reason I now like working with JavaScript is because of the libraries (such as jQuery and Prototype) that insulate me from some of the problems with the language, rather than changes to the language itself.

JavaScript also continues to pop up in places other than browsers. JSON, the JavaScript object notion, has become a very popular, lightweight alternative to XML for transmitting data between computers. And Node.js, a JavaScript library for creating high-performance network and server applications using JavaScript, has begun to make serious inroads.

Once you’ve put together your application, where are you going to host it? You still can put it on a server that you own or on a fraction of a server that you rent, but cloud computing has taken hold of the industry, not only because it makes hosting so much easier, but also because it means you no longer need to hire a full IT staff to run the servers.

Finally, whereas we think of Web applications as having to do with people, the fact is that many applications are for machine-to-machine communication. The growth of various microformats, along with JSON and XML-based systems continues to rise. Moreover, the growth (and importance of) APIs has exploded during the past few years. Although it used to be a nice thing for a Web application to offer an API, it now is almost expected that everyone will offer an API, for use by desktop applications, mobile applications, aggregation systems or new uses that mix and match what already has been done.

Right now, we’re enjoying what might seem to be the best of all possible worlds: easy, cheap and

scalable hosting, programming languages and frameworks that lend themselves to rapid, maintainable development, and storage systems that are flexible, which connect seamlessly to our programming framework of choice. The main limits to creating Web applications today have more to do with skill and time than money, as we can see from the rapid growth of applications on Facebook, for example.

The Future

So, where are things going?

First, we already can see that the notion of the Web as something people browse, with large centralized servers providing static information, is largely inaccurate. People and machines are both surfing, and they are doing it with programs that are increasingly not Web browsers, but rather that contain HTTP client libraries. The servers are spread all over, and the information is far from static. Just as I was writing this column, Google announced that it had changed the way its search system works, such that it updates the page of search results as you type keywords, not just when you click the Submit button. Just as the Web is always changing, and just as each person sees a different, personalized slice of the Web, your search results now also will give you a view of data that is uniquely yours.

We also can expect to see an even greater decline in desktop software. This is actually good news for fans of Linux and other open-source operating systems, because it means there will be less of a lag between the quality, availability and user experience that Windows and Macintosh users have long enjoyed with their desktop software. The Web browser is indeed becoming, years after Marc Andreessen predicted it while working at Netscape, the main focus for application development, deployment and usage. Even those programs that aren't browsers will be browsers, connecting to the Internet and retrieving (or sending) information, exchanging data with other servers.

When the idea of Web services first became popular about a decade ago, everyone used the example of a spell-checker as a Web service to which your word processor could connect. The reason for this example was not only that it was easy to grasp, but also that we had no idea just what Web services could provide. Nowadays, such services can provide private information (such as contact info) or public information (such as maps and photos). We will continue to see growth on the Web services front, although outside the enterprise, it seems that developers have largely abandoned SOAP in favor of lighter-weight technologies.

One of the reasons Web-based applications will become so good is because of HTML5, a combination of improvements to HTML, CSS and JavaScript

that are being implemented piecemeal, but which together will make the browser far more than the "modern dumb terminal" description that often is applied to it. New form features, new ways to validate data, easier access to the DOM, new CSS selectors and features, and a greater variety of semantic markers in the HTML will make this a very important upgrade. My only worry and complaint is that each browser manufacturer is implementing different parts of HTML5 at different times, meaning we'll need to worry about graceful degradation for some time.

Future Columns

So, what do I intend to discuss in future installments of *At the Forge*? I'll certainly try to cover some of the basic technologies that are useful to Web developers, such as the recent release of Ruby on Rails 3 and the release of PostgreSQL 9.0. I'll spend some time exploring the HTML5 standard, looking both at the new tags we can enjoy in our HTML and at the improvements in JavaScript we can use in our applications.

I also intend to look into some of the newer languages that have emerged, as well as the Web frameworks built on such languages. The three languages and frameworks that intrigue me the most are Lift (for Scala), Compojure (for Clojure) and Seaside (for Smalltalk).

Storage—the non-relational databases will gain popularity. More important, they will gain features we have grown to expect in relational databases, such as joins and data integrity. The end result will be a number of different non-relational options that can be mixed and matched for an application, much as a developer might mix and match the use of arrays and hashes. Will they trump non-relational databases? I doubt it, but I'll try to cover developments from this world and how they affect developers, as things happen.

Finally, the growth of "microformats", tiny JSON- and XML-based document formats designed to ease machine-to-machine communication is something I intend to look into. How do you use a microformat and when would you want to do so?

It continues to be a privilege to write for *Linux Journal*. I enjoy hearing from readers when they contact me and helping inform fellow open-source developers of the latest on the Web technology front. And, I look forward to writing an even more comprehensive retrospective in another eight years, when we'll reach *LJ* #300. ■

Reuven M. Lerner is a longtime Web developer, architect and trainer. He is a PhD candidate in learning sciences at Northwestern University, researching the design and analysis of collaborative on-line communities. Reuven lives with his wife and three children in Modi'in, Israel.



DAVE TAYLOR

Generating Turn-by-Turn Driving Directions

Maps? We don't need no stinking maps. With bash and a few command-line utilities, we can extract the textual driving instructions from an on-line map source and dispense with all those pesky GUI maps.

I'm happy to report that this month, I'm answering a reader's question about how to script something. Dunno what's up with the rest of you readers, but apparently writing to me with your weird and challenging shell-scripting puzzles isn't making the short list right now. Reader Paul M. asks:

Is there a way to screen-scrape Google Maps direction results? I'm after the text (turn left at Ho-ho-kus Blvd), not the maps. When I look at a saved results page, all I can see is CSS and JavaScript code. If I do a manual copy and paste of the directions, however, the turn-by-turn directions appear. Got any suggestions on how to grab turn-by-turn driving directions automatically, Dave?

Ah, those tricky programmers over at Google Maps make this pretty darn difficult! Poke around at

Ah, those tricky programmers over at Google Maps make this pretty darn difficult!

the source pages generated by maps.google.com looking for directions, and it's clear that they're using a `method=post` or other advanced way to hide the starting and ending points from the URL itself, along with some very fancy coding to make the Web pages highly interactive. So to heck with it!

After much digging around and looking at how the different mapping sites work, I settled on Expedia.com as the best place to get driving directions so that we'll be able to specify start and stop points via URL and also understand the output. To get started, check out Expedia's interactive driving directions in your Web browser at www.expedia.com/Directions.

On Expedia, enter a starting and ending address for directions, and you'll find that it's all stored in a scary-complex URL like this: [\[1600+Pennsylvania+Ave+NW&city1=Washington&stnm1=DC&zipc1=20006®n1=0&labl1=1600+Pennsylvania+Ave+NW%2C%0AWashington%2C+DC+20006&lats2=28.393142483519902&lons2=-81.57198620078931&alts2=5&strt2=N+World+Dr&city2=Orlando&stnm2=FL&zipc2=32830®n2=0&labl2=World+Dr%2C%0AOrlando%2C+FL+32830&\]\(http://www.expedia.com/pub/agent.dll?qscr=mrd&rtp=0&unit=0&lats1=38.89872&lons1=-77.036379&alts1=5&strt1=1600+Pennsylvania+Ave+NW&city1=Washington&stnm1=DC&zipc1=20006®n1=0&labl1=1600+Pennsylvania+Ave+NW%2C%0AWashington%2C+DC+20006&lats2=28.393142483519902&lons2=-81.57198620078931&alts2=5&strt2=N+World+Dr&city2=Orlando&stnm2=FL&zipc2=32830®n2=0&labl2=World+Dr%2C%0AOrlando%2C+FL+32830&\) \(Eagle-eyed readers will notice that I'm offering the Obama family driving directions to Disney World.\)](http://www.expedia.com/pub/agent.dll?qscr=mrd&rtp=0&unit=0&lats1=38.89872&lons1=-77.036379&alts1=5&strt1=</p>
</div>
<div data-bbox=)

You can strip some of the superfluous information out of the URL and create a simple command-line call to get the map and directions:

```
start="strt1=1600+Pennsylvania+Ave+NW&city1=Washington&stnm1=DC"
dest="strt2=N+World+Dr&city2=Orlando&stnm2=FL&zipc2=32830"
```

```
curl --silent "http://www.expedia.com/pub/agent.dll?start&
➔$dest&qscr=mrd&rtp=0&unit=0"
```

You can see that Expedia wants an address unwrapped and split by street address, city, state and zip code (though if it can figure out the location, it appears you can skip the zip code, as shown in `start` above).

Now that we have that, let's use `sed` to extract just the table of results, without the other superfluous information. This is done by manual analysis of the source file and noting that it's all in a table that starts with this HTML line:

```
<TABLE BORDER=1 BORDERCOLOR=#E4E4E4 CELLSPACING=0 CELLPADDING=4>
```

Not surprisingly, the line we seek that denotes the end of the table is `</TABLE>`. Here's the code that lets you slice things as desired:

```
sed -n '/BORDERCOLOR=#E4E4E4/,/ </TABLE > /p'
```

Put them all together and save the output to a temp file. After that, the next challenge is to turn that HTML table into something you actually can read.

To do that, we're going to turn to a great open-source utility called Lynx. You might already have Lynx on your system, but if you don't, grab a copy of the Lynx text-based Web browser from lynx.isc.org.

We'll use that to interpret and convert the HTML markup to raw text.

Fortunately, Lynx excels at this kind of challenge, as demonstrated by the working code:

```
curl --silent "http://www.expedia.com/pub/agent.dll?start&
➤$dest&qscr=mrdr&rtp=0&unit=0"| \
sed -n '/BORDERCOLOR=#E4E4E4/,/ </TABLE>/p' | \
lynx -dump -stdin
```

Yup, that's it. Specify a correct start and destination, make sure that the script knows where to find Lynx on your system, and the output will look like this:

```
Directions Distance Time
Start: Depart Start on Local road(s) (East) 0.1 < 1min
1: Turn RIGHT (South) onto E Executive Ave NW 0.1 0:01
2: Turn LEFT (East) onto Alexander Hamilton Pl NW, then
   immediately turn RIGHT (South) onto 15th St NW 0.1 0:01
3: Turn LEFT (East) onto Pennsylvania Ave NW, then immediately
   turn RIGHT (South) onto 14th St NW 0.3 0:02
```

To do that, we're going to turn to a great open-source utility called Lynx.

```
4: Keep STRAIGHT onto US-1 [14th St NW] 1.1 0:02
...
22: Take Ramp (LEFT) onto Western Way (Disney World) 1.9 0:02
23: Turn LEFT (North) onto Bear Island Rd 2.1 0:03
24: Turn RIGHT (East) onto Floridian Way 0.3 0:01
25: Keep STRAIGHT onto World Dr 0.4 0:01
End: Arrive End < 0.1 < 1min
Total Route 881 mi 13 hrs 2 mins
```

I'll leave it as an exercise to you, dear reader, to create a wrapper that prompts people for starting and ending addresses and then uses the curl invocation to Expedia and subsequent invocation of Lynx to display turn-by-turn driving directions. ■

Dave Taylor has been hacking shell scripts for a really long time, 30 years. He's the author of the popular *Wicked Cool Shell Scripts* and can be found on Twitter as @DaveTaylor and more generally at www.DaveTaylorOnline.com.



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



**Powerful.
Intelligent.**



As the Chief Financial Officer for Silicon Mechanics, Steve is an Expert where value is concerned. That's why he's pictured here with the Rackform iServ R143.

The R143 is a flexible and affordable 1U server. It features an Intel® Xeon® Processor 3400 Series, with powerful features like Turbo Boost, Hyper-Threading, and DDR3 memory. This processor is also available in a low-voltage version, which can optimize power usage and help contain energy costs. With 6 DDR3 DIMM sockets, 2 Gigabit Ethernet adapters, a PCIe expansion slot, and 4 hot-swap SAS/SATA drive bays, the R143 can handle a lot more than entry-level workloads. With a price that starts around \$1250, you don't have to be a CFO to understand the value.

When you partner with Silicon Mechanics, you get more than a flexible, affordable entry-level server — you get an Expert like Steve.

For more information about the
Rackform iServ R143
visit www.siliconmechanics.com/R143

Expert included.

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.



MICK BAUER

Building a Transparent Firewall with Linux, Part IV

Arm your stealth firewall with a custom iptables script.

I've been writing a multipart series on building a transparent (bridging) firewall using Linux. Specifically, I'm using the distribution OpenWrt running on a Linksys WRT54GL broadband router, a hardware choice driven mainly by my curiosity about the WRT54GL's built-in five-port Ethernet switch and its ability to run OpenWrt Linux.

So far I've covered installing OpenWrt, recompiling a new OpenWrt image with iptables' bridging functionality enabled and configuring networking using OpenWrt's uci (Unified Configuration Interface) command.

This month, I review the example network topology and finally begin configuring iptables, the heart of the whole undertaking. Before I do so, however, there are a few OpenWrt housecleaning tasks to get out of the way: tweaking the kernel and network configurations, and disabling OpenWrt's native firewall system.

Kernel Parameters and a Network Tweak

Recompiling the OpenWrt image with `CONFIG_BRIDGE_NETFILTER=y` set in the Linux kernel

is the first of two steps in enabling iptables' bridging mode in OpenWrt. The second step is either to delete the following parameters in `/etc/sysctl.conf` or set each of them to "1" rather than "0":

```
net.bridge.bridge-nf-call-arptables=0
net.bridge.bridge-nf-call-ip6tables=0
net.bridge.bridge-nf-call-iptables=0
```

In addition, I need to correct an error I made in the OpenWrt network configuration I showed you last time. You may recall that I changed OpenWrt's default configuration, such that all Ethernet ports were assigned to a single VLAN and bridge.

But possibly due to the way the Linux kernel interacts with the bridge hardware on my Linksys WRT54GL, with that configuration, I find that iptables ignores inter-VLAN traffic—that is, traffic between ports on the same VLAN. In order to get iptables to work properly on this hardware and on OpenWrt, I actually need two VLANs: one corresponding to my

OpenWrt Performance as a Transparent Firewall

In researching this article, I had a nasty surprise. Although in the past I had seen articles and how-tos on making transparent firewalls with OpenWrt, this mode of operation is not supported by default in the Kamikaze and Backfire releases. Reportedly, running iptables in bridging mode under OpenWrt reduces overall system performance by a whopping 40%!

I proceeded writing this series anyhow, because I wanted to see for myself just how big an effect this is, and it seemed to me that the series still would be useful just for the sake of explaining how to install and use OpenWrt, and for explaining how to write iptables rules for transparent firewalls. However, at several points, I've written of my doubts as to the example OpenWrt/WRT54GL's suitability for high-bandwidth/high-availability settings.

Also, hopefully without sounding *too* grandiose, I hoped that

by spurring greater interest in OpenWrt's flawed capability, I might encourage someone to get to the bottom of *why* OpenWrt performance plunges when run with iptables in bridging mode. Surely there's a reason that this not terribly new kernel feature is problematic in OpenWrt!

I say all this because I want to be clear that although transparent Linux firewalls in general constitute an interesting and useful technology, the specific combination of a \$65 broadband router plus OpenWrt running in this mode is probably suitable *only* in a home or lab setting, not for any situation where you need to move large volumes of packets very quickly and very reliably (which is hopefully unnecessary for me to say, given that the WRT54GL is marketed to home users in the first place). I say it also so you understand why you have to go through the hoops of recompiling the OpenWrt image and editing `/etc/sysctl.conf` to get iptables bridging working in OpenWrt.

“uplink” (the Ethernet port connected to the outside world) and my “LAN” (everything else). These two VLANs, however, are still associated with the same bridge interface.

To create a separate VLAN for my uplink port, which is my WRT54GL’s “WAN” port (or “port 4” to OpenWrt), I issue these commands on my router:

```
root@sugartongs:/etc/config# uci set network.eth0_1=switch_vlan
root@sugartongs:/etc/config# uci set network.eth0_1.device="eth0"
root@sugartongs:/etc/config# uci set network.eth0_1.vlan="1"
root@sugartongs:/etc/config# uci set network.eth0_1.ports="4 5"
```

(Port 5, you’ll recall, is a virtual port associated with the kernel, that must be included in *all* “ports” statements in OpenWrt network configurations, which is why our “...ports” statement is set to “4 5” rather than just “4”.)

To remove the WAN port from the other VLAN (eth0_0) I set up last time, I use this command:

```
root@sugartongs:/etc/config# uci set network.eth0_0.ports="0 1 2 3 5"
```

Next, in my bridge configuration, for the network I named “lan”, I associate both VLANs with the bridge:

```
root@sugartongs:/etc/config# uci set
```

Listing 1. Corrected /etc/config/network

```
config 'switch' 'eth0'
    option 'enable' '1'

config 'switch_vlan' 'eth0_1'
    option 'device' 'eth0'
    option 'vlan' '1'
    option 'ports' '4 5'

config 'switch_vlan' 'eth0_0'
    option 'device' 'eth0'
    option 'vlan' '0'
    option 'ports' '0 1 2 3 5'

config 'interface' 'loopback'
    option 'ifname' 'lo'
    option 'proto' 'static'
    option 'ipaddr' '127.0.0.1'
    option 'netmask' '255.0.0.0'

config 'interface' 'lan'
    option 'type' 'bridge'
    option 'proto' 'static'
    option 'netmask' '255.255.255.0'
    option 'ipaddr' '10.0.0.253'
    option 'ifname' 'eth0.0 eth0.1'
```

```
network.lan.ifname="eth0.0 eth0.1"
```

And finally, I list my new network configuration to make sure everything’s correct, commit the changes and reboot:

```
root@sugartongs:/etc/config# uci show network
root@sugartongs:/etc/config# uci commit
root@sugartongs:/etc/config# reboot
```

Listing 1 shows what the resulting /etc/config/network file looks like.

Note that on your system, sections may be listed “out of order”, for example, with one VLAN section near the top and another near the bottom. Commands *within* a given section need to be in the correct order, but the sections *themselves* do not, so don’t worry!

Disabling OpenWrt’s DHCP and Firewall System

You also have to disable OpenWrt’s native DHCP and iptables systems. The need for disabling DHCP services is obvious: acting as a DHCP server wouldn’t be very “transparent” behavior! So, disable it with these two commands:

```
root@sugartongs# /etc/init.d/dnsmasq disable
root@sugartongs# /etc/init.d/dnsmasq stop
```

OpenWrt’s native iptables script (/etc/init.d/firewall) is fine if you want to use OpenWrt as a standard “Layer 3” (routing) firewall. Leaving this script enabled allows you to use the uci command and the file /etc/config/firewall to manage iptables in a manner very similar to how you manage network configuration and other OpenWrt system settings.

However, this system doesn’t lend itself very well to running iptables in bridging mode—to use it that way, you’d need to hack the script extensively, which would be a bewildering task given the large number of custom tables it uses beyond “INPUT”, “OUTPUT” and “FORWARDING”. Therefore, disable it like this:

```
root@sugartongs# /etc/init.d/firewall disable
root@sugartongs# /etc/init.d/firewall stop
```

Now you can create a custom iptables script more suitable for a transparent firewall.

Example Network Topology

In order to write a firewall script, you need to consider your network’s topology and how the transparent firewall fits in. Figure 1 shows the example home network I sketched out in Part II of this series, with a firewall cabled between the network’s Internet uplink (via DSL router or cable modem) and its backbone (which collapses back to a wireless broadband

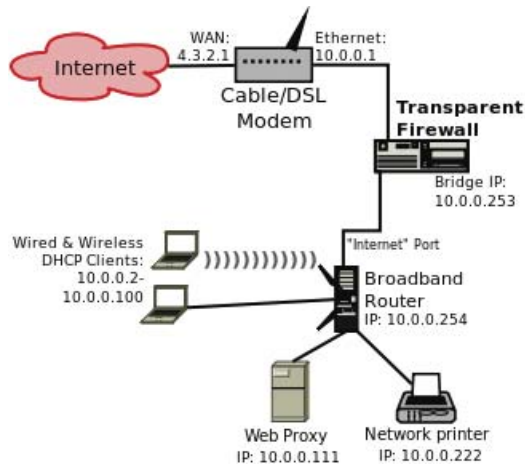


Figure 1. Example Home Network

router configured with Internet uplink and LAN on the same logical subnet).

You could use a number of topologies instead. If you have only a few hosts on your internal network, and your Internet uplink device is already providing DHCP services, you could use your transparent firewall as your broadband router (though configuring WLAN on OpenWrt is outside this series' scope). If your cable modem or DSL router includes a switch and/or wireless LAN access point, you could connect some of your network nodes directly to that and use your transparent firewall to protect other devices.

I'm going to stick with the topology in Figure 1, however, for simplicity's sake. It should be clear enough how to customize my sample iptables script for whatever topology you choose. Let's take a closer look at Figure 1.

In order to write a firewall script, you need to consider your network's topology and how the transparent firewall fits in.

The first thing you should notice is that everything on this network resides on the same logical subnet (10.0.0.0/24) except, of course, for the cable/DSL modem's WAN interface (the one connected to the Internet), which has the Internet-routable address 4.3.2.1. That WAN address is strictly illustrative; in actual practice, WAN IP addresses in *any* residential Internet scenario are assigned by your Internet service provider, often automatically, so please don't attempt to set yours to 4.3.2.1!

Another important point is that on this example network, client PCs are assigned IP addresses via DHCP from the pool 10.0.0.2 through 10.0.0.100.

My diagram doesn't indicate which host is providing DHCP services. Is it the cable/DSL modem, the broadband router or the Web proxy?

As a matter of fact, it doesn't matter! Because this entire network fabric is switched, DHCP requests will propagate freely, including through the transparent firewall. However, if the cable/DSL modem acts as the DHCP server, you will need to write rules on the firewall to allow DHCP through in both directions.

Logical Firewall Design

Now that you understand what the network looks like, let's decide how to manage its dataflows. In my example scenario, the firewall will have a "deny by default" policy, as any good firewall should. The task, therefore, will be one of anticipating and allowing the dataflows you need the firewall to allow.

First, assuming the LAN's DHCP server is upstream of the firewall, you need to allow DHCP traffic between UDP port 67 (the DHCP server port) and UDP port 68 (the DHCP client port).

Next, you don't want to lock yourself out of the firewall itself! You need to allow traffic from the LAN to TCP port 22 on the firewall.

As you can see in Figure 1, the example network has an outbound Web proxy. Because one of the best uses of a firewall is to enforce use of a Web proxy, you'll for sure want to allow only outbound Web traffic originating from the Web proxy. You'll also allow outbound DNS queries (and corresponding replies).

That's it! Things downstream of the firewall—that is, transactions between hosts connected to the broadband router shown in Figure 1—don't need to be allowed by the firewall. For example, print jobs sent from wired and wireless DHCP clients to the network printer don't need an "allow LPR" rule, because those packets should never reach the transparent firewall in the first place.

(If, however, you have only a few hosts on your LAN and elect to omit the downstream switch or broadband router and cable them directly to the transparent firewall, this will not be the case. You *will* need to allow for "LAN-to-LAN" transactions of that type.)

Creating a Custom Firewall Script

Now, finally, you're ready to write a custom firewall script! You could, of course, simply edit the file `/etc/init.d/firewall`. But, that would make it harder to revert to OpenWrt's native `uci-driven` firewall system later—better to leave that script alone. I prefer to create a new script from scratch, arbitrarily named `/etc/init.d/iptables.custom`.

Listing 2 shows what `/etc/init.d/iptables.custom` needs to look like in order to implement the firewall policy we arrived at in the previous section. Let's

Listing 2. Custom iptables Startup Script

```
#!/bin/sh /etc/rc.common
# Customized iptables script for OpenWrt 10.03

START=46

IPTABLES=/usr/sbin/iptables
LOCALIP=10.0.0.253
LOCALLAN=10.0.0.0/24
WEBPROXY=10.0.0.111

stop() {
    echo "DANGER: Unloading firewall's Packet Filters!"
    $IPTABLES --flush
    $IPTABLES -P INPUT ACCEPT
    $IPTABLES -P FORWARD ACCEPT
    $IPTABLES -P OUTPUT ACCEPT
}

start() {
    echo "Loading custom bridging firewall script"

    # Flush active rules, custom tables
    $IPTABLES --flush
    $IPTABLES --delete-chain

    # Set default-deny policies for all three default tables
    $IPTABLES -P INPUT DROP
    $IPTABLES -P FORWARD DROP
    $IPTABLES -P OUTPUT DROP

    # Don't restrict loopback (local process intercommunication)

    $IPTABLES -A INPUT -i lo -j ACCEPT
    $IPTABLES -A OUTPUT -o lo -j ACCEPT

    # Block attempts at spoofed loopback traffic
    $IPTABLES -A INPUT -s $LOCALIP -j DROP

    # pass DHCP queries and responses
    $IPTABLES -A FORWARD -p udp --sport 68 --dport 67 -j ACCEPT
    $IPTABLES -A FORWARD -p udp --sport 67 --dport 68 -j ACCEPT

    # Allow SSH to firewall from the local LAN
    $IPTABLES -A INPUT -p tcp -s $LOCALLAN --dport 22 -j ACCEPT
    $IPTABLES -A OUTPUT -p tcp --sport 22 -j ACCEPT

    # pass HTTP and HTTPS traffic only to/from the web proxy
    $IPTABLES -A FORWARD -p tcp -s $WEBPROXY --dport 80 -j ACCEPT
    $IPTABLES -A FORWARD -p tcp --sport 80 -d $WEBPROXY -j ACCEPT
    $IPTABLES -A FORWARD -p tcp -s $WEBPROXY --dport 443 -j ACCEPT
    $IPTABLES -A FORWARD -p tcp --sport 443 -d $WEBPROXY -j ACCEPT

    # pass DNS queries and their replies
    $IPTABLES -A FORWARD -p udp -s $LOCALLAN --dport 53 -j ACCEPT
    $IPTABLES -A FORWARD -p tcp -s $LOCALLAN --dport 53 -j ACCEPT
    $IPTABLES -A FORWARD -p udp --sport 53 -d $LOCALLAN -j ACCEPT
    $IPTABLES -A FORWARD -p tcp --sport 53 -d $LOCALLAN -j ACCEPT

    # cleanup-rules
    $IPTABLES -A INPUT -j DROP
    $IPTABLES -A OUTPUT -j DROP
    $IPTABLES -A FORWARD -j DROP
}
```

dissect it.

First, note the includes file `/etc/rc.common` at the top: this provides functions like `enable`, `disable` and other housekeeping functions that OpenWrt uses to manage startup files.

Next, `START=46` specifies the priority/order for running this script at startup. 46 is the same slot that the default OpenWrt “firewall” startup script uses, which is to say, after networking is enabled but before the DropBear SSH server and other network services are started.

Next come some “shorthand” variables we’ll use throughout the script. `IPTABLES`, obviously enough, specifies the full path to the local iptables command. `LOCALIP` is the firewall’s bridge IP address; `LOCALLAN` is the network address of the local LAN, and `WEBPROXY` gives the IP address of the Web proxy.

The “stop” function (as in `./iptables.custom stop`) causes the script to flush all iptables rules from kernel memory and to load default ACCEPT policies for all three default firewall tables, `INPUT`,

`FORWARD` and `OUTPUT`. This does *not* “stop all traffic”; rather, it stops all *restrictions* on traffic (thus, the warning message).

Now we come to the heart of the script: the “start” function, containing the firewall policy in the form of a list of iptables commands.

First, flush any active rules and delete any custom tables, so you begin with a clean slate (`$IPTABLES --flush` and `$IPTABLES --delete-chain`). Next, set default deny policies for the `INPUT`, `FORWARD` and `ACCEPT` chains. (You could just as easily choose `REJECT` as the default policy, but because this involves sending ICMP replies to jilted clients, versus `DROP`’s simply ignoring them, there’s a slight performance benefit to `DROP`.)

Next come two rules to allow interprocess communication on the firewall itself, by allowing all packets arriving from and destined for the “loopback” interface. This is followed immediately, however, by an antispoofing rule that blocks traffic addressed to the firewall *from* the firewall’s own IP address.

Next are two rules allowing DHCP requests—that is, packets from UDP port 68 sent to UDP port 67—and DHCP responses—that is, packets from UDP port 67 to UDP port 68. These two rules are necessary only if your DHCP server is on the other side of your firewall from your DHCP clients.

You may have noticed that these two DHCP rules and the subsequent rules for SSH, HTTP proxying and DNS are “stateless”. Rather than invoking the iptables “state” module, which lets you allow, for example, outbound DHCP queries while letting the kernel decide what constitutes a valid response, you’re explicitly allowing the reply traffic. This is an admittedly archaic way to write iptables rules.

However, as I mentioned in the sidebar, OpenWrt has significant performance issues when used as a bridging firewall. Because the “state” module imposes still more of a performance hit, and because this firewall policy is simple to begin with, I’m doing it the old-fashioned way. For a bridging firewall on a better-performing distribution/hardware combination, I definitely would take advantage of Linux’s state-tracking features!

For a bridging firewall on a better-performing distribution/hardware combination, I definitely would take advantage of Linux’s state-tracking features!

The next pair of rules in Listing 2 allows SSH connections to the firewall itself, but only from the local LAN. Note that the “incoming” leg of SSH transactions is handled in the INPUT table, whereas the “outbound” leg is processed in the OUTPUT table. If you were using `-m state`, the OUTPUT leg would be implicit.

Next come two pairs of rules allowing only the Web proxy to send and receive traffic to/from TCP ports 80 and 443, which, of course, correspond to HTTP and HTTPS, respectively.

This wouldn’t work unless DNS did also, so next are rules allowing DNS queries to TCP and UDP ports 53 (ordinarily, DNS queries just use UDP, but once in a while they can occur over TCP as well).

Finally, the script ends with three “cleanup” rules that place a “drop all” rule at the bottom of each of the default tables. These are, of course, redundant with the default “DROP” policies I set near the beginning of the `start()` function, but specifying such cleanup rules are a firewall best practice; sometimes redundancy is desirable!

When you type in any firewall script, be careful! At the very least, double- and triple-check the SSH rules that allow access to the firewall. If there’s any

problem with those rules, you’ll be locked out once you run the script, and you may even need to re-flash your firewall to recover. You can fix other things if SSH works, but if SSH doesn’t work, you’ll be stuck.

Once you’re confident enough to test your rules, save the new script. Be sure to set the “execute” bit on it like so:

```
root@sugartongs:/etc/init.d# chmod a+x ./iptables.custom
```

And, enable the script at startup, like this:

```
root@sugartongs:/etc/init.d# ./iptables.custom enable
```

Now for the moment of truth—load the rules:

```
root@sugartongs:/etc/init.d# ./iptables.custom start
```

Test the rules by making sure the things you want to work still do (connecting back to the firewall via SSH, surfing the Web via your Web proxy and so forth). Also, be sure to test some things you don’t expect to work, such as surfing the Web *without* going through the proxy or connecting to an FTP server using an FTP client application. In my own experience, the challenge with OpenWrt is getting iptables to “see” and act on traffic; the real test is ensuring that it’s blocking *anything*!

Conclusion

And with that, I’ve completely filled up this month’s allotted space. I’ll wrap up the series next month with some tips and tricks, and a suitably flowery “Conclusion” paragraph that I promise will be much more worthwhile than this one. For now, I’ll simply say, “good luck!” ■

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”.

Resources

Home Page for the OpenWrt Project:
www.openwrt.org

OpenWrt’s Unified Configuration Interface
Documentation: wiki.openwrt.org/doc/uci

The OpenWrt Forum (where you’ll end up asking for help sooner or later, if you use OpenWrt more than very casually): <https://forum.openwrt.org>



How Much Are You Really Spending On Your Hosting?

Save up to 70%

On a Dedicated Server without the Hassle

- » Try our free calculator to see what you're really spending
- » Discover the advantages of a managed server over dedicated server and in-house solutions
- » Download a Free Internet.com Tech Brief

**Try our calculator by visiting www.verio.com/tco-calculator
or call us at 1.877.946.0299**

VERIO[®]
An NTT Communications Company



KYLE RANKIN

Working on My Temper

Find out whether I can port my beer fridge controller from a laptop to a Pogoplug without losing my temper.

In my August 2010 column, I wrote about how I used an old laptop, some spare X10 devices I had and a TEMPer USB thermometer to control my fridge so I could ferment beer in a nice, controlled environment. Since then, I've received a number of e-mail messages from *Linux Journal* readers talking about their own projects to create the perfect beer fridge. After reading about Arduino-powered fridges and other hardware-hacking solutions, I became jealous and began to think that my laptop solution, although simple, was overkill. My hardware-hacking skills aren't quite to Arduino levels yet, so I ultimately decided to look at other, lower-powered Linux devices I had around (it turns out

serial dongle. Sorry, Spykee robot, but my final choice was a pink Pogoplug (Figure 1). The Pogoplug is a nifty little Linux-powered appliance that allows you to connect your own USB hard drives and then provide that data to you wherever you are, as long as you have an Internet connection. Think of it somewhat like a personal Dropbox, except you have full control of the storage and data. There's also a good-size community around the Pogoplug that provides third-party applications and even a custom Linux distribution based on Arch Linux.

Plugbox Linux Installation

The best things about the Pogoplug for my purposes were that it was small, low-powered, had a custom Linux distribution with package management, and most important, it had four USB ports. Once I decided to use it instead of my laptop, the next step was to get the Arch Linux-based Plugbox Linux installed on it. This distribution is hosted on www.plugapps.com, and the site provides all sorts of third-party applications and packages for Pogoplug, DockStar and SheevaPlug devices. I simply followed the install document for Plugbox Linux verbatim, so instead of reposting it here, just follow the steps at www.plugapps.com/index.php5?title=PlugApps:Pogoplug_Setboot. Be sure to follow the advice about setting up `openntpd`; otherwise, Plugbox's time will be way off, and you'll get strange warnings and errors as you try to install and compile software.

Essentially, Plugbox Linux installs itself on an external USB drive that you provide and then sets up the bootloader so that if the drive is inserted, it will attempt to boot from it; otherwise, it will boot from the native Pogoplug firmware. This provides a simple, relatively low-risk way to modify the device to do what you want while still being able to go back to defaults. Although I might have been able to get the native Linux install to do what I wanted, Plugbox has simple package management using Arch Linux's `pacman`, so I also could go back to the standard Pogoplug firmware at any time. In my case, I used a spare 1GB thumbdrive for the OS, and that seemed to be plenty.

Bottlerocket Is a Breeze

After the Plugbox installation completed, I was able to boot in to the environment and configure everything over SSH. Now, all I needed to do was repeat

The Pogoplug is a nifty little Linux-powered appliance that allows you to connect your own USB hard drives and then provide that data to you wherever you are, as long as you have an Internet connection.

I had quite a few) to see if I couldn't coax one of them into controlling my fridge.

The kind of device I needed had to meet a few criteria. One, it should be able to run Linux. Second, it ideally would have at least one USB port that I could use (with a hub if necessary) to connect my USB thermometer, and if it didn't have a serial port, also connect a USB serial adapter so I could use my X10



Figure 1. Pogoplug—Little, Pink, Different

my steps to get the CPAN modules my TEMPer device needed, install bottlerocket, and I should be done. I guess I'm spoiled by all the packages available in Debian, but then again, I didn't expect bottlerocket to be packaged for this custom distribution. I figured I'd be able to get the CPAN modules I needed, but the real question was whether I could get bottlerocket to compile. Without it, this project would be over. It turns out it wasn't too difficult. I just used pacman to install my build environment and then downloaded and compiled bottlerocket like you would in the old days:

```
# pacman -S gcc make
# wget http://www.linuxha.com/bottlerocket/bottlerocket-current.tar.gz
# tar xvzf bottlerocket-current.tar.gz
# cd bottlerocket-0.04c/
# ./configure
# make
# make install
```

When I tested bottlerocket, I noticed something interesting. The USB-to-serial adapter I plugged in was detected and appeared to work; however, after I ran bottlerocket the first time, any subsequent execution would result in an error for /dev/ttyUSB0. I had to unload and reload that particular USB-to-serial module between each bottlerocket execution for it to work properly. I know from experience that not all USB-to-serial adapters are created equal and that some do a better job, for instance, with sending break signals to your console, so I figured this was just a case of a cheap serial adapter without full serial port support. Instead of just living with the kludge of reloading the module each time, I replaced this adapter with another one I had that I knew had excellent support for break signals and overall had better compatibility. With this new adapter, I was able to power my fridge on and off without issues.

Note: a Quick Rant about Pacman

I have to admit that Plugbox was my first foray into the Arch Linux distribution. I know a number of people who really like Arch, and I don't really have an opinion one way or the other about the distribution itself, but I wanted to say a few words about pacman. First, I love the name. Second, who came up with the command arguments? In my experience, capital letters in arguments always are reserved for when you run out of lowercase arguments (or alternatively, to do the opposite of what the lowercase argument does), and although sometimes arguments have no relation to the action you are performing, at least for most programs, the most commonly used actions are the ones with the intuitive arguments. For instance, I fully expected pacman to use -i to install a package or possibly to use a longhand --install. The -S argument it does use simply makes no sense (I know it stands for "Sync" but that's a long way from "install"), and the -Ss argument to search for packages that match a keyword makes even less sense. I know it takes only a second to get used to it, but I'd still argue there are a number of more intuitive letters to choose.

CPAN Is a Harsh Mistress

Because it was so easy to build bottlerocket, I assumed it would be a relative cakewalk to install all the Perl modules I needed using CPAN. For those of you who aren't Perl hackers, Perl provides a vast repository of extra modules on-line at www.cpan.org. If there is a particular Perl module you need, you can use the cpan binary on your local system to pull down and build those modules for your system much like a modern package manager. I figured that even if Plugbox didn't include a lot of Perl modules, I would identify the modules I needed and install them one by one, like in my previous TEMPer column. The reality was that although Plugbox did include Perl and even included the cpan binary, it didn't include a lot of the necessary modules you need for CPAN actually to work.

Honestly, everything else about this project was simple once I got CPAN working, but this part of the process took the most time and effort, and it really reminded me of what Linux was like back in the day when you would find a cool project on Freshmeat, download the source and then spend the next two days tracking down all the dependencies. I finally found the magic list of packages and libraries I needed, and in

Low Cost Panel PC

PDX-089T

- Vortex86DX 1 GHz Fanless CPU
- Low Power Consumption
- 1 RS232/422/485 serial port
- Mini-PCI Expansion slot
- 2 USB 2.0 Host Ports
- 10/100 BaseT Ethernet & Audio
- PS/2 mouse & keyboard
- CompactFlash & MicroSD card sockets
- Resolution/Colors: 1024 x 600 @ 256K
- Resistive Touch Screen
- Linux, Embedded XP or Windows CE OS Options






Setting up a Panel PC can be a *Puzzling* experience. However, the PDX-089T comes ready to run with the Operating System installed on flash disk. Apply power and watch the Windows XPE user interface appear on the vivid color LCD. Interact with the PDX-089T using the responsive integrated touchscreen. Everything works out of the box, allowing you to concentrate on your application rather than building and configuring device drivers. Just Write-It and Run-It... Starting at \$450 Qty 1.

For more info visit: www.emacinc.com/panel_pc/pdx089.htm

Since 1985

OVER

24

YEARS OF

SINGLE BOARD

SOLUTIONS



EQUIPMENT MONITOR AND CONTROL

Phone: (618) 529-4525 • Fax: (618) 457-0110 • www.emacinc.com

Listing 1. Refrigerator Temperature Controlling Script

```
#!/usr/bin/perl

my $temp_min = 67;
my $temp_max = 69;

use 5.010;
use strict;
use warnings;
use Carp;
use Device::USB;
use Device::USB::PCSensor::HidTEMPer::Device;
use Device::USB::PCSensor::HidTEMPer::NTC;
use Device::USB::PCSensor::HidTEMPer::TEMPer;
use lib;
use Device::USB::PCSensor::HidTEMPer;

my $pcsensor = Device::USB::PCSensor::HidTEMPer->new();
my @devices = $pcsensor->list_devices();
my $logfile = '/var/log/temper.log';
my $time = localtime();
my $temperature;

foreach my $device ( @devices ){
    $temperature = $device->internal()->fahrenheit();
}

die unless $temperature;

open LOG, ">> $logfile" or die "Can't open $logfile: $!\n";

# B4 = Fridge power, B5 = Heater power

# turn on heater if I'm 1F below the low temp
if($temperature < ($temp_min - 1)){
    system('/usr/local/bin/br --port /dev/ttyUSB0 B5 ON');
    system('/usr/local/bin/br --port /dev/ttyUSB0 B4 OFF');
    print LOG "$time\t$temperature\tHON\n";
}
elseif($temperature < $temp_min){
    system('/usr/local/bin/br --port /dev/ttyUSB0 B4 OFF');
    system('/usr/local/bin/br --port /dev/ttyUSB0 B5 OFF');
    print LOG "$time\t$temperature\tOFF\n";
}
elseif($temperature > $temp_max){
    system('/usr/local/bin/br --port /dev/ttyUSB0 B4 ON');
    system('/usr/local/bin/br --port /dev/ttyUSB0 B5 OFF');
    print LOG "$time\t$temperature\tCON\n";
}
else{
    print LOG "$time\t$temperature\t\n";
}

close LOG;
```

this case, I had to install a few packages from Plugbox along with a manual module compilation:

```
# pacman -S perl-yaml perl-xml-libxml
# pacman -S perl-text-query perl-text-diff perl-text-reform
# wget http://search.cpan.org/CPAN/authors/id/R/RC/RCLAMP/
#   Text-Glob-0.08.tar.gz
# tar xfvz Text-Glob-0.08.tar.gz
# cd Text-Glob-0.08
# perl Build.PL
# perl Build test
# perl Build install
```

From this point, I was able to get the cpan binary to function, and I could follow the installation steps I laid out in my previous column to finish the process (I was glad I had that nicely documented for myself). Afterward, all I had to do was edit my temper.pl script so that it referenced /dev/ttyUSB0 instead of a local serial port and created a file called /etc/cron.d/temper that contained the following data:

```
* * * * * /usr/local/sbin/temper.pl
```

After I restarted cron (/etc/rc.d/crond restart), I was able to check my logs and see that the script was, in fact, reading the temperature and controlling the fridge just like my laptop. In case you'd like to do something similar with your Pogoplug, Listing 1 is the current iteration of the script.

Now that I've had this running without issues for a few weeks, I do like how quiet and low-power the device is. Plus, it takes up less space on the top of my fridge. The only real drawback I've seen is that there is no screen on the device. On my laptop if I wanted to check the temperature, I just had to open the lid; now, I have to ssh in to the Pogoplug. Although I used a Pogoplug for this, I imagine you could translate these steps for a number of other small Linux devices that have USB ports, like the NSLU2—provided it included Perl and a gcc build environment. Now I can use my old laptop for nobler pursuits—like maybe some day powering my smoker. ■

Kyle Rankin is a Systems Architect in the San Francisco Bay Area and the author of a number of books, including *The Official Ubuntu Server Book*, *Knoppix Hacks* and *Ubuntu Hacks*. He is currently the president of the North Bay Linux Users' Group.

Resources

Pogoplug: www.pogoplug.com

Applications for Pogoplug, Including Plugbox:
www.plugapps.com

9TH USENIX CONFERENCE ON FILE AND STORAGE TECHNOLOGIES (FAST '11)

Sponsored by USENIX in cooperation with ACM SIGOPS
FEBRUARY 15–18, 2011, SAN JOSE, CA, USA
<http://www.usenix.org/fast11>

WORKSHOP ON HOT TOPICS IN MANAGEMENT OF INTERNET, CLOUD, AND ENTERPRISE NETWORKS AND SERVICES (HOT-ICE '11)

Co-located with NSDI '11
MARCH 29, 2011, BOSTON, MA, USA
<http://www.usenix.org/hotice11>
Submissions due: December 9, 2010

8TH USENIX SYMPOSIUM ON NETWORKED SYSTEMS DESIGN AND IMPLEMENTATION (NSDI '11)

Sponsored by USENIX in cooperation with ACM SIGCOMM and ACM SIGOPS
MARCH 30–APRIL 1, 2011, BOSTON, MA, USA
<http://www.usenix.org/nsdi11>

EUROPEAN CONFERENCE ON COMPUTER SYSTEMS (EUROSYS 2011)

Sponsored by ACM SIGOPS in cooperation with USENIX
APRIL 10–13, 2011, SALZBURG, AUSTRIA
<http://eurosys2011.cs.uni-salzburg.at>

13TH WORKSHOP ON HOT TOPICS IN OPERATING SYSTEMS (HOTOS XIII)

Sponsored by USENIX in cooperation with the IEEE Technical Committee on Operating Systems (TCOS)
MAY 8–10, 2011, NAPA, CA, USA
<http://www.usenix.org/hotos11>
Submissions due: January 15, 2011

3RD USENIX WORKSHOP ON HOT TOPICS IN PARALLELISM (HOTPAR '11)

MAY 26–27, 2011, BERKELEY, CA, USA
<http://www.usenix.org/hotpar11>

2011 USENIX FEDERATED CONFERENCES WEEK JUNE 12–17, 2011, PORTLAND, OR, USA

EVENTS INCLUDE:

2011 USENIX ANNUAL TECHNICAL CONFERENCE (USENIX ATC '11)

JUNE 15–17, 2011
<http://www.usenix.org/atc11>
Submissions due: January 12, 2011

2ND USENIX CONFERENCE ON WEB APPLICATION DEVELOPMENT (WEBAPPS '11)

JUNE 15–16, 2011
<http://www.usenix.org/webapps11>
Submissions due: January 21, 2011

20TH USENIX SECURITY SYMPOSIUM (USENIX SECURITY '11)

AUGUST 10–12, 2011, SAN FRANCISCO, CA, USA
<http://www.usenix.org/sec11>
Submissions due: February 10, 2011

23RD ACM SYMPOSIUM ON OPERATING SYSTEMS PRINCIPLES (SOSP 2011)

Sponsored by ACM SIGOPS in cooperation with USENIX
OCTOBER 23–26, 2011, CASCAIS, PORTUGAL
<http://sosp2011.gsd.inesc-id.pt/>
Deadline to register abstracts: March 11, 2011

25TH LARGE INSTALLATION SYSTEM ADMINISTRATION CONFERENCE (LISA '11)

DECEMBER 4–9, 2011, BOSTON, MA, USA
<http://www.usenix.org/lisa11>

Stay Connected...



<http://www.usenix.org/facebook>



<http://twitter.com/usenix>

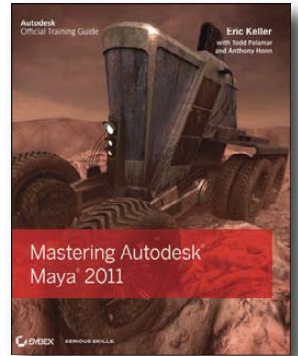
USENIX: THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

TECHNICAL SESSIONS AND TRAINING PROGRAM INFORMATION AND HOW TO REGISTER ARE AVAILABLE ONLINE AND FROM THE USENIX OFFICE:
<http://www.usenix.org/events> | Email: conference@usenix.org | Tel: +1.510.528.8649 | Fax: +1.510.548.5738

Eric Keller's *Mastering Autodesk Maya 2011* (Sybex)

If you just know enough Autodesk Maya to be dangerous, pick up Eric Keller's new book *Mastering Autodesk Maya 2011*, and take it to the next level. In this book, Keller offers professional-level Maya instruction, exploring topics such as modeling, texturing, animation, visual effects and other high-level techniques for film, television, games and so on. Included are pages of scenarios and examples from some of the leading professionals in the industry so that the reader can master the entire CG production pipeline. The book also covers the very latest Maya tools and features, including Dynamics, Maya Muscle, Stereo Cameras, rendering with mental ray and others.

www.sybex.com



Marvell's ARMADA 628 Processor

Silicon solutions provider Marvell recently rolled out its new ARMADA 628 processor, which the firm bills as the world's first 1.5GHz tri-core application processor, delivering dual-stream 1080p 3-D video and graphics for smartphones and tablets. The ARMADA 628 incorporates a full SoC design with three high-performance, ARM-compliant CPU cores. The tri-core design, with its two high-performance symmetric multiprocessing cores and a third core optimized for ultra low power "is analogous to a hybrid muscle car", says Marvell. The ARMADA 628 can perform like a racecar engine on demand, but it relies on the frugal third core for routine user tasks and system management. In real-world terms, this enables the ARMADA 628 to play more than ten hours of full 1080p HD video or 140 hours of music on a single charge while still providing 3GHz of raw computational horsepower. Marvell also says that the ARMADA 628 is the first mobile CPU to provide high-speed USB 3.0 connectivity.



www.marvell.com

Adeptol's Text Extraction Software

Adeptol's new Text Extraction application is designed to extract text from documents in more than 150 file formats, which then can be processed by content aggregation tools and used for storing, publishing, archiving or searching. Adeptol's Java-based software mines text at up to 15,000 words per second and can be deployed on Linux, Solaris or Microsoft Windows. Some of the more than 150 file formats include Microsoft Office, OpenOffice.org and PDF. The software's output can be exported to a text file or text stream, which can be saved into a database or passed on to other applications. Developers also can leverage Text Extraction to build text extraction capabilities directly into their applications.

www.adeptol.com



Paragon Software Group's Paragon NTFS and HFS for Linux Combo Professional

Knocking down remaining barriers created by incompatible operating systems is the mission of Paragon Software Group and its upgraded Paragon NTFS and HFS 8.1 for Linux Combo Professional. Paragon calls the application suite "the industry's highest-performance kernel driver for NTFS and HFS+ filesystems with advanced read and write operations for all types of files". Tested on Linux kernels up to 2.6.33, Paragon NTFS and HFS for Linux demonstrates read and write performance similar to Linux native Ext3FS with up to 80MB/sec read/write speed. Version 8.1 offers innovations, such as a 40% performance gain on NTFS filesystems, support for compressed NTFS files, full read/write support for HFS+ and HFSX, and creation and repair of HFS+ volumes. Paragon also says that its NTFS driver is more robust than native Microsoft's own. Personal and commercial editions are available.

www.paragon-software.com



Timesys Corporation's Web Factory

Timesys Corporation is calling on Linux application developers to test-drive Web Factory, a new and free cloud-based application that gives platform and application developers an easy-to-use tool for building Linux applications. The Web Factory application combines the Linux kernel, toolchain, debugger, the TimeStorm Eclipse-based IDE and how-to documentation to provide a complete embedded Linux build system. Everything is included that developers need to test and evaluate a processor without having to set up host build environments and before finalizing hardware selection. An easy-to-use wizard guides them through each step. Developers also do not need to spend time learning each free BSP/SDK provided by board vendors while testing boards. Key processor architectures including ARM, MIPS, Nios II, Power Architecture, SuperH and x86 are supported. Users can upgrade to Timesys's Desktop Factory subscription anytime if they need live, expert Linux support or advanced features and in-depth customization on a selected platform.

www.timesys.com

Dovie On-line Video Platform

The startup Dovie, Inc., has come into existence to give you one thing: dovie.tv—an enterprise-grade on-line video platform with built-in HTML5-ready players, Flash-free playlist embeds, analytics and advertising. The SaaS platform, says Dovie, offers a video control cloud with enough power to launch an on-line TV show, channel or network from a desktop computer, and it is simple and affordable enough for almost anyone to use. That power is complemented by the platform's tools that focus on making on-line video management and monetization accessible to everyone from mom-and-pop shops, to mid-market entrepreneurs, to enterprise-level professionals. Dovie also comes plugged in to major ad networks and ready-to-run CPM-optimized pre-roll video spots. Commercial content is welcome and doesn't require video producers to relinquish any rights to use the system.



dovie.tv

Recompute Linux PCs



The new PC maker, Recompute, is now offering Linux-based, OS-free and Windows-based desktop computers that offer a fresh approach to sustainability during the entire product life cycle. The company's approach is based on an intensive study exploring the most sustainable way to produce, use and dispose of PCs. Surprisingly, Recompute cases are manufactured of (recyclable and renewable) corrugated cardboard and treated only with nontoxic glues and flame retardant. To avoid the waste of electronic components that the user doesn't need, only memory, power supply and a hard drive come standard, while the eight USB ports allow the user to accessorize as needed. The design philosophy further allows for easy dismantling and sustainable local recycling, avoiding the typical fate of hard-to-handle e-waste from PCs: exporting it to poor

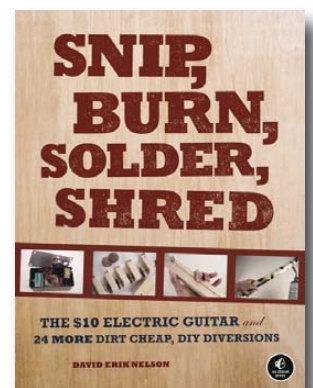
countries where its processing does tremendous damage to people and the environment. Recompute states that its goal is to change how we deal with our dead computers and hopes that others will follow their lead. The effort makes you wonder why more Linux gurus aren't in charge.

www.recompute.com

David Erik Nelson's *Snip, Burn, Solder, Shred* (No Starch)

Raising the next generation of creative geeks is serious business, and since Dr Spock never wrote the definitive guide to geek parenting, we're on our own. If Spock were alive, he might hand you David Erik Nelson's new book *Snip, Burn, Solder, Shred: Seriously Geeky Stuff to Make with Your Kids*. *Snip* is a book of geeky, do-it-yourself crafts and toys that one can make for \$10 or less. It includes illustrated, step-by-step instructions for 24 projects, such as water-powered bottle rockets, an oversized joy buzzer that (safely) administers a 100-volt jolt, booming thunderdrums made from salvaged x-ray film, a cigar-box synthesizer, a powerful muzzleloader that shoots marshmallows, homemade board games and more. As the readers build, they learn the basics of carpentry, sewing, circuitry and soldering. No technical experience is required, and the projects won't break the bank.

www.nostarch.com



Please send information about releases of Linux-related products to newproducts@linuxjournal.com or New Products c/o *Linux Journal*, PO Box 980985, Houston, TX 77098. Submissions are edited for length and content.

Fresh from the Labs

Whyteboard—PDF Annotation

whyteboard.org

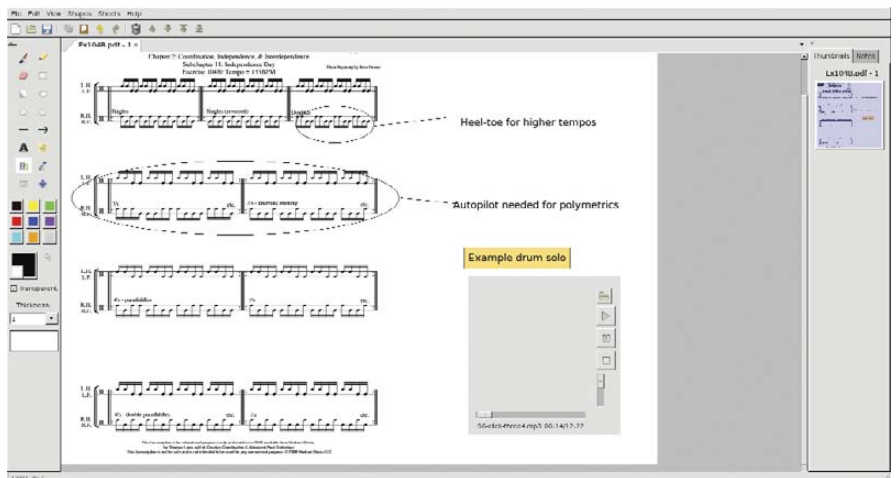
Every now and then an incredibly simple idea can totally change the way you take something mundane for granted, and Whyteboard is such a project. To quote the Web site: “Whyteboard is a free painting application for Linux, Windows and Mac. It is suited toward creating visual presentations and for overlaying PDF images with annotations.” Also according to the Web site, Whyteboard’s features include the following:

- Draw on a canvas using common tools: pen, rectangle, circle and text.
- Annotate over PDF files.
- Drawn shapes can be resized, moved, rotated and re-colored.
- Tabbed drawing: each tab represents a whiteboard “sheet”. Each sheet has unlimited undo/redo operations.
- Drawing history can be replayed on a per-sheet basis.
- Each sheet has a thumbnail of the canvas that updates as you draw.
- Closed sheets can be re-opened, restoring their data.
- Notes, similar to Post-It Notes—a tree in a side panel gives an overview of all notes.
- Resize the canvas easily by dragging it around.
- Embed an audio/video player onto the canvas.
- Translated into many languages (French, German, Spanish, Italian, Galician, Russian, Dutch and more).

Installation As far as binaries go, RPM and Debian packages are available straight from the Web site, as well as Windows binaries for those lesser mortals (all angry letters to the address below). The usual source tarball also is available,



Whyteboard is a tool to manipulate otherwise static PDFs. Here I’m using it to deface an old article of mine!



Features are plentiful in Whyteboard. Here I’m able to add notes to a Thomas Lang drum lesson and even add an example audio file.

and that’s what I’ll be running with here, but don’t worry, it doesn’t even need compiling to run.

As for library requirements, the documentation says you need the following:

- Python: 2.5, 2.6, 2.7 (untested on other major versions; should work on 2.4). Whyteboard does not work with Python 3 (www.python.org/download).

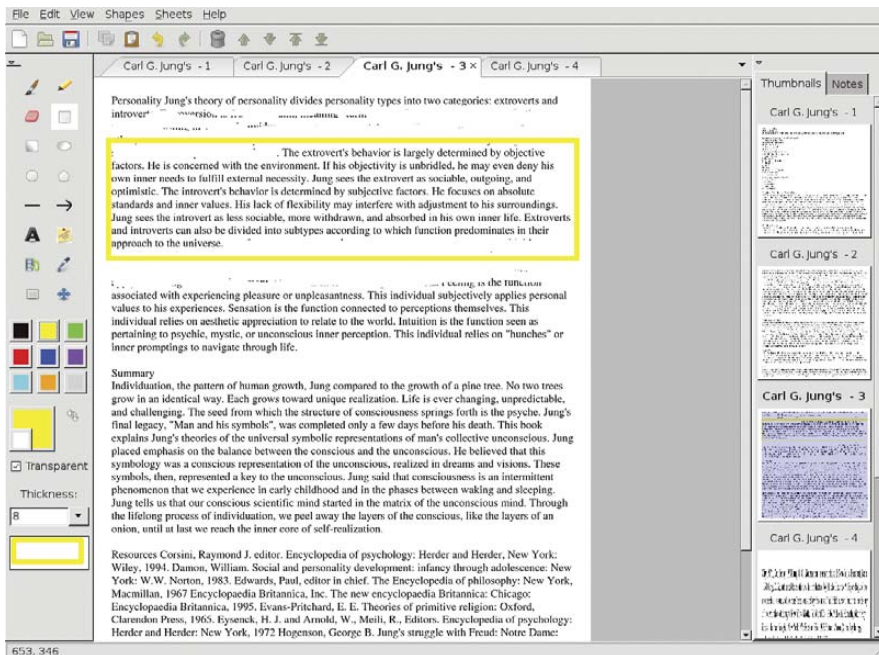
- wxPython: the latest version is *always* recommended (currently 2.8.11.0). Use the Unicode build. wxPython 2.8.9.0 needed at minimum (www.wxpython.org/download.php).

- ImageMagick: www.imagemagick.net.

Grab the latest tarball, extract it, and open a terminal in the new folder. Assuming you have the needed libraries installed, you can run the program simply by entering:

```
$ python whyteboard.py
```

Usage Upon entering the program, you’ll notice the GUI is strongly reminiscent of the classic Microsoft Paint, so I’m sure most of the computing public will feel instantly familiar with the interface.



Of particular use is the eraser tool, which I'm using here to scrub out surrounding parts of text and highlight a key point.

Before starting, you need to import a PDF file. From the menu, choose File→Import File→PDF, choose a PDF, and you're away. A word of warning, however; a large PDF (several hundred pages for instance) takes a long time to load and uses a lot of processing power, so I recommend sticking to something that's only a few pages.

For fun, to get started, choose a color from the left and start scribbling and doodling over the page. Now that that's out of the way, let's look at some of the interesting options.

Straight from the World of Paint is the Erase tool, and on PDFs, it's a revelation. Going over a technical document and being able to add notes and clear away sections around crucial text is extremely logical and quite indispensable once you're used to it. You'll wonder how you ever did without it.

Although I'm not sure whether you can rescale a page at this point in time (you can rescale the content though, as you'll see later), the resize tool is interesting in that you actually can add more whitespace to the side or bottom of a page, or crop off sections. Do this either by clicking and dragging your mouse in the gray sections outside the page or by using the Resize Canvas tool under the Sheets menu.

Using the Shape and Resize button,

not only can you rescale a page's content within the actual page, while the page stands still (pretty cool in itself), but you also can rescale the content as mentioned, as well as rotate it or even just relocate it.

The most unusual feature is embedding a multimedia player within parts of the page. For instance, I could be looking at a long and technical document, such as Jung's "Structure and Dynamics of the Psyche", for example, and include something related like a John Betts' Jungian podcast next to key points, or even a video lecture.

The most entertaining feature is the History Viewer. This shows you not only the modifications you've made, but also actually smoothly re-animates each brush stroke you've made, as you were making it. This makes for an amusing movie of your thought processes, and it really helps show whatever you were doing to get to where you are now.

Whyteboard is a beautifully simple program that anyone will be able to pick up. It re-applies existing technologies in new and innovative ways, and once you've become used to it, you won't be able to do without it. Although it's a little buggy for now, as development continues, I can see this becoming one of those daily tools used by the masses.

Transcoder Audio Edition—Audio Extraction and Conversion

transcoderae84.sourceforge.net

If you're looking for an audio converter, you could do much worse than this elegant little program. However, Transcoder's real draw is not its conversion abilities, but its extraction abilities. Feed it a video for which you've always wanted the sound (music videos spring instantly to mind), and you can extract it to play anytime you like. To quote the Web site: "Transcoder Audio Edition is an audio converter for Linux that can convert from one audio format into another and can extract audio tracks from video files and convert them into audio formats. It uses GTK+ as the GUI toolkit and FFmpeg as the back end."

Installation Available at the Web site is a 32-bit Debian binary (the recommended choice if possible), along with a "Binary+Source" tarball. I cover both here.

Documentation is sorely lacking, but fortunately, usage is very simple. Library requirements are minimal, with the only two real dependencies being the libglib2 and libgtk2 libraries.

I went with the Debian package first, but I had to force the architecture, as I'm running a 64-bit OS. Once installed, the program just worked with no issues. For the binary, run the program with the command:

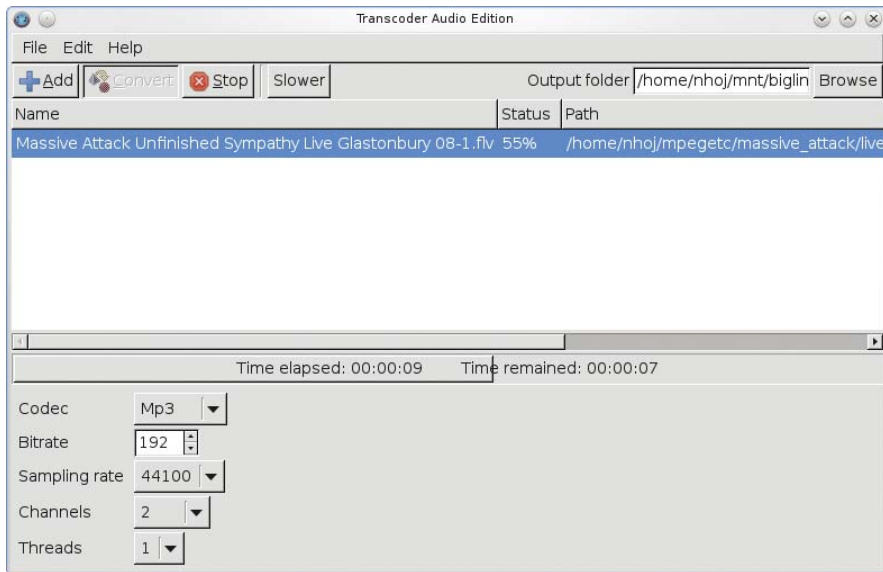
```
$ Transcoderae
```

If you're going with the tarball, simply extract the tarball, open a terminal in the folder, and enter the command:

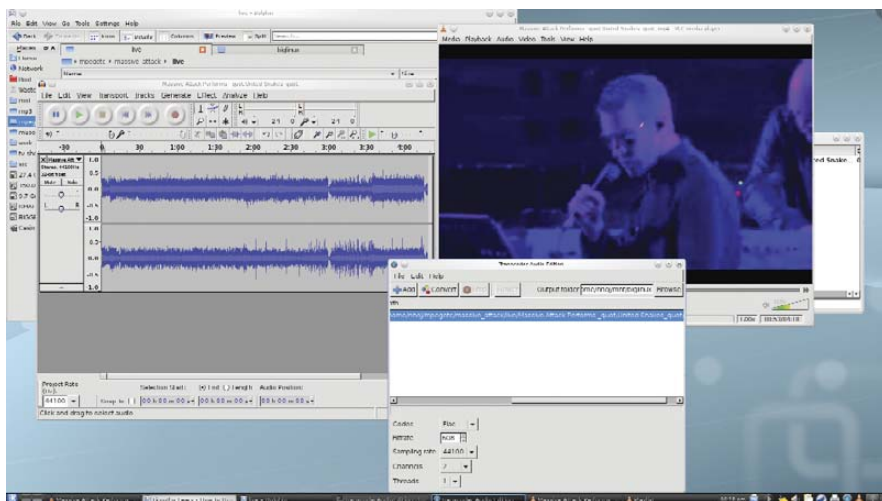
```
$ ./Transcoder
```

Usage My time with Transcoder was very easy; the interface is as simple as they come. First, click the Add button, and choose either the sound file you want to convert or the video from which you wish to extract audio. On the right is the field for the Output folder, where your resulting file ends up. If you don't want the file ending up in Home, click Browse and choose another folder.

Down below are the encoding options (very important—my installation had the choice of Vorbis, AAC, MP3, MP2, AMR-NB and FLAC). Next, you can specify the bitrate, followed by the sampling rate (44100 is CD audio quality; 48000 is what you get on DVDs). Finally, you have the Channels option, set to 2 by default (stereo), and you also can specify how many processing threads to use.



Transcoder AE is an elegant and simple tool for extracting audio from video files.



Here I've used Transcoder to extract the audio from a bad mix of a live performance to remaster it independently and sync it back with the video.

Then, press Convert, and that's pretty much it.

Although this idea is by no means new, the execution is wonderful, and no one is going to be put off by such a simple interface. The applications for this program are incredibly useful. For instance, you could extract the audio from a clip you grabbed from YouTube and play it in your car. Or, you could remaster some bad audio in a video file (which I'm currently attempting on both a *Metallica* and a *Massive Attack* video, where some crucial sound has been lost after a surround-to-stereo downmix). Or you simply can

convert one sound file to another in a clutter-free GUI that doesn't get in the way.

Either way, Transcoder Audio Edition is a painless program that will be of instant use to countless multimedia users. ■

John Knight is a 26-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.

Brewing something fresh, innovative or mind-bending? Send e-mail to newprojects@linuxjournal.com.

Top Five Projects

Looking back over the years, here are my top five favorite projects I've covered previously in this space. Thanks to all the readers for helping us reach our milestone 200th issue!

Tor—The Onion Router

(www.torproject.org) from the April 2010 issue. In a world of increasingly draconian Net surveillance, Tor has become an indispensable tool among journalists, activists, whistle-blowers, humanitarian workers and more. Using clever techniques to lose your IP address, Tor is the new standard in on-line anonymity.

htop (htop.sourceforge.net) from the October 2009 issue. Our old friend top gets a much needed makeover with a semi-GUI-style interface that still runs purely on the command line. Adding new and handy features along the way, hopefully htop will provide a home for a new generation of command-line users needing to control their system processes.

Danger from the Deep

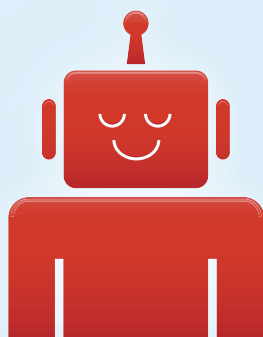
(dangerdeep.sourceforge.net) from the December 2009 issue. With an approach of passion and authenticity, *Danger from the Deep* is a WWII German submarine simulator with graphics, a soundtrack and an interface that many would associate with a commercial project. *Danger from the Deep* caters to true fans in a way that a commercial projects usually can't.

Longomatch

(longomatch.ylatuya.es) from the August 2009 issue. A video editing tool designed specifically around sporting analysis, Longomatch lets you take game footage and make highlights from your own home, using a clever timeline interface. Tying together several freely available technologies, this is one of those innovative programs that only OSS can deliver to the general public.

Gnaural

(gnaural.sourceforge.net) from the July 2009 issue. Finally, my favorite project of all time—Gnaural. Beneath a bland gray window with a few lines on it lies an incredible concept: alter the speed of your brainwaves just by using sound. Using two close frequencies running side by side, referred to as Binaural Beats, Gnaural can train your brain to be more relaxed or more alert—mind-bending stuff.



Lullabot™

Learn Drupal & jQuery

FROM THE COMFORT OF
YOUR LIVING ROOM



The Lullabot Learning Series includes everything you need to become a Drupal & jQuery expert from the comfort of your living room! The videos are available in both DVD format and high-definition video download.

Purchase the videos at <http://store.lullabot.com>

HARDWARE

ZOTAC ZBOX HD-ID11

With the arrival of the NVIDIA ION GPU, you can build a media server that will fit in the palm of your hand. And, of course, it runs Linux. STEVEN EVATT

For years I have toyed with the idea of setting up a media server for my entertainment center. The challenges in my way included cost, features, aesthetics, background noise and user-friendliness. All of those things are important because I'm not the only person who will be using the system I build. Hardware and software technologies are coming together to address all these challenges. With the advent of the Intel Atom processor and the NVIDIA ION GPU, affordable hardware now is available that allows for an HDTV media server. Likewise, software, such as XBMC and Boxee, has matured and provides a fun and friendly user experience for all levels of users.

For \$249.99, the HD-ID11 is small, quiet and looks slick. The chassis is all plastic and feels a bit flimsy when deconstructed. However, the plastic is fairly thick and has tight tolerances. When the cover is in place and set screws tightened, it feels solid.

The front of the HD-ID11 has a 3.5mm headphone jack, a 3.5mm microphone jack, a USB port, an SD card reader, a power button and activity LEDs. The top of the case has a large blue O that lights up while the machine is on. It looks nice, but it can be turned off in the BIOS if it is bothersome. On the side, there is a USB port with a rubber stopper, and on the back, there are four more USB ports. The back also sports HDMI and DVI outputs, 10/100/1000 Ethernet, eSATA, optical out and a port for the power brick.

Hardware

What makes the ZOTAC ZBOX HD-ID11 special is all the power that's packed into the small package. The machine is only 7.4" x 7.4" x 1.73" (188mm x 188mm x 44mm). Here are some of the main technical specs:

- CPU: Intel ATOM D510 (dual-core, 1.66GHz), 667MHz front-side bus.
- Chipset: Intel NM10 Express chipset.
- GPU: NVIDIA ION GPU (with 512MB DDR3 memory).

- Networking: Gigabit (10/1000/10000 Mbps), 802.11b/g/n.
- Audio: onboard 8-channel digital audio, stereo analog audio.
- I/O: HDMI, DVI (DVI-to-VGA dongle included), S/PDIF, mic/headphone, 6 x USB 2.0, RJ45, eSata.
- Memory slot: 1 x 200-pin DDR2-800 SO-DIMM slot.
- Hard drive slot: 1 x 2.5" hard drive (SATA 3.0Gb/s).

One thing separating the ZBOX from the competition is that it does not ship with memory or a hard drive. This allows you to tailor the computer to your needs without buying too much hardware or paying an inflated price for those components.

The HD-ID11 has support for up to 4GB of memory by using a single 200-pin DDR2-800 memory module. I installed 2GB of Kingston DDR2 RAM, which performed flawlessly. If you plan on using the ZBOX for playback only, 1GB of RAM should be sufficient. With the extra RAM available, I decided to use a 640GB hard drive and went with local management for the media. The Intel Atom processor is powerful enough to do a good job with video playback (via the NVIDIA ION GPU) and manage the library at the same time.

BIOS

The ZBOX uses a standard American Megatrends BIOS that can be entered by pressing the Delete key during the boot phase. The settings I felt worth changing included the boot priority, turning off the ZBOX logo at boot time and having the ZBOX restart after a power failure. The other BIOS settings had sane defaults.

The first thing I noticed when I booted the ZBOX with Ubuntu was it did not take long before the CPU fan would spin up to maximum and start to sound like a jet engine. This was worrisome, as I



Figure 1. The ZOTAC ZBOX HD-ID11 with dual-core Atom D510 CPU, NVIDIA ION GPU and HDMI output makes a great low-cost home-theater media box.

intended to keep the ZBOX in my living room. Fortunately, there is a BIOS update available to fix this problem. As with most BIOS updaters, the updater used by the ZBOX requires a DOS boot disk to run. See wiki.fdos.org/Installation/BootDiskCreateUSB for some easy-to-follow instructions for creating a free DOS USB boot drive. Once the free DOS image is booted, you can switch to the drive with your BIOS flasher and follow the updater instructions.

OS Installation

Installing Debian on the box was the only time the Intel Atom processor felt slow. This step took more than twice as long as I was expecting. After booting into Ubuntu via the USB memory stick, I formatted my internal hard drive and ran debbootstrap to install Debian Squeeze on the hard drive. Once debbootstrap is complete, do not forget to fix the fstab, networking and install GRUB before rebooting.

H.264 Decode Acceleration

The main reason to opt for an Atom/ION box is for watching high-definition content. The ION GPU supports full hardware decode acceleration of all H.264 content



Figure 2. Back panel of the HD-ID11 has eSATA, four USB, 10/100/1000 Ethernet, DVI and HDMI outputs, optical out and a connector for power.



Figure 3. The front of the HD-ID11 has a 3.5mm headphone jack, 3.5mm microphone jack, SD card reader, USB port, activity LEDs and power button.



Figure 4. HD-ID11 motherboard with 2GB of Kingston RAM installed. Mount and thumbscrew is for the 2.5" HDD/SSD.

(1080i/p) with HDMI out. With the right software, you can watch both Blu-ray and ripped BD content.

I installed both XBMC and Boxee on the HD-ID11 to access my content. Both software packages provide a great user

experience and give you the ability to play virtually any type of content. They both provide easy-to-use interfaces into your own content and give you access to content available on the Internet. Boxee is a fork of XBMC, adding social networking

to your home-theater experience.

My greatest struggle with my HD-ID11 was getting the digital 8-channel audio to work over HDMI. Although instructions are located several places on the Web (including ubuntuforums.org/showthread.php?p=6589810), I was not able to get the audio over HDMI to function properly. This was not a deal-breaker, as the box does support audio out via the 3.5mm headphone jack.

Day-to-Day Use

I have used the HD-ID11 for about a month to play back my local content and to stream content from the Web. I must admit, I love the experience. With few exceptions, the ZBOX has been able to handle any type of content I've thrown at it from inside XBMC and Boxee.

The only content the ION GPU struggles with is Flash video. According to Anadtech.com, the problem stems from the NVIDIA driver requiring too much data to be copied back and forth between the system memory and the GPU framebuffer. There is not enough bandwidth over the single PCIe1 lane to handle this load, which leads to the video stuttering. Even a 480p window does not play smoothly once full-screened. The good news is that NVIDIA is working on an updated driver to fix this problem.

Conclusion

When I started looking at the ZOTAC ZBOX HD-ID11, I wanted to build an affordable system to watch my high-definition content—one that looked nice, was quiet and user-friendly. With a little work, the HD-ID11 fits the bill. Its sleek design and quiet fans allow it to fit into my entertainment center without being noticed. The combination of the Intel Atom processor with the NVIDIA ION GPU provide all the power necessary to make for an enjoyable entertainment experience. Although I would prefer the sound going over HDMI and better Flash video playback, those are issues that should be addressed via driver updates in the future. I'm enjoying the media box so much, I am planning on purchasing a second ZBOX for my bedroom to give me more access to my content. ■

Steven Evatt is an IT manager in Houston, Texas, and has been using Linux for more than 16 years. He is active in the local technology community and regularly can be found at barcamps in Texas and Louisiana. In his spare time, he enjoys playing with Ruby on Rails on his site: pricechirp.com.

Barnes & Noble's Nook

For some retailers, e-books now are outselling paper books. E-book readers put a library in your hand—it may be time for you to make the plunge into e-books.

BILL CHILDERS

I'd never given e-readers much thought before. I've been reading e-texts on portable computers since I was in college in 1991 (I started on an HP 95LX Palmtop reading Project Gutenberg texts), so spending additional money on a dedicated device didn't make much sense to me. After all, I'd gone from the HP Palmtop PCs (95, 100 and 200LXs), to a Palm V, to a Palm LifeDrive, to a BlackBerry Curve, and I ended up using Stanza on the iPhone. I'm a fairly voracious reader—I read a book every week or ten days—so the idea of having to carry “yet another device” to get my literary fix just didn't sound like a great plan. As a result, I automatically discounted the Kindle, Nook and other e-readers due to their lack of utility, until a friend suggested I give the Nook a closer look. I'm glad I did; I was rather surprised by what I found.

What's a Nook, Anyway?

The Barnes & Noble Nook is an e-reader with a monochrome electronic ink (E Ink) screen, much like its primary competition, the Amazon Kindle. However, the Nook has one major distinguishing feature at first glance: a smaller, color, touchscreen below the primary E Ink screen. This little touchscreen serves as navigation device, virtual keyboard and optional page-flipping mechanism. There also are four hardware buttons on either side of the screen dedicated to page forward/reverse, which makes the Nook friendly for left-handed users. The Nook comes in two hardware models: one with Wi-Fi and 3G, and one with Wi-Fi only. It weighs about 12 ounces, and it's approximately 7.7" long, 5" wide and .5" thick. It reminds me of a Moleskine notebook in dimension, and I actually picked up a cover for the Nook that makes it look like one. The primary E Ink display is 600x800 “pixels” and measures 6" diagonally, while the secondary color touchscreen is 480x144 and measures 3.5" diagonally. Battery life on the Nook is impressive, running close to ten days of average use time if the wireless is turned off (two days with wireless on).



Figure 1. Barnes & Noble's Nook

This is due in large part to the E Ink screen, which uses power only when changing pixels on the display—it costs nothing to keep the display “lit”.

The 3G access (on 3G-capable Nooks) is locked to the Barnes & Noble on-line store *only*. You can't use it to surf the Web or use other on-line applications. The 3G is used only for buying books and getting content from Barnes & Noble's “The Daily” application, and it works only in the country in which you purchase the Nook (Barnes & Noble isn't going to pick up the tab for international roaming).

Other technical specifications of the Nook are:

- Samsung S3C6410 ARM CPU (seems to be clocked at 533MHz).
- 256MB of RAM.
- Internal 2GB Flash storage (it's actually a Sandisk MicroSD card, but it's not user-accessible).
- CyberTAN 802.11b/g Wi-Fi chipset.

- Sierra Wireless MC8777v 3G modem (on 3G-capable models *only*, locked to access only Barnes & Noble's e-store).
- User-accessible MicroSD card slot under back cover (for add-on data storage).
- MicroUSB port (for side-loading content from a PC and charging).
- Headphone jack.
- Speakers.
- Android 1.5 “Cupcake” operating environment.
- Linux 2.6.27 kernel.

The Nook can read several different media types, including:

- ePub e-books (with and without DRM).
- Mobi/e-reader e-books (with and without DRM).
- MP3 audio files (can be played in the background while reading).
- JPEG, GIF, PNG and BMP image types.

Using the Nook

When you first fire up the Nook after creating your Barnes & Noble store account and signing in (a mandatory process), you're greeted with wallpaper on the E Ink screen and several icons on the touchscreen (Figure 3). Each icon represents a function of the Nook. The Daily is content that Barnes & Noble updates each day. My Library is where you go to select content you already own (from the Barnes & Noble store or loaded from a PC). Shop takes you to the Barnes & Noble on-line store. Reading Now opens your current book, and Games gives you Sudoku and Chess games. There also is a Web browser included in the Nook (more on that later). The little n button between the E Ink screen and the touchscreen serves as a

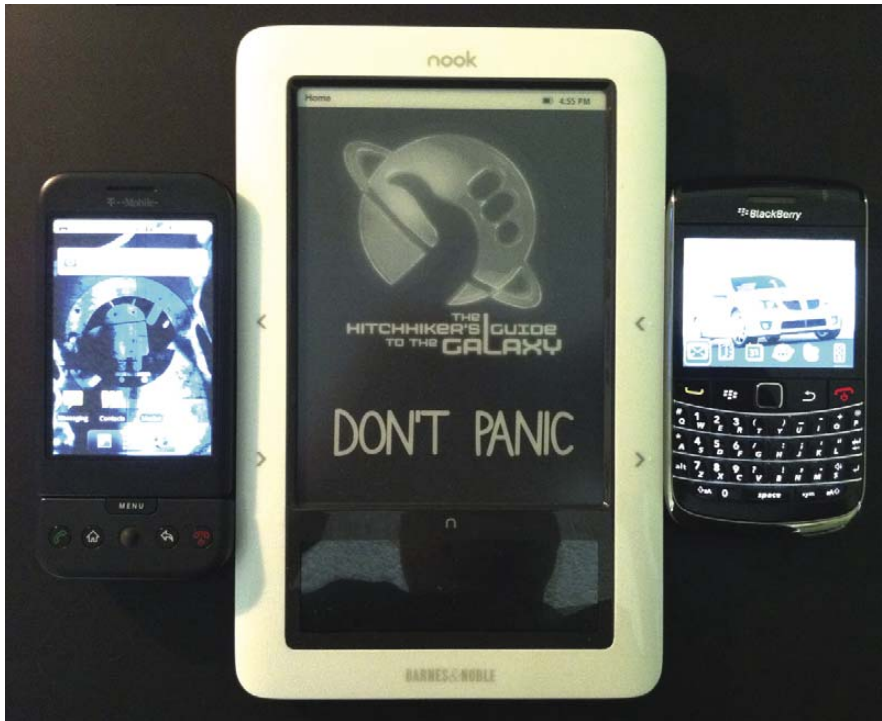


Figure 2. The Nook Next to a T-Mobile G1 and BlackBerry Bold for Size Comparison

quick shortcut to take you back to this main menu.

The Nook provides a very good reading experience, although it's not without a bump or two in spots. The E Ink screen is simply lovely for reading text. I was very skeptical at first about E Ink technology, due to the lack of backlight, slow response time and monochrome-only display. After using it, however, it's simply perfect for an e-reader. It's easy to read in reasonable lighting, and the response time is an issue for e-reading only when flipping pages, where it shows up as a black flash each time the page-forward button is pushed. The flash when flipping pages does take a little getting used to, but it's not terribly bad, and it's a decent trade-off for the excellent readability and extremely low-power consumption of the screen. The contrast of the screen improves in better lighting. In sunlight, my wife and I actually prefer it to a printed page. In Figure 4 you can see what the experience is like while reading John Scalzi's book *The*



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



**Powerful.
Intelligent.**



Charles heads the web development team here at Silicon Mechanics: he's responsible for the configurators and power calculator on our site, among other things. As a software expert, he offers a useful perspective on our server and storage products.

When asked what he would do if he had a Rackform iServ R413 configured with 4 8-core Intel® Xeon® Processor 7500 Series CPUs and 32 DDR3 DIMMs, he said, "32 virtual machines . . . one per core . . . in one rack unit." But he didn't stop there.

Charles knows that to make the best use of a server with that kind of processing horsepower in a virtualized environment, he needs I/O to match. He paired the 4P server with a Storform iServ R516 storage server, configured with 24 2.5-inch Intel X25-E solid state drives. Think of it as a developer's dream team: multi-core processing and high memory counts for blistering performance, and high-performance storage for blazing I/O speed.

Need a "dream team" of your own? Talk to the Experts at Silicon Mechanics for the perfect match.

When you partner with Silicon Mechanics, you get more than just the power and performance of the latest Intel technologies—you get an Expert like Charles.



For more information about the
Rackform iServ R413
visit www.siliconmechanics.com/R413

Expert included.

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.

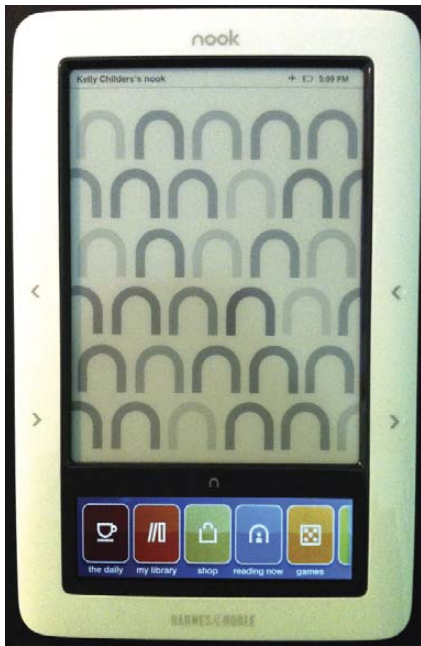


Figure 3. The Nook's Main Menu

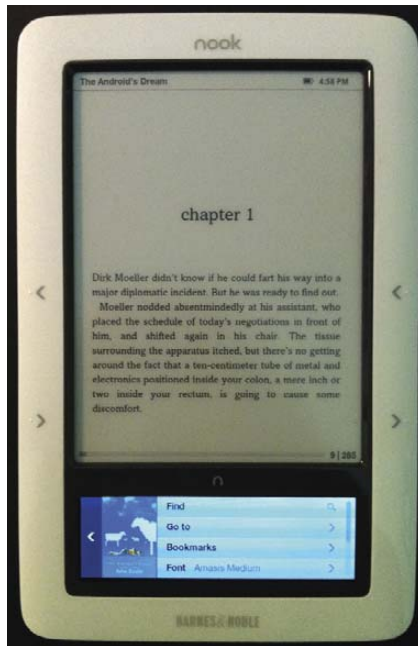


Figure 4. Reading *The Android's Dream*

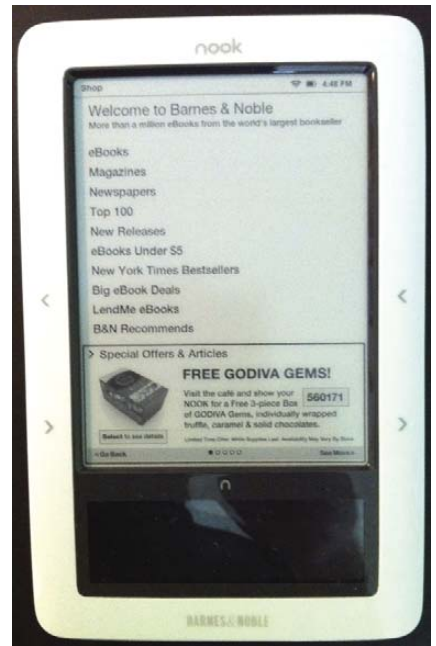


Figure 5. The Barnes & Noble Nook Store

Android's Dream (a hysterical book, by the way). The Nook's size and weight are just about perfect, and it feels good in your hand as you read. Kudos to Barnes & Noble for putting duplicate page-forward/-reverse buttons on both sides of the screen, so readers can switch hands without hassle. It has features that are expected to be in an e-reader, like text highlighting, bookmarking and the ability to look up a word in an onboard dictionary without

leaving the book.

The Nook's on-line store also is easy to use. Simply push the Shop icon from the main menu, and you're connected to the Barnes & Noble on-line store— instantaneously, via 3G if you're on a 3G-capable Nook; otherwise, the Nook needs to be associated with a Wi-Fi access point. You navigate the store using the lower touchscreen, and the search results and books are displayed on the E Ink screen. I've been using touchscreen devices for a

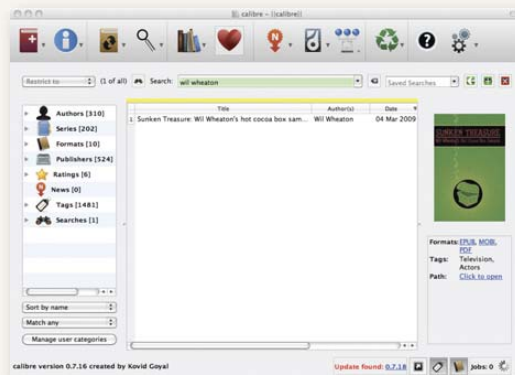
while, and I kept wanting to touch the E Ink screen to click on things at first, but that instinct didn't take long to overcome. Once you select a book, either via searching or browsing, buying it is as simple as clicking Buy Now on the touchscreen, and it's sent wirelessly to your Nook. An advantage of this is if there's ever an issue with your Nook and you need to replace it, once your replacement Nook is re-associated with your Barnes & Noble account, all your content automatically will be downloaded to the new Nook. Another tidbit about the Nook store is occasionally there will be offers for freebies if you take your Nook to a brick-and-mortar Barnes & Noble store. I've seen offers for free coffee, and Figure 5 shows an offer for free chocolate. I like free chocolate.

Using the Nook as a Web browser, unfortunately, represents an exercise in frustration. The processor is clocked just a little bit too slow to make for a decent browser. The E Ink screen's response time is abysmal when trying to select something, but the worst part about the experience is the split screen. You have to navigate the site using a small "window" of the site as seen through the touchscreen, but you can see the whole site on the E Ink screen (Figure 6). It gives you a schizophrenic feeling when trying to navigate a Web page. Sadly, it's just a bad experience.

Managing Your Content with Calibre

Calibre is a cross-platform open-source program that's designed to manage electronic books and other texts. It'll convert between e-reader formats as well as PC formats, like PDF and HTML, and it will download cover art and other e-book attributes automatically from the Internet. It's really easy to use, and it syncs with the Nook flawlessly.

If you have a lot of pre-existing content in other formats you want to put on your Nook, Calibre's the only way to go.



Calibre, an E-book Management Program

If You Use Linux, You
Should Be Reading

LINUX JOURNAL



- » In-depth information providing a full 360-degree look at featured topics relating to Linux
- » Tools, tips and tricks you will use today as well as relevant information for the future
- » Advice and inspiration for getting the most out of your Linux system
- » Instructional how-tos will save you time and money

Get *Linux Journal* delivered to your door monthly for 1 year for only \$29.50! Plus, you will receive a free gift with your subscription.

SUBSCRIBE NOW AT:
WWW.LINUXJOURNAL.COM/SUBSCRIBE

Offer valid in US only. Newsstand price per issue is \$5.99 USD; Canada/Mexico annual price is \$39.50 USD; International annual price is \$69.50. Free gift valued at \$5.99. Prepaid in US funds. First issue will arrive in 4-6 weeks. Sign up for, renew, or manage your subscription on-line, www.linuxjournal.com/subscribe.

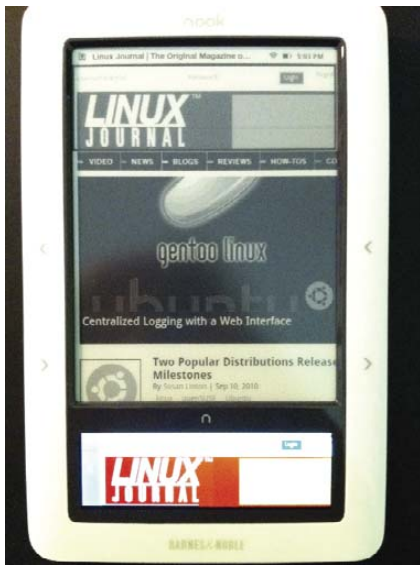


Figure 6. *Linux Journal's* Web Site, as Seen on the Nook

I think the Barnes & Noble folks tossed a Web browser in the unit because they could, and it gave them feature parity with the Amazon Kindle. And, I'm not sure there is a way to improve the experience on the current hardware.

The two games that ship with the unit are adequate, although if I'm going to play a game, I'll reach for my iPhone, laptop or Nintendo DS before I pick up the Nook. Having the ability to play an MP3 file and listen to it on speakers or headphones while reading is nice, but it's something I'll rarely use since I have other devices that play that kind of media better.

Conclusion

Overall, I like the Nook. Since I got my Nook, I've found myself "unplugging" from the computer and reading more. I like my Nook so much that the day after I got mine, I ran back to Barnes & Noble and bought my wife a Wi-Fi Nook. If I had to make the purchase again, I'd have gotten a Wi-Fi-only Nook myself, as I've used the 3G to purchase a book exactly once. The Nook's shortcomings aren't showstoppers to owning one, unless you need to rely on its Web browsing ability or need the fastest in screen refresh—in which case you probably shouldn't get any E Ink e-reader. It's hard to go wrong with the Wi-Fi Nook at the current \$149 price point, and although the extra \$50 for the 3G probably won't bankrupt anyone,

Softrooting the Nook

The Nook is an Android device, and as such, is capable of being "rooted", giving you full access to the hardware via the removal of software constraints. The softroot for the Nook is fairly easy to do, and you can do simply by downloading and installing a couple firmware bundles. After your Nook's been softrooted, you'll have the ability to change the launcher's main menu icons, as well as install other Android apps like Pandora. There are a couple Nook-specific applications as well: Trook (a way to download books you already own via Wi-Fi outside of the Barnes & Noble store) and Twook (a Twitter client with many of the same faults as the Web browser app). Note that at the time of this writing, new Nooks with serial numbers starting in the 1003 series are *not* compatible with the current softroot. Take care and check with the NookDevs site (see Resources) before attempting a softroot. If in doubt, don't do it!

it's more gimmicky than functional, as it's not very hard to find Wi-Fi around for downloading books. At any rate, if you're in the market for a dedicated e-reader, check out the Nook. ■

Bill Childers is an IT Manager in Silicon Valley, where he lives with his wife and two children. He enjoys Linux far too much, and probably should get more sun from time to time. In his spare time, he does work with the Gilroy Garlic Festival, but he does not smell like garlic.

Resources

Barnes & Noble Nook Home Page:
www.nook.com

NookDevs Home Page: nookdevs.com

Calibre Home Page:
calibre-ebook.com

Calibre Home Page:
calibre-ebook.com

READERS' CHOICE AWARDS 2010

The votes are in! Read on to find out how your favorites fared in this year's awards.



Welcome to the 2010 *Linux Journal* Readers' Choice Awards. We love doing these awards because we get to interact with you, our readers, more than usual. This year, more than 12,000 of you generously took time to participate and share your perspectives on what tools are helping you work and play. We always are fascinated by your preferences and how your usage patterns change over time. This year, we have more categories than ever, so let's get right to the results. Here, ladies and gentlemen, Linux geeks of all kinds, are the winners of your 2010 *Linux Journal* Readers' Choice Awards.

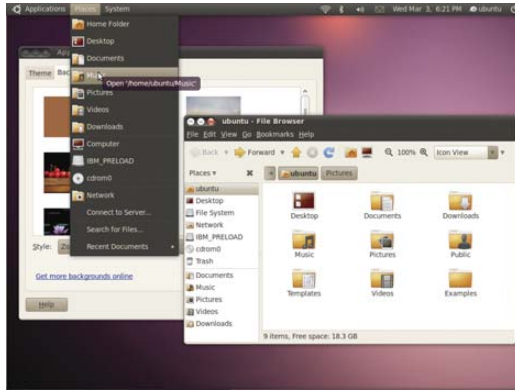
JAMES GRAY

BEST LINUX DISTRIBUTION

Ubuntu

Honorable Mention: PCLinuxOS

Times they are a changin' (just a bit) in the distribution department this year. Although Ubuntu in all its tasty flavors remains the Roger Federer-esque champion of Linux distros, the dynamics of this category have changed from the past two years. In 2008, Ubuntu was dubbed the "big distro that did" for "unexpectedly leaving its myriad 'rivals' as mere dots in the rear-view mirror". Then in 2009, Ubuntu received the "Energizer Bunny Award" for increasing its popularity and becoming untouchable (for the time being). But what a difference a year can make (just ask Tiger Woods). Ubuntu remains in the commanding lead, but it slipped a considerable 13% from of your votes last year. Meanwhile, a feisty, upwardly mobile distro, PCLinuxOS, grew from the single digits to a full 15%, meriting a worthy honorable mention recognition.



BEST DISTRIBUTION FOR NETBOOKS/LIMITED HARDWARE

Ubuntu Netbook Remix

Honorable Mention: Android OS

New for the 2010 Readers' Choice Awards is the category Best Distribution for Netbooks/Limited Hardware. This will be an interesting category to monitor as time goes by, but the current leader is Ubuntu Netbook Remix, which got a commanding 36% of your votes. The honorable mention winner, Android OS, was far behind, but it broke the 10% barrier. We predict that this category will become more fiercely competitive in the future.



BEST MOBILE OS

Google Android

Honorable Mention: MeeGo



Last year, Android and the T-Mobile G1 smartphone took home Linux Product of the Year honors, despite being absent from every other category selected by us (point taken, dear readers). One year wiser, we present a new category for 2010, Best Mobile OS, and its logical winner is Google's Android. But wait! Although Android's vote haul was impressive at 66%, the new MeeGo made a strong showing for honorable mention, which is noteworthy in its own right. The upstart MeeGo, a fusion of Nokia's Maemo and Intel's Moblin projects, garnered 10% on its own. Then, add the 7% of you who selected Maemo and 3% who selected Moblin individually to arrive at a healthy 20%. Meanwhile, old-timer Symbian missed the cut.

Best Desktop Environment

TIE: **GNOME** and **KDE**

The results from the Best Desktop Environment make more intuitive sense to us (and to this KDE fan) than in the previous two bouts, when GNOME edged out KDE by surprisingly healthy margins. In the 2010 battle royale, KDE jiu-jitsu-ed GNOME and gobbled 7% of its lunch (as well as 3% of Xfce's) to even the desktop score from last year. No other desktop surpassed the 3% mark.



BEST WEB BROWSER

Firefox

Honorable Mention: Chrome

In the 2009 Readers' Choice Awards, 87% of you voted to seat the Mozilla Firefox browser comfortably on the throne of Best Web Browser. Meanwhile, Google Chrome was just making its debut on the Linux stage (most commonly in the form of the CrossOver Chromium Project) and began appearing on your "to tinker with" lists. We suggested last year that by awards time in 2010, you should "look for an inevitable battle royale if Google can deliver a polished Chrome for Linux in time for you to give it a test-drive". Well, folks, that battle has ensued, and the era of unchallenged Firefox supremacy is over. Chrome leaped from a barely perceptible 0.35% of the vote in 2009 to 24% this year.



BEST E-MAIL CLIENT

Thunderbird

Honorable Mention: Gmail Web Client

In 2009, we suggested the apparently inevitable decline of the desktop e-mail client in favor of Web-based clients like Gmail. It looks like the official LJ Magic 8-Ball was in need of another shake, for Thunderbird handily has won the Readers' Choice Award for Best E-mail Client for the fourth consecutive year. Surprisingly, the previously upward trend for Gmail hit a ceiling (at least for now), and it lost a few points from last year, at the primary expense of Kmail, which came back onto your radar screens to garner a healthy 14% of the vote.



BEST MICROBLOGGING CLIENT

Gwibber

Honorable Mention: Choqok

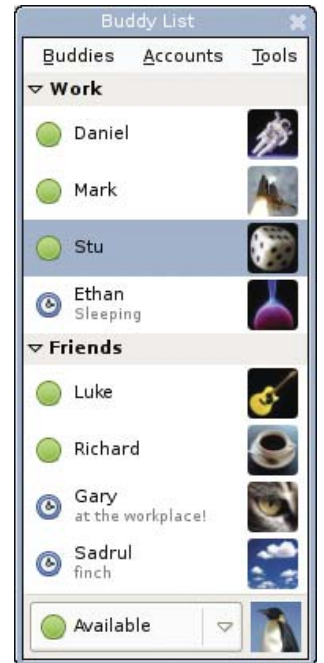
This new category for 2010, Best Microblogging Client, also would have the top five entrants in Best-Named Linux Application. The hands-down winners would be Gwibber, Choqok, Nitwit and our personal favorite, Spaz. Despite parity regarding the name-related coolness factor, the generalist and GNOME-based Gwibber easily took the crown for Best Microblogging Client, followed by the more specialist KDE-based Choqok.

BEST IM CLIENT

Pidgin

Honorable Mention: Skype

Despite our redubbing of this category from Favorite Communications Tool to Best IM Client, the results changed only slightly. For the third year in a row, the no-protocol-left-behind Pidgin Internet Messenger took top honors—with an identical 43% share of your votes vs. last year to boot. In the honorable mention column, the closed-source but so useful Skype took top honors, dropping a point from 2009 (17% vs. 18%).



BEST IRC CLIENT

Pidgin

Honorable Mention: XChat

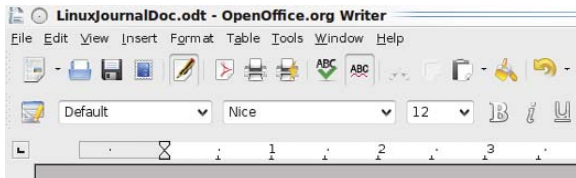
Although you have plenty of choices when it comes to IRC chat clients, the overwhelming majority of you stay put on Pidgin or fire up the multiplatform XChat for your IRC-based chats—39% of you prefer Pidgin and 33% prefer XChat.



BEST OFFICE SUITE

OpenOffice.org

No shake-ups here, gang. OpenOffice.org remains your uncontested choice for Best Office Suite, and no program even passed the 10% threshold to warrant honorable mention. Keep your eye on Google Docs though, because it showed up this year for the first time with 8% of the vote. In this as in most categories, cross-platform capability appears to boost a program's popularity significantly. Are Web-based apps going to take over, or will desktop apps remain dominant in this space? This area will be interesting to watch.

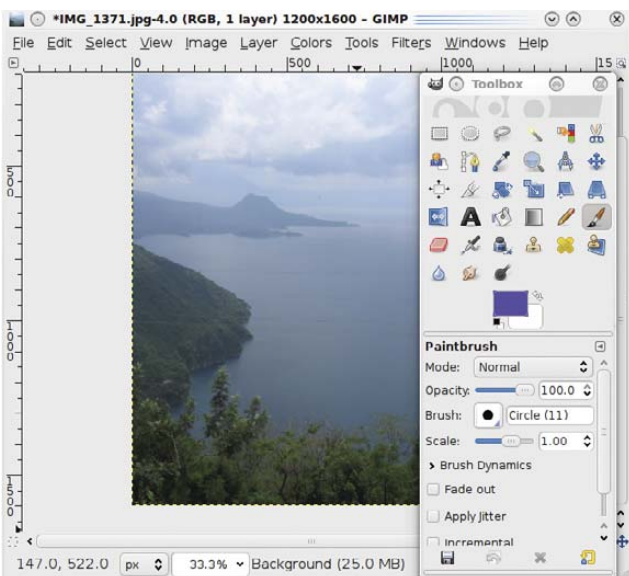


BEST SINGLE OFFICE PROGRAM

OpenOffice.org Writer

Honorable Mention: AbiWord

Given that there are many more office applications than office suites, we created this new category (Best Single Office Program) to understand the nuances of our community's work habits better. Although the OpenOffice.org apps Writer and Calc both performed well, with Writer winning the category handily with 39% of the vote. AbiWord from GnomeOffice also helps you get your work done quickly and effectively.

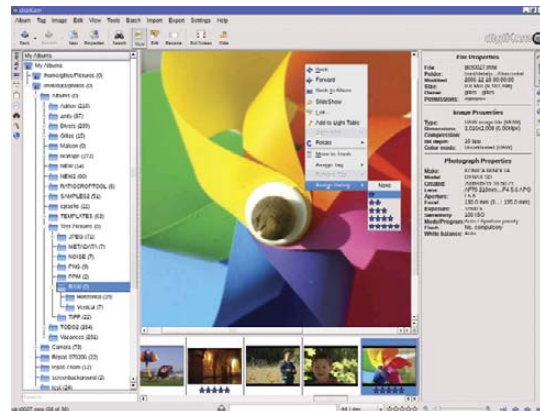


BEST DIGITAL PHOTO MANAGEMENT TOOL

digiKam

Honorable Mention: Picasa

The back-and-forth tussle between digiKam and Picasa is looking more and more like a WWE SmackDown. In 2008, the two photo apps were tied for your favorites in the crowded category of Best Digital Photo Management Tool. Then last year, Picasa gave digiKam a royal piledriver and racked up nearly triple the votes of its poor rival. But this year, digiKam had a surprise Diving Bulldog up its sleeve and had enough energy left to pin Picasa to the mat for the win.



BEST GRAPHICS/DESIGN TOOL

GIMP

Honorable Mention: Inkscape

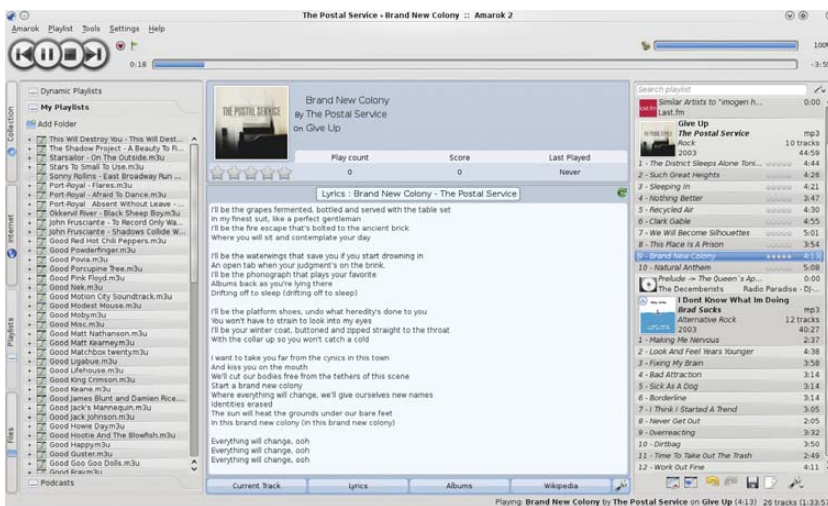
This year's plebiscite features identical award winners in the Best Graphics/Design Tool category—GIMP for the win and Inkscape for honorable mention. The only difference from last year is that GIMP inched down a few points and Inkscape up a few points.

BEST AUDIO PLAYER

Amarok

Honorable Mention: Rhythmbox

Your preferences for Best Audio Player this year are consolidating around two favorites, namely the winner, Amarok, and the sole honorable mention, Rhythmbox. Previously, this category was more crowded with contenders. However, Amarok is one of those apps that is riding the wave of resurgence we're seeing in KDE and its related applications. It doesn't hurt that Amarok is one kick-butt music app too.



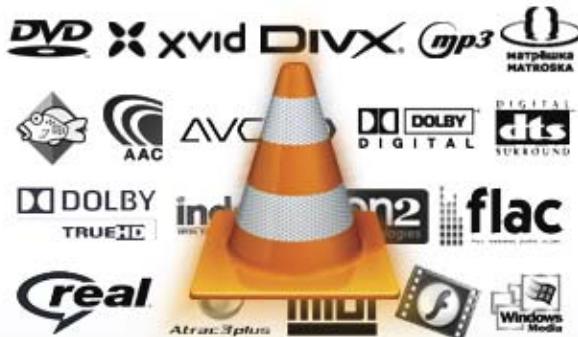
BEST MEDIA PLAYER

VLC

Honorable Mention: MPlayer

After winning the Best Media Player award for 2009 by a single percentage point, there's no looking back for VLC, which won again in 2010, this time with more than triple the votes of its nearest rival, MPlayer.

VLC's attraction could be the fact that it plays nearly any audio (or video) format you send its way, as well as its cross-platform capabilities. If you happen to find yourself in front of a Mac or Windows (or BeOS!) box, you're never far from "the cone".



BEST AUDIO TOOL

Audacity

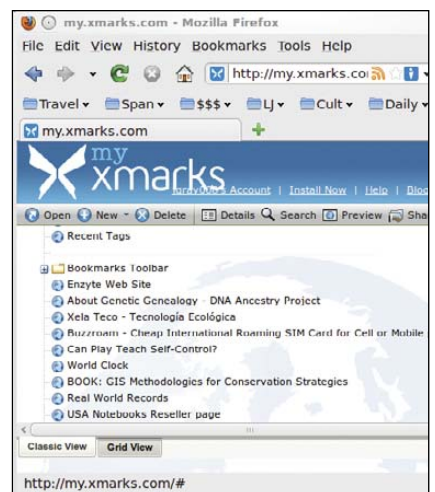
In the Best Audio Tool category, the program Audacity becomes more dominant every year, winning the 2010 award with a resounding 81% of your preferences, up 8% from last year. The popular cross-platform sound recorder and editor has little competition—only 8% of you consider Ardour your favorite audio tool.

BEST BOOKMARK SYNCING TOOL

Xmarks

Honorable Mention: Delicious

Nobody works on more machines than a Linux geek, so we're perfect guinea pigs for rating the product in this new category, Best Bookmark Syncing Tool. Your two favorites are Mitch Kapov's Xmarks (formerly Foxmarks) and Yahoo!'s Delicious (er, deli.icio.us). Firefox's own Weave tool fared well, but its Firefox-centricity may keep it from jousting at a higher level with its multibrowser-capable rivals.



BEST ON-LINE COLLABORATION TOOL



Google Docs

Honorable Mention: Wikis

On the surface, it's somewhat counterintuitive how you love Google Docs as an on-line collaboration tool but merely like it as an office suite. (See the results for Best Office Suite above.) Perhaps it's because you use OpenOffice.org more for your day-to-day work tasks but use Google Docs when the task specifically calls for collaboration? Despite your penchant for Google's tools, a solid number of you turn to wikis to share information with your dispersed colleagues.

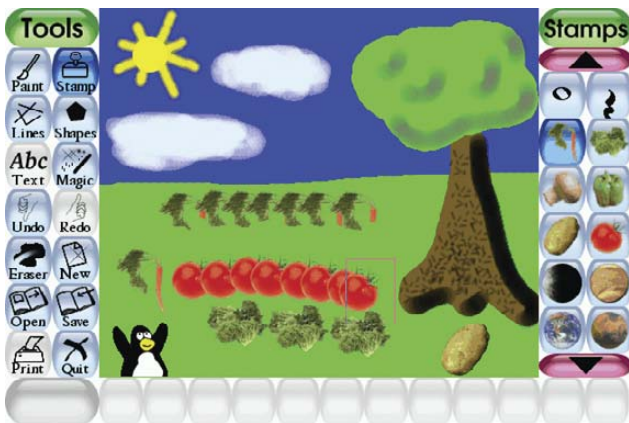
BEST CLOUD-BASED FILE STORAGE



Dropbox

Honorable Mention: Ubuntu One

For creating a redundant off-site backup copy of your important files, your tendency is to choose the popular Dropbox and Ubuntu One services. Knowing how we Linux geek cheapskates work, I bet you're signed up for the free 2GB accounts—40 separate ones to back up your 80GB drive.



BEST KID-FRIENDLY APPLICATION

Tux Paint

Honorable Mention: GCompris

The winner of the new category Best Kid-Friendly Application is Tux Paint, a gleefully fun drawing program for kids. Reading your comments, we also know a good number of you have managed to get Linux into your local school, so you've probably installed Tux Paint free of charge for the students to use as a tool for art instruction. On that same install, a sensible companion would be GCompris, a somewhat more diverse set of educational applications for children.

BEST GAME

Frozen Bubble

Honorable Mention: Doom and Battle for Wesnoth (tie)

It is beginning to seem that no game will ever knock *Frozen Bubble* from its lofty perch as Best Game. *Bubble* has won the title every time. Not only does *Frozen Bubble* lure you in with its pure simplicity, but those penguins are just too darn cute! Take note though that the *Battle for Wesnoth* has won honorable mention for the first time ever.



BEST DATABASE

MySQL

Honorable Mention: PostgreSQL

We'll have to do something about the LAMP acronym if you ever give up MySQL, which remains your choice for Best Database three years running. Nevertheless, we've seen a bit of erosion in support for MySQL toward your honorable mention pick of PostgreSQL, which has gained ground every year. SQLite may soon find itself on the board with the heavy hitters if it keeps climbing as well.

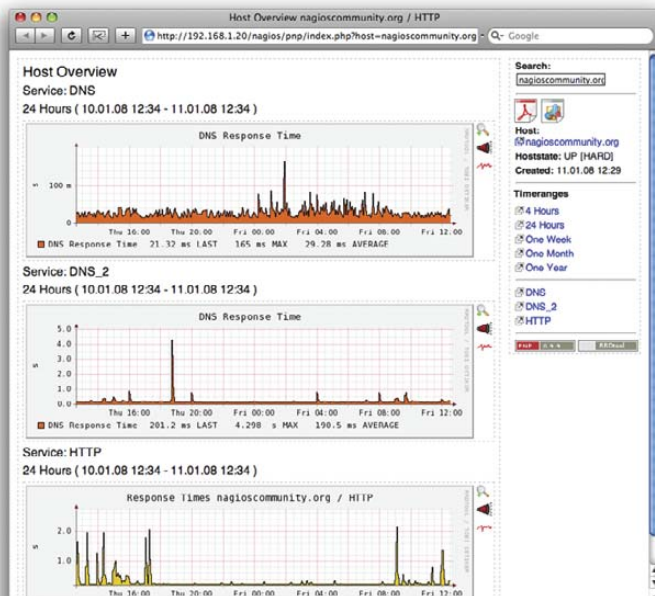


BEST BACKUP SOLUTION

rsync

Honorable Mention: tar

Here's one case where we retrenched and reduced our categories, namely combining Best Backup System and Best Backup Utility into a single category called Best Backup Solution. Your choice for champion in this category was rsync, the winner of Best Backup Utility in 2009, followed by tar. Last year's winner and honorable mention in the Best Backup System category, Amanda and Bacula, respectively, also fared well in the votes.



BEST VIRTUALIZATION SOLUTION

VirtualBox

Honorable Mention: VMware

If we had an award for Most Upwardly Mobile Linux Solution, it just might be VirtualBox. You've been favoring VirtualBox more and more with each passing year, such that a full 55% of you chose it as Best Virtualization Solution this year, which is up an impressive 23% from last year and more than double VMware's result of 23%.



BEST MONITORING APPLICATION

Nagios

Honorable Mention: Zabbix

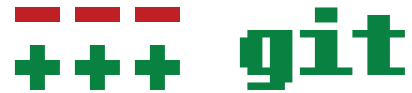
When speaking of upwardly mobile Linux apps, we should include Zabbix in the same breath, the classic monitoring application that could. Coming out of the tiny country of Latvia, the enterprise-class monitoring solution Zabbix went from also-ran last year to contender this year in the category Best Monitoring Application. Of course, let's not forget that 34% of you still favor the winner Nagios. However, Nagios is down 17% from last year's tally. Zabbix definitely is worth keeping on your radar.

BEST REVISION CONTROL SYSTEM

git

Honorable Mention: Apache Subversion

Your two top picks for the Best Revision Control System are this year's winner, git, and last year's winner (and this year's sole honorable mention), Apache Subversion. In case you haven't done so already, you might want to investigate the origins of git, which was created by Linus Torvalds. The source of its name will give you a chuckle.



BEST PROGRAMMING LANGUAGE

Python

Honorable Mention: C++

Despite the tough field of contenders, Guido van Rossum's Python won Best Programming Language for the second straight year. Your votes came down nearly exactly the same as last year, with C++ in second place.

BEST SCRIPTING LANGUAGE

Python

Honorable Mention: PHP

Two years ago, we tried to make our own distinction as to which languages were programming languages and which were scripting languages. We set up an elaborate set of criteria and attempted to justify our position. Well, you didn't appreciate our micromanagement, so we scrapped that idea for good. Now you decide which is which, and you have decided that Python is both the best programming language and scripting language out there.



BEST IDE

Eclipse

Honorable Mention: KDevelop

Eclipse is batting 1.000 in the Best IDE category, winning both of its two years in existence. You've told us you like how Eclipse lets you work in a lean environment and add and subtract an incredible array of functionality via plugins.

BEST OPEN-SOURCE CONFIGURATION MANAGEMENT TOOL

Puppet

Honorable Mention: OpenQRM



The new category Best Open-Source Configuration Management Tool turned out to be one of the most crowded fields, yet Puppet Labs' Puppet application managed to pull ahead of the pack. Perhaps it's how Puppet considers each piece of infrastructure as code that has helped you simplify new configurations and helped reduce the time you spend on mundane tasks.



BEST PLATFORM FOR DEVELOPING RICH INTERNET APPS

Adobe AIR

Honorable Mention: JavaFX

Adobe AIR seems to be solidifying itself as your choice for Best Platform for Developing Rich Internet Apps, winning the category both this year and last. AIR is trending upward, rising a few percentage points this year vs. 2009.



BEST PACKAGE MANAGEMENT APPLICATION

apt

Honorable Mention: Synaptic

Despite your continued overwhelming preference for Ubuntu and siblings, the success of package managers typically associated with these distros is a tad enigmatic. Let's parse it. You prefer Ubuntu's default apt as Best Package Management Application but a full 10% less than last year. At the same time, 8% more of you like Synaptic than last year to give it honorable mention again, so it's reasonable to assume that a good number of the Ubuntu-ite tribe are trying Synaptic and digging it. We can comprehend all of that, but KPackageKit, Kubuntu's default package manager, wasn't even on the radar despite KDE's surge to tie GNOME for Best Desktop Environment. What's going on here?

BEST CONTENT MANAGEMENT SYSTEM

WordPress

Honorable Mention: Drupal

Your narrator had a total blast building a sweet, slick Web site with WordPress, so he's totally in agreement your decision to grant it Best Content Management System for 2010. My experience was positive, and I put together a great design despite nearly no skill in graphic design. We at *Linux Journal* also gave our vote to honorable mention winner Drupal, the platform we have used for the previous and the recently updated LinuxJournal.com.

BEST LINUX-FRIENDLY WEB HOSTING COMPANY

GoDaddy.com

Honorable Mention: Contegix

Although we have a clear winner for the Best Linux-Friendly Web Hosting Company for 2010, GoDaddy.com, the reality is that five companies have been jockeying for the title. Although GoDaddy.com won in 2008, it fell out of the winner's circle in 2009. Then, there's your honorable mention pick, Contegix, which was the winner last year. Companies like Dreamhost, Rackspace and 1&1 also have been in contention every year.

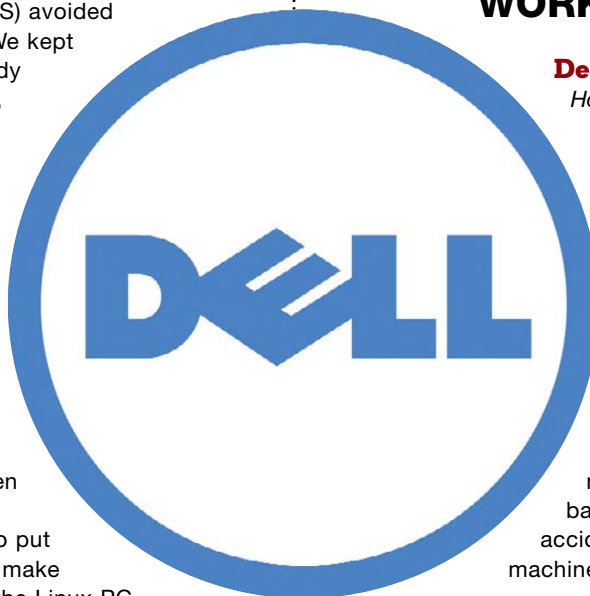


BEST LINUX LAPTOP VENDOR

Dell

Honorable Mention: Lenovo

For years, Dell (at least in the US) avoided Linux-based PCs like the plague. We kept hearing how the market wasn't ready yet. We kept saying "Sheesh, guys, build it and they will come." Well, now they've built it, and we have come—in droves! Dell dips its influential toe in the water and suddenly gets more votes than anyone for Best Linux Laptop Vendor. We're very curious to know how many of you are buying Dell laptops preinstalled vs. self-installation. Are we right to assume you're doing more of the latter? Our own hats are tipped graciously to Lenovo, who has taken arguably more risks than any other Windows-dominant laptop maker to put out great Linux-based laptops and make them mainstream. Also, we salute the Linux PC specialists who have been configuring our machines since the Precambrian. We hope you don't forget them.



BEST LINUX DESKTOP WORKSTATION VENDOR

Dell

Honorable Mention: Hewlett-Packard

The world's two biggest PC makers, Dell and HP, are your two top choices for Best Linux Desktop Workstation Vendor. This makes perfect sense, as installing Linux on desktops remains more trouble-free than doing so on laptops. Note that a spunky company, System 76—which proudly peddles Linux machines and doesn't make you enter through some backdoor link, only to find you've accidentally ordered a Windows machine—also did well in the voting.

BEST LINUX SERVER VENDOR

IBM

Honorable Mention: Dell

In the Big Iron category, Best Linux Server Vendor, you gave more votes to IBM than any other company. It makes sense given IBM's long-term business strategy involving Linux. In 2009, Dell was the winner here, and this year, it placed a close second behind Big Blue.



BEST LINUX BOOK

Linux in a Nutshell

Honorable Mention: *Running Linux*

Given the hundreds of books on Linux-related topics, it was a Herculean task for any single book to win the category Best Linux Book. Nevertheless, we asked you to write in your favorite title, and the classic work *Linux in a Nutshell* (E. Siever, et al., O'Reilly) was your top pick, acquiring more than double its nearest competitor, *Running Linux* (Dalheimer and Welsh, O'Reilly). We're wondering though, why more of you didn't write in *Just for Fun: The Story of an Accidental Revolutionary* (Torvalds and Diamond, Harper) or *The Cathedral and the Bazaar* (Raymond, snowballpublishing.com). Who needs another copy of the syllabus for official Linux geeks?

BEST LINUX JOURNAL COLUMN



Hack and / by Kyle Rankin

Honorable Mention:
Paranoid Penguin
by Mick Bauer

Choosing the winner for Best *Linux Journal* Column is just as Herculean as with Best Linux Book, except the problem is not the abundance of quantity but rather extreme quality of each offering. Kyle Rankin's *Hack and /* column has become the page that more of you flip to first than any other. (Incidentally, Kyle's column has been trending upward for some time—he tied for winner last year.)

BEST BRAND OF VIDEO CHIPSET

NVIDIA

Honorable Mention: *ATI*

Before wrapping up, let's get back to some of your favorite gear. This year, we introduced the new category Best Brand of Video Chipset, which was won handily by NVIDIA. Although we as a community are frustrated with NVIDIA's proprietary drivers, we can rejoice in the performance and Linux support, which is better than most.



BEST LINUX SMARTPHONE

Nokia N900

Honorable Mention: *HTC Nexus One*

No *Linux Journal* award show would be complete without a Nokia N-series device, and the 2010 Readers' Choice Awards is no exception. The Nokia N900 takes the award for Best Linux Smartphone. We're not too surprised that you chose the N900 given that it's the most early-adopter phone out there. That's us. The honorable mention in this category is an Android: HTC Nexus One.

BEST LINUX-BASED GADGET

Amazon Kindle

Honorable Mention:
TomTom Navigation System

The device that made reading an e-book a viable option is the Linux-based Amazon Kindle, your winner for this year's Best Linux-Based Gadget.



BEST NEW OPEN-SOURCE PROJECT (RELEASED IN 2009 OR 2010)

MeeGo

Honorable Mention:
OwnCloud

For the category Best New Open-Source Project released in 2009 or 2010, we asked you to

write in your favorites without any prompting from us. Although the responses were numerous, the winner is MeeGo! A little bit of Moblin in your Maemo, or Maemo in your Moblin, this merger of two mobile operating systems is quite exciting. It's fairly new, but will this open-source powerhouse become the next big thing? Your votes seem to imply it, we'll have to wait and see.



BEST NEW COMMERCIAL APPLICATION (RELEASED IN 2009 OR 2010)

Fluendo DVD Player

The race for Best New Commercial Application was neither as crowded nor closely fought as the one for Best New Open-Source Project. The hands-down winner was Fluendo DVD Player. Although we Linux geeks hate to pay, the product makes playing any DVD possible, fully functional (and legal).



PRODUCT OF THE YEAR

Android

Honorable Mention: KDE

We close with the category that requires a drumroll (drumroll, please): the 2010 *Linux Journal* Readers' Choice Product of the Year. And, the winner is Android! The open-source operating system from Google has proven to be a formidable opponent in the mobile-phone world. Because it's getting Linux into the hands of people who don't even realize they're using Linux, we can see why it's your choice for Product of the Year. We look forward to Android's 3.0 release, which Google claims will support tablet computers as well. Perhaps when version 4.0 rolls around, we'll have androids running Android!

Be sure to check out our Web site for an expanded write-up of this year's Readers' Choice Awards, including third-, fourth- and fifth-place winners: www.linuxjournal.com/rc10. ■

James Gray, Products Editor for *Linux Journal*, is an environmental education specialist at the University of Wisconsin-Parkside. When not helping people of all ages have hands-on learning experiences with nature, he enjoys tinkering with open-source GIS, mapping and image-analysis apps. He and his family reside in Kenosha, Wisconsin.

Parallel Programming with NVIDIA CUDA

Using hardware acceleration via General Programming on stock GPUs (GPGPU), I've sped up my algorithms by more than tenfold. This article shows how you can achieve these results too! ALEJANDRO SEGOVIA

Programmers have been interested in leveraging the highly parallel processing power of video cards to speed up applications that are not graphic in nature for a long time. Here, I explain how to do this with the CUDA API from NVIDIA. If your GPU is not from NVIDIA, you are not out of luck, as the same can be achieved with other APIs, such as the ATI-based Stream SDK or OpenCL.

GPGPU and Stream Processing

With GPGPU, general-purpose applications are executed directly on the streaming processors of video cards. Under the stream processing paradigm, a data set is named a stream. You can think of it much like "file streams" provided by an OS's pipe function.

Streams can be any isolated piece of data, such as a stream of business events or a set of scientific data. Parallel operations are applied on streams with operators, such as split, compute or merge. Figure 1 shows several streams of data and compute operators in parallel.

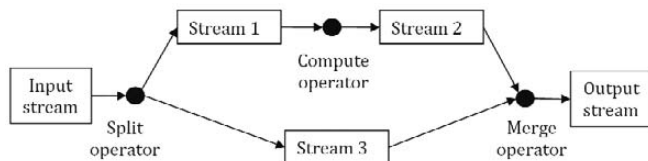


Figure 1. Stream Processing Diagram

Stream processing has been used successfully for general programming, including dataflow programming, financial calculation and industrial automation, just to name a few. Furthermore, system engineers and vendors such as Dell, ASUS, Western Scientific and Microway are building clusters of video cards that are similar to supercomputers, and they're available at a fraction of the cost of their CPU-based counterparts.

You can find many examples of real-life applications that were sped up using CUDA acceleration showcased by NVIDIA at www.nvidia.com/cuda.

Identifying an Algorithm to Parallelize

Now that I've brushed upon what CUDA and stream processing are, let's start looking into a couple compute-intensive algorithms you can use to give it a spin.

Vector fields are constructs employed in a variety of professions. In robotics, vector fields can help a mobile robot navigate

through a room. Let's define a destination and add one or more obstacles. A good scenario for testing CUDA consists in calculating a series of vectors that indicate the direction a robot should follow in order to reach its destination while avoiding all the obstacles present. The robot should also avoid local minima (see below). Figure 2 shows the robot and vector field (the green arrows are the "vectors").

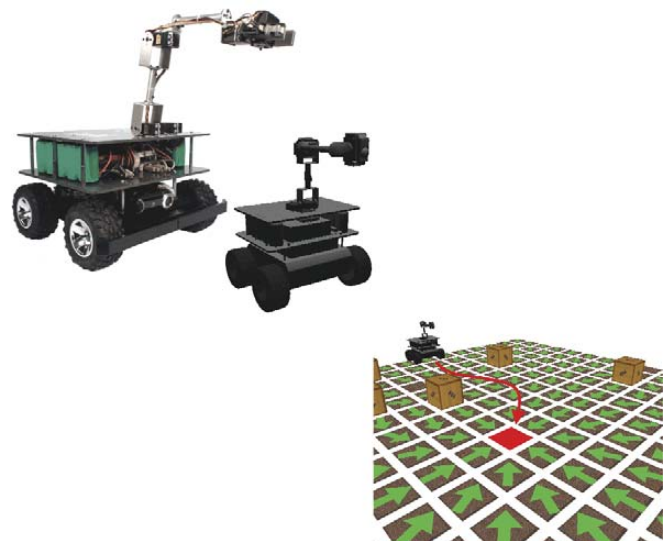


Figure 2. The mobile robot wants to reach the center of the board. The vector field shows the way.

I refer to the target point as an attractor and to obstacles as repulsors—the arrows point toward the attractor and away from repulsors (Figure 3). So, how do you calculate the vector field? The vector field is composed of a series of individual fields, one for each attractor and repulsor.

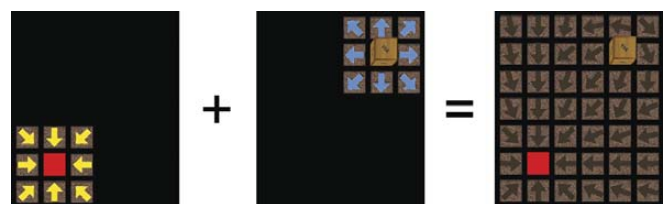


Figure 3. Attractor and Repulsor

Each individual field is calculated by computing the direction toward the attractor and away from repulsors at each point in the room. Once all of the vectors have been calculated, you obtain the complete vector field by adding them up.

For this example, I will have three streams and two compute operators. The list of attractors and repulsors will be used as the input stream. Then, a compute operator will be applied to it to obtain a second stream: the vector field. Finally, a second compute operator will provide another stream: the local minima field.

Some Problems Can Be Parallelized, Others Not So Much

Why is this a good demonstration of CUDA? When deciding whether an algorithm is a good candidate for parallelization, you should consider the following criteria:

- Is the problem compute-intensive?
- Can the problem be modeled as a stream process?
- Is the code independent of any shared resources?
- What sequences of code are independent of any other code?
- Can the data be represented as arrays of 32-bit objects?
- Are there no optimizations of the sequential algorithm possible?

In my case, the vector field may be large and could take a long time when evaluating the whole field. The path a robot should follow can be modeled easily with streams. There is no access to shared resources, and the computation of each element in the field is independent from all the others.

In terms of computation, robotic engineers usually constrain their algorithms to calculate only the part of the vector field that is needed at a given time, never evaluating the entire vector field. Next, I show you how you can use stream processing for calculating the whole vector field in real time. Let's get started.

From Sequential to Parallel

Let's develop a sequential version of a vector field calculation algorithm. The input stream is a list of attractors and a list of repulsors, as shown in Figure 2 and without obstacles (Figure 4). The output stream is a matrix called field. The attractors and repulsors are lists of 2-D points that indicate their positions.

A "field" matrix will hold your vector field data structure. Each element `field[y][x]` will hold a 2-D vector indicating where the robot should move toward when standing at point (x, y) . This vector will be the sum of vectors associated with each attractor and repulsor (Figure 5).

When processing each attractor, the associated vector will be pointing from the current point (x, y) toward the attractor's position. When processing each repulsor, the associated vector will be pointing away from the repulsor's position. Note that

Advertiser Index

CHECK OUT OUR BUYER'S GUIDE ON-LINE.

Go to www.linuxjournal.com/buyersguide where you can learn more about our advertisers or link directly to their Web sites.

Thank you as always for supporting our advertisers by buying their products!

Advertiser	Page #	Advertiser	Page #
1&1 INTERNET, INC. www.oneanddone.com	1	LOGIC SUPPLY, INC. www.logicsupply.com	25
ABERDEEN, LLC www.aberdeeninc.com	5	LULLABOT www.lullabot.com	45
ACE COMPUTERS www.acecomputers.com	73	MICROWAY, INC. www.microway.com	C2, C4
ASA COMPUTERS, INC. www.asacomputers.com	67	POLYWELL COMPUTERS, INC. www.polywell.com	79
CODERO www.codero.com	7	SERVERBEACH www.serverbeach.com	23
EMAC, INC. www.emacinc.com	37	SERVERS DIRECT www.serversdirect.com	9
GENSTOR SYSTEMS, INC. www.genstor.com	71	SILICON MECHANICS www.siliconmechanics.com	29, 49
GUTSY GEEKS www.gutsygeeks.com	74	TECHNOLOGIC SYSTEMS www.embeddedx86.com	19
INTEL www.intel.com	3	USENIX ASSOCIATION www.usenix.org/lisa10/lj	39
IXSYSTEMS, INC. www.ixsystems.com	C3	VERIO www.verio.com	35
LINODE, LLC www.linode.com	69	ZT SYSTEMS, INC. www.ztsystems.com	13

ATTENTION ADVERTISERS

March 2011 Issue #203 Deadlines

Space Close: January 3; Material Close: January 6

Theme: Linux in Your Pocket

BONUS DISTRIBUTIONS: Southern California Linux Expo (SCALE), USENIX FAST, SxSW, SHARE (Anaheim)

Contact **Joseph Krack**, +1-713-344-1956 ext. 118, joseph@linuxjournal.com

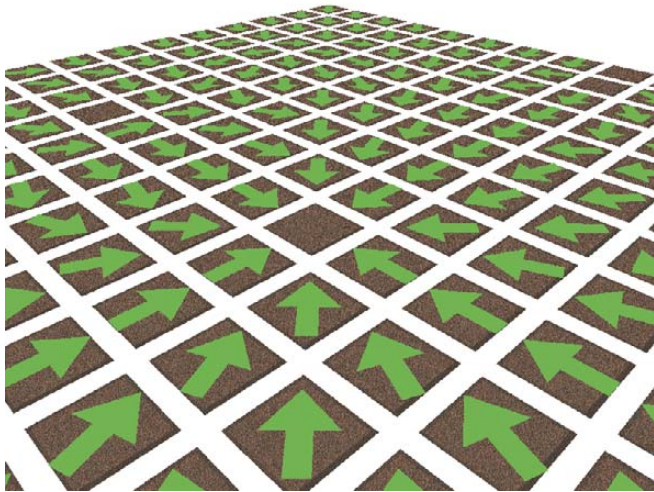


Figure 4. Composing Attractors and Repulsors



Figure 5. Vector Addition

the plus and minus operators are performing vector addition and subtraction. Sequential pseudo-code:

In Parameters: a list of attractors, a list of repulsors

Out Parameters: a zero-initialized vector field

calculate_vector_field_cpu(in attractors, in repulsors, out field):

```

for (y = 0 to height):
  for (x = 0 to width):
    for (attractor in attractors):
      vector = attractor - point(x,y)
      field[y][x] = field[y][x] + vector

    for (repulsor in repulsors):
      vector = point(x,y) - repulsor
      if norm(vector) <= 2:
        field[y][x] = field[y][x] + vector

return

```

Okay, so the sequential pseudo-code is ready. Now, let's partition the problem in order to use most of the processing cores on the GPU.

The calculation of each vector field element is independent from the calculation of the other elements. You can leverage this property to parallelize your algorithm. You can calculate each element of the vector field matrix in its own thread, effectively dividing the problem into smaller pieces.

Don't worry about the number of threads. Spawning as

many threads as possible when developing CUDA algorithms is encouraged by NVIDIA. It will allow the algorithms to scale across several generations of devices, automatically increasing throughput, as NVIDIA adds more and more processing cores to its video cards.

With this in mind, let's develop a parallel version of our previous algorithm. Parallel pseudo-code:

In Parameters: list of attractors, list of repulsors

Out Parameters: a zero-initialized vector field

calculate_vector_field_gpu(in attractors, in repulsors, out field):

```

x = blockIdx.x * BLOCK_SIZE + threadIdx.x
y = blockIdx.y * BLOCK_SIZE + threadIdx.y

```

```

for (attractor in attractors):
  vector = attractor - point(x,y)
  field[y][x] = field[y][x] + vector

```

```

for (repulsor in repulsors):
  vector = point(x,y) - repulsor
  if norm(vector) <= 2:4444444444
    field[y][x] = field[y][x] + vector

```

```

return

```

Notice I did away with both external for loops, and the points (x, y) are now calculated using a parallel statement.

The new pseudo-code is implemented as a kernel. A kernel is a function that executes on several GPU cores at the same time. Kernels are launched by a host program controlled from the regular CPU that configures the execution environment and supplies the parameters.

How does each thread know what position of the vector field it has to compute? This is where the blockIdx and threadIdx built-in CUDA variables come into place.

As you look at this code, it may not be obvious how this is a parallel implementation, but it's the blockIdx and threadIdx and the CUDA magic associated with them that makes it parallel. When the function is invoked, it actually is invoked multiple times using multiple threads, each thread calculating one part of the result (see the next section).

Assigning Work to Each Thread

When the host code sets up an execution environment, it has to determine how the processing cores will be assigned work. As part of its duties, the host must determine how threads will be arranged logically. CUDA allows developers to arrange their threads in a 1-D, 2-D or 3-D structure. It helps developers express design in a natural manner.

Think about the example algorithm. Let's use the GPU to fill a 2-D matrix with data. It would be very convenient if you somehow could assign each thread a "position" in a two-coordinate system, because each thread could use its assigned coordinates to decide which element to compute.

CUDA provides a mechanism through which developers can

specify how they want their threads arranged. The compiler takes care of the rest. This feature is available through what is known as a grid of thread blocks. In this example, the host will use a 2-D grid.

You probably are wondering, why a "grid of thread blocks" instead of a "grid of threads"?

Threads do not exist inside grids by themselves, but rather, they are arranged into thread blocks. Each thread is assigned an identifier within its block. Each block, in turn, is assigned an identifier within the grid. The built-in blockDim and threadIdx variables help determine the current thread and block identifiers. From within a kernel, these identifiers can be seen simply as a C structure containing the thread's x, y and z coordinates.

Using this mechanism, you can have each thread calculate a global thread ID and calculate the x and y variables in your parallel pseudo-code. The pair (x, y) determines which element of the matrix has to be computed. Because each thread will have different values for (x, y), every point of the matrix could, in theory, be computed at the same time if you have enough threads.

Putting Two Stream Operators in Sequence

Now, let's apply a second operation that detects local minima on the computed vector field. Local minima are those places where all vectors are converging (Figure 6). Flagging out the local minima will prevent the mobile robot from stopping in one of them with none of the vectors guiding it out.

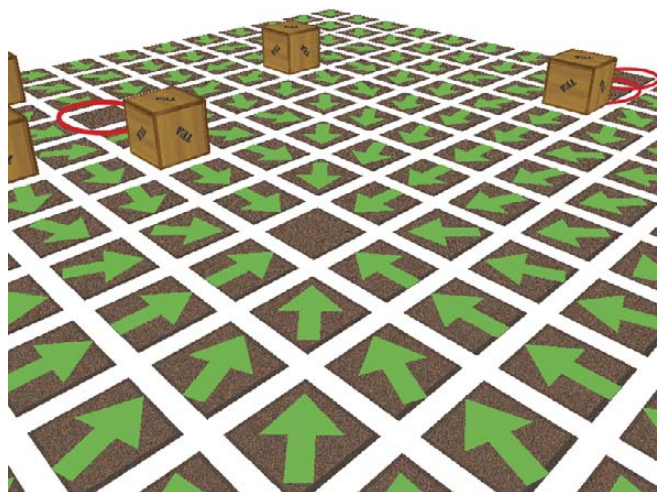


Figure 6. Three Local Minima the Mobile Robot Should Avoid

Under the stream processing model, operators can be daisy-chained: a second operator consumes the output of a first operator, much like the pipe operator of an operating system. In the example CUDA implementation, you will consume the vector field matrix stored in GPU memory. Sequential local minima detection pseudo-code:

In Parameters: calculated vector field, a decimal threshold
Out Parameters: a boolean matrix called "minima"

```
detect_local_minima_cpu(in field, in threshold, out minima):
```



ASA COMPUTERS

Want your business to be productive?

The ASA servers powered by the Intel® Xeon® Processor provide the quality and dependability to keep up with your growing business.

" Since 1989 Integration and service with pride "

ASA 1U Series



Intel® Atom™ processors D510 and D410 based platform and are designed for embedded industrial PC (IPC) applications. These quiet, energy saving solutions make ideal network appliances, print and email servers.

ASA 2U Series

QPI, Intel® Xeon® Processor 5500 Series in a high-density 2U form-factor are ideal for network infrastructure, front-end enterprise, and minimal downtime cluster server systems maximum upto 4 nodes in 2U.



ASA 3U Series

Server series excel under iSCSI/NAS/JBOD environments, the 3U servers support high availability storage and mission critical business applications.



ASA 4U Series

Optimized for enterprise-level high-capacity storage applications, features 36x (24 front + 12 rear) 3.5" Hot-swap HDD bays, reliable and hassle-free maintenance storage system.



ASA Blade Series

Blade server with, Highest computing density (20 DP nodes and 2.56TB mempry in 7U) Fastest and Most Cost-Effective Networking Solution (Infiniband DDR/QDR support)



E-mail - sales@asacomputers.com
Call - 1800-REAL-PCS



ASA Computers, Inc
645 National Ave,
Mountain View, CA 94043
www.asacomputers.com



**Powerful.
Intelligent.**

Intel, the Intel logo, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation in the U.S. and other countries.

```

for (y=0 to h):
  for (x=0 to w):
    minima[y][x] =
      (norm(field[y][x]) < threshold)? true : false

return
    
```

The sequential algorithm takes the vector field as input and fills in a Boolean matrix of the same dimensions with values "true" or "false", depending on whether the length is below a given threshold. Conversely, the matrix "minima" at position (x, y) indicates whether the norm of the vector located at (x, y) is less than the given threshold. Parallel local minima detection:

In Parameters: the calculated vector field, a decimal threshold
 Out Parameters: a boolean matrix called "minima"

```

detect_local_minima_gpu(in field, in threshold, out minima):
  x = blockIdx.x * BLOCK_SIZE + threadIdx.x
  y = blockIdx.y * BLOCK_SIZE + threadIdx.y
  minima[y][x] = (norm(field[y][x]) < threshold) ? true : false
    
```

The output is a field of Boolean values that indicates whether a given point is a local minimum.

Building Up a Test Benchmark

At this point, I have implemented four algorithms. You can, of course, download all the source code from our Web site for free and try them out yourself.

So, how does a CUDA algorithm stack up against its CPU equivalent? Next, I compare the parallel versions against their sequential counterparts in order to find out. The hardware used for the benchmark implementation includes:

- Intel Core 2 Duo E6320, running at 1.6GHz with 4GB of RAM.
- NVIDIA GeForce 8600GT GPU.
- Ubuntu Linux 8.10.
- CUDA version 2.2.

I implemented all four algorithms in one C++ program that can switch between the CPU and the CUDA versions of the algorithms dynamically. Not only does this make the benchmarking process easier, but it also is a good technique for developing programs that can fall back to the CPU on a computer

where CUDA is not supported.

Benchmark Results

Each of the benchmarks uses different vector field configurations, increasing the size of the field as well as the number of repulsors. The number of attractors always is set to just one. The size of the vector fields are: 16x16, 32x32, 64x64, 128x128 and 256x256. The repulsors are randomly distributed on the field with a ratio of one repulsor per 32 vector field points. Hence, the number of repulsors is 8, 32, 128, 512, 2048 and 8192.

Figure 7 shows the results of the benchmarks. I am using the notation "WxH/R", where WxH denotes the vector field's dimensions and R the number of repulsors present. The execution time is in milliseconds on a logarithmic scale (so a small difference in graph size is actually a much larger speedup than it appears to be visually).

How much faster is the GPU? The speedup is calculated by dividing the execution time of the sequential algorithm by the execution time of the parallel algorithm (Figure 8).

Computation times are the closest in the case of a small vector field. However, even in that case, we get a speedup of 2.5 times just by switching to the CUDA implementation

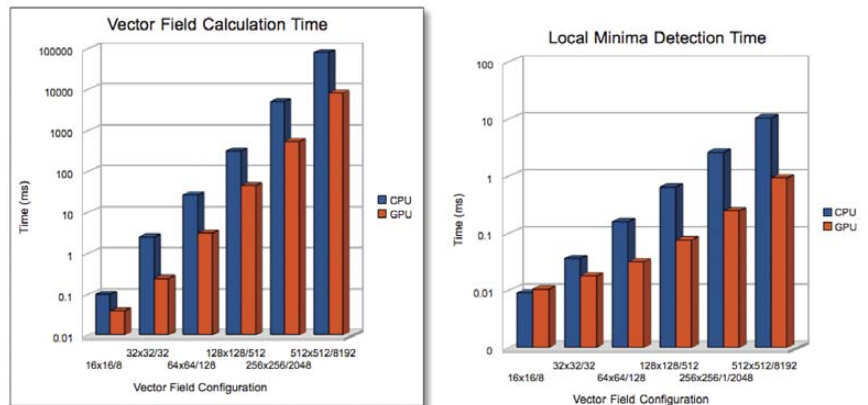


Figure 7. Calculation Times

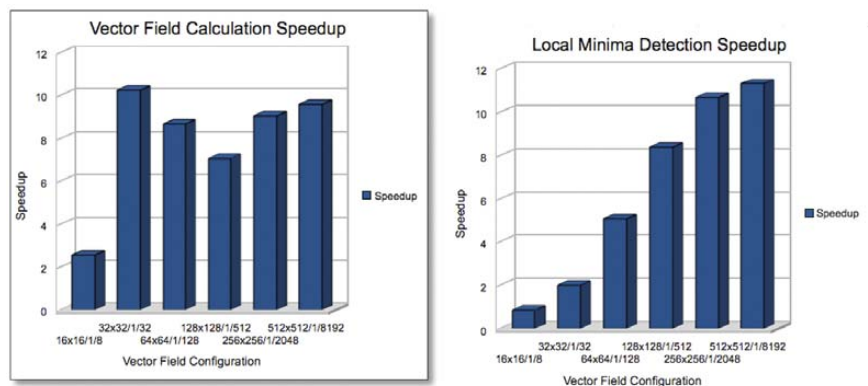


Figure 8. Speedup

of the vector field calculation. The local minima detection becomes interesting to parallelize only with slightly larger data sets that are more compute-intensive than smaller ones.

On average, the speedup is around eight times for our algorithms. In layman's terms, this means if you have a computation that takes one work day to complete, just by switching to CUDA, you can have your results in less than one hour.

This provides significant benefits for computations that require a user to run a computation several times while correcting the parameters each time. Such iterative processes are frequent, for instance, in financial models.

Conclusion

If parallelization of your algorithm is possible, using CUDA will speed up your computations dramatically, allowing you to make the most out of your hardware.

The main challenge consists in deciding how to partition your problem into chunks suitable for parallel execution. As with so many other aspects in parallel programming, this is where experience and—why not—imagination come into play.

Additional techniques offer room for even more improvement. In particular, the on-chip shared memory of each compute node allows further speedup of the computation process. ■

Alejandro Segovia is a parallel programming advisor for CoroWare. He is also a contributing partner at RealityFrontier. He works in 3-D graphic development and GPU acceleration. Alejandro was recently a visiting scientist at the University of Delaware where he investigated CUDA from an academic standpoint. His findings were published at the IEEE IPCCC Conference in 2009.

Resources

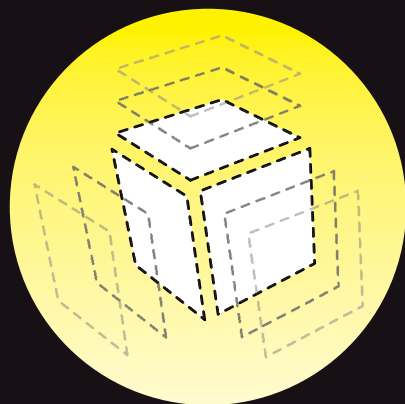
Hardware-Accelerated Computation: gpgpu.org

Source Code and Video: www.coroware.com/streamprocessing

Getting Started on CUDA: www.nvidia.com/cuda

ATI-Based Stream SDK: developer.ati.com/gpu/atistreamsdk

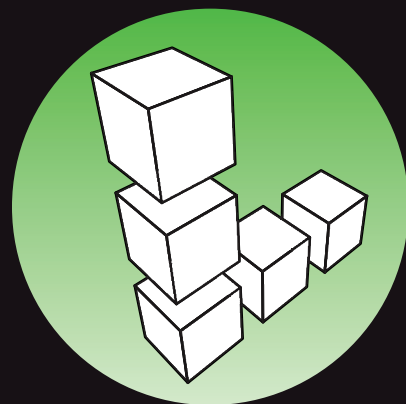
OpenCL Official Web Site: www.khronos.org/opencv



Develop.



Deploy.



Scale.

Full root access on your own virtual server for as little as \$19.95/mo

Multiple Linux distributions to choose from • Web-based deployment • Five geographically diverse data centers • Dedicated IP address • Premium bandwidth providers • 4 core SMP Xen instances • Out of band console access • Private back-end network for clustering • IP fail-over support for high availability • Easily upgrade or add additional Linodes • Free managed DNS

For more information visit www.linode.com or call us at 609-593-7103



linode.com

Getting Started with PiTiVi

The PiTiVi video editor is an intuitive video editor for creating your next masterpiece.

JONO BACON

Video editing on Linux has had a long and complicated history. Although Linux has long-bathed in image editing, music production and other creativity-enabling applications, the platform traditionally has struggled with video production. In the early days, the problem was pinned on the complexities of supporting different video codecs, but as codec support continued to improve, the spotlight instead was shone on video editors.

Unfortunately, video editors were largely divided into two camps: hugely complex, resource-hungry behemoths, such as Cinelerra, or over-simple, limited offerings such as Kino. Across the world Linux, users craved a middle ground, desperately hoping for a simple, usability-orientated editor that supports a wide variety of media formats fused with the ability to be used on real-world projects.

The PiTiVi Story

Three years ago at the GNOME developer conference, Edward Hervey, a French developer living in Barcelona, presented the first cut of his video editor project, PiTiVi. With it, Edward made a series of firm decisions. First, it was based on the GStreamer multimedia framework—arguably the most popular and recommended way to handle low-level multimedia operations and content. Second, he focused his project firmly on usability and ease of use. Edward never set out to produce a super-complex professional editing tool, but rather a tool focused on simple real-world projects, such as editing your honeymoon video and putting it on YouTube. Finally, Edward was willing to delay the development of PiTiVi to “do things right”.

The latter was particularly apt. It took him three years to get PiTiVi in a shape where it could be used for the “real” projects he targeted. Much of the reason for this was that as Edward hacked on PiTiVi, he would find bugs and missing features in GStreamer, so he would step away from the PiTiVi coal-face to fix the GStreamer bug or feature before returning back to PiTiVi. Although it was a frustrating and time-consuming process, Edward’s work paid off. GStreamer is an incredibly powerful and stable framework for building applications, so much so that I myself created a music multitracker project with it called Jokosher. Edward’s work not only generated a better GStreamer, but also a more powerful and mature PiTiVi.

Getting PiTiVi

PiTiVi is available for all major Linux distributions and is now bundled by default with Ubuntu 10.04 Lucid Lynx. You can find packages for these different distributions in most distribution archives, so use `apt-get`, `yum` or `emerge` to grab PiTiVi for your system. When you have installed it, you can click Applications→Sound & Video→PiTiVi to load it. If all else fails, you can download PiTiVi from its Web site at www.pitivi.org.

With PiTiVi ready to roll, you also need to ensure you have the right video codecs installed for the video formats you want to edit. This is as simple as ensuring you have the `gstreamer-good` package installed. If you want more codecs, but ones that are legally restricted in terms of redistribution, install the `gstreamer-ugly` package. Finally, there is another package with a set of work-in-progress codecs called `gstreamer-bad`, which you can try. Personally, I install them all so I have the widest codec coverage.

The Interface

Start PiTiVi by clicking Applications→Sound & Video→PiTiVi. First, let’s take a look at the PiTiVi (Figure 1).



Figure 1. The Main PiTiVi Interface

The interface consists of four main areas: the source list, previewer, timeline and toolbar.

The source list is the main white area on the left part of the window. This is where you can import the different video clips, photos and sound files that will be used in your project. You can either drag the files onto this white area or click the `Import clips...` button to load them. Whichever approach you use, each clip will appear in the source list area ready for use in your project. Test this by either dragging a video onto the source list or importing it with the `Import clips...` button and selecting a video. If for some reason PiTiVi can’t load your file (most likely due to it being an unsupported format), an error appears at the bottom of the source list.

The previewer is to the right of the source list and provides a place to view the video in your project. The black box is where you will see your video, and the buttons underneath are standard transport buttons to control playback. The previewer is used not only for playing back your edited project, but also for previewing clips in the source list. Test this by dragging the video you imported into the source list and dropping it on the black box in the previewer. Now, use the transport controls to play it, and you should see the

video play back. Click the different transport buttons to fiddle with the playback.

The timeline is the long area underneath the source view and previewer. This area is where you perform the editing on your project. The concept is simple: the timeline provides a literal timeline of your project with the far left being the very beginning of your video project and time increasing to the right. In the timeline, you can drop clips, cut them into pieces and arrange them in your desired order. The timeline also can be used to adjust volume, add pictures and overlay music over different parts. Test the timeline by dragging the video from the source list and dropping it on the timeline; the clip appears in the timeline, similar to Figure 2.

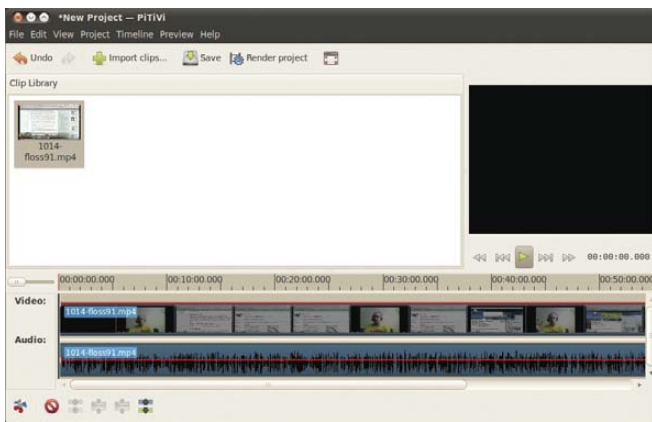


Figure 2. A Video Loaded into the Timeline

When a clip is loaded, there are two bars: the Video and Audio tracks. If you see only the second bar, you may need to click the clip in the timeline and drag it up to the video track to see both. Each bar represents exactly what it says—the different video frames and the audio content. This content is displayed separately, so you can remove one and not the other if you like. This is common for removing audio and replacing it with something else, such as a soundtrack.

Before moving on, the timeline has two additional special features: the ruler and the zoom. The ruler is the gray part with the numbers at the top of the timeline. If you click any part of the ruler, you can skip to a different part of the timeline, and that part of the project is shown in the previewer. When you click on the ruler, you can see the current position in the video by the red line that is drawn vertically on the timeline (this is called the playhead). To the left of the ruler is the zoom. This little slider can be used to adjust the scale of the timeline. This is useful for zooming out on the project to see it as a whole or zooming in closely to specific parts of a clip to cut a specific scene at just the right point.

The final area to look at is the toolbar, which is below the timeline. This line of buttons provides a simple palette of tools that you can use to edit content on the timeline. Let's explore some of these tools now.

Editing

All editing in PiTiVi takes place in the timeline. Fortunately, editing in PiTiVi is incredibly simple, and you really need to know how to



Linux - FreeBSD - OpenSolaris - etc.

Proven **Technology.**

Proven **Reliability.**

When you can't afford to take chances with your business data or productivity, rely on a Genstor server customized to your specifications.

POWER

PERFORMANCE

Fly into the Cloud with Genstor Systems



- Up to 48 cores in a 1U.
- AMD Opteron 6100 series.
- Single high-efficiency power supply.
- Up to 512GB DDR3 memory.
- Ideal as front end processing servers.



- Up to 12 cores in a 2U
- Dual redundant high efficiency power.
- Up to 96GB DDR3 memory.
- Server Power Capping via Intel Intelligent Power Node Manager.
- Ideal as front end processing and/or storage.



- Up to 4 GPU cards.
- Dual redundant high efficiency power.
- Up to 192GB DDR3 memory
- Up 24 cores using 2 CPUs.
- Up to 8 3.5" disks.

Genstor Systems, Inc.



780 Montague Express. # 604
San Jose, CA 95131
www.genstor.com
E-mail: sales@genstor.com
Phone: 877-25 SERVER
408-383-0120

Intel®, the Intel® Logo, Intel® Xeon®, and Xeon® Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

use only one main tool: the cutter.

Video editing is all about cutting video into different pieces and putting them next to each other to tell a story. To perform a cut in PiTiVi, place the playhead where you want the cut to happen (by clicking on the ruler), and click the scissors icon on the toolbar. The clip splits in half, and you can click on each clip to move it around. Now, if you move your mouse to the edge of one of the clips, the cursor changes, and you can shorten and lengthen the clip to taste. To get your cut at just the right place, you may want to use the ruler to zoom in and out of the timeline and resize until the clip is just right. If you get tired of constantly toggling the play button in the previewer to start and stop playback, use the spacebar as a shortcut for toggling playback.

One of the wonderful aspects of video editing is that it is so simple to perform, but the impact that is delivered lies in how you organize the clips, the timing between cuts, how those cuts line up with music and more. Each of these skills fundamentally boils down to cutting, trimming, moving and viewing the results in the previewer.

Swapping Out Audio

While editing, you may want to get rid of some of the audio and replace it with other audio, such as if you want a series of fast

One of the wonderful aspects of video editing is that it is so simple to perform, but the impact that is delivered lies in how you organize the clips, the timing between cuts, how those cuts line up with music and more.

cuts to music. You do this by splitting the audio from the video in a given clip and deleting the audio part. By default, your two clips are two different units and when you drag them, the video and audio are stuck together in each clip. To split the audio from the video in a given clip, first click on the clip on the timeline (it will turn a little darker to indicate that you selected it), and then click the far-right button on the toolbar (the button has an icon with two blocks and an up-and-down arrow). Clicking this icon unsticks the video and audio from the selected clip. Now, click the audio clip and only that will be selected. Delete this by clicking the red circle icon on the toolbar and only the audio clip will vanish, leaving the video present. If you now play back the video at that point in the timeline, you will hear no audio but see the video.

Now, import an audio file into the source list, and drag that to the timeline where the audio you just deleted was. You now have video over an entirely different audio track. Play with cutting the video into lots of different pieces, resizing them and placing them next to each other over the audio. Before you know it, you will have lots of fast-changing clips with your new audio soundtrack.

Performing Fades

When you add any clip to the timeline, a horizontal red line is

drawn over the duration of the clip. This is the fade line. With it, you can control where audio fades (on the audio track) or video fades (on the video track). An audio fade is where you smoothly adjust the volume of the clip from one level to another. As an example, at the end of a video you may want to fade your audio to silence. On the video track, the fade line lets you control how you fade from video to black. In the same example, at the end of a project, you could use this line to fade the video to black slowly.

Using the fade line is simple. Double-click at the point you want a fade to happen, and a small handle appears on the fade line. Now, click the handle and move it to adjust the fade. On the volume track, the higher the red line the louder the volume, and on the video track, the higher the red line the more video is visible instead of black. Next, click another part of a fade line and the red line connects the dots. This is how you do fades: add two points on a line and adjust their settings, and PiTiVi performs the fade for you. As an example, add a fade handle a little bit before the end of a video and one right at the end. Move the far-right handle to the bottom and the far-left one to the top, and the clip will fade out at the end.

One final note about fades is that they are tied to a clip and not part of the wider timeline. As such, if you add a fade to a clip and then move the clip around, the fade moves too. This makes it really simple to perform fades and then reorganize the clips in your project later.

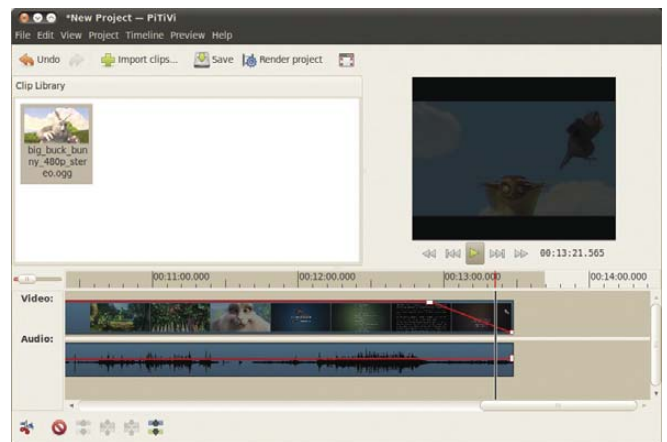


Figure 3. Performing fades is simple in PiTiVi.

Rendering Your Project

When you have finished editing your project, the final step is to combine all of your edits into a final video that you can share with others. This process is known as rendering your video. Fortunately, like everything else in PiTiVi, this is simple.

Click the Render project button and a dialog box appears. Click the Choose File button and select where you want your rendered file to be saved. You now can click the Render button, and your video will be rendered by default as an Ogg Theora video.

If you want to control what kind of video format and size is rendered, click the Modify button.

In this dialog box, set the visible size of the video in the Video Output section. There are a variety of defaults here, but you can



Figure 4. PiTiVi supports a wide variety of output formats.

select Custom if you want to specify your own size. In the Audio Output section, you can specify the audio quality of your video. Finally, in the Export to section, you can choose which codecs are used in your project. An important note here is that the Container file type can often contain a different type of codec. As an

example, the .avi container can have many different types of video codecs inside.

When you are happy with the settings, click the OK button to accept the settings, and then click the Render button in the render project dialog box to start rendering the video. Rendering can take a while, so go grab a cup of coffee. Afterward, you will have your fully edited video ready to view and show others. Congratulations!

Wrapping Up

In this article, I've explained how to get started with PiTiVi and how its simple and usability-oriented design and toolset can be used to create a range of different types of videos. Although simplicity sometimes can be confused for lack of capability, PiTiVi certainly can meet the needs of many home videos for sharing with friends and family.

Although PiTiVi is simple and flexible, the project is by no means standing still. The team already is working on transitions and titles for videos as we speak. PiTiVi has a rosy future, and so, therefore, does video on Linux. Be sure to let *Linux Journal* know what kind of videos you are making with it. Good luck! ■

Jono Bacon is the Ubuntu Community Manager at Canonical, author of *The Art Of Community* published by O'Reilly, founder of the Community Leadership Summit and co-presenter on *Shot Of Jaz* and *FLOSSWeekly*.

GPU Computing Specialist

Workstations with up to 1792 GPU cores



4U Servers with up to 3584 GPU cores



Preconfigured CPU/GPU Clusters



WCSA Contract # B27157

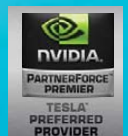
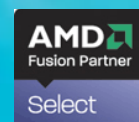


GSA Schedule Contract # GS35F-0400T



US Dept. of Energy BPA # DE-EM0000349

See Ace at Booth #2204



TECH TIPS

► Sonar Ping

I use this simple script when troubleshooting network problems on machines when I can't see the screen—for example, when I'm under a desk, wiggling Ethernet cables to find a bad one. When I hear the pings, I know it's fixed. Or, you can use this to drive your coworkers nuts by running it on their machines and sending single pings at random times during the day (or perhaps substitute moo for ping).

I made this into a script because I can't remember that big long line, and I would hate to type it in a lot. Here's sonar.sh:

```
#!/bin/bash
#
# Written by Mike Studer a long time ago
# Make sure you obtain a nice submarine ping sound.
# ie., ping with an echo (sonar.au used here)

/usr/sbin/icmpinfo -vv | \
  /usr/bin/nawk '$4 == "ICMP_Echo"
  {print $0;
  system("/usr/bin/aplay -q ~/sounds/sonar.au")}'
```

You need to install icmpinfo and aplay to use this.



The first and only radio show broadcast in the USA dedicated exclusively to spreading the word about the LINUX OPERATING SYSTEM and FOSS.

GUTSY GEEKS
COMPUTER SHOW

gutsygeeks.com

Usage: Run this on the machine on which you want to make noise (test):

```
sudo sonar.sh
```

Run this on a machine that is trying to get to the test machine for a nonstop ping barrage:

```
ping {testmachine}
```

For a single ping, run this:

```
ping -c 1 {testmachine}
```

—MIKE STUDER

► Get Even More from Less

In addition to viewing text, the less command can be used for viewing nontext files. This is done by using less' ability to invoke a preprocessor for input files. These preprocessors then can change the way the file's contents are displayed. For example, suppose you had a script lesspipe.sh:

```
#!/bin/sh
case "$1" in
  *.tar.gz) tar -tzvf $1 2>/dev/null
  ;;
  esac
```

Make sure the script is executable, and set the LESSOPEN environment variable to:

```
LESSOPEN='|/path/to/lesspipe.sh %s'
```

Now you can use less to view the contents of .tar.gz files:

```
$ less autocorrect.tar.gz
-rwxrwxrwx raogr/raogr 84149 2009-02-02 03:20 autocorrect.dat
-rwxrwxrwx raogr/raogr 443 2009-02-02 03:21 generator.rb
-rwxrwxrwx raogr/raogr 181712 2009-02-02 03:21 autocorrect.vim
```

More-sophisticated versions of lesspipe.sh are available. You already may have a version installed, or you may have the lessopen.sh script installed. If not, search the Internet for lesspipe.sh. With the more-sophisticated versions, you can do things like this:

```
$ less knoppix_5.1.1.iso
CD-ROM is in ISO 9660 format
System id: LINUX
Volume id: KNOPPIX
Volume set id:
Publisher id: KNOPPER.NET
...
/KNOPPIX
/autorun.bat
/autorun.inf
```

```
/autorun.pif
/boot
/cdrom.ico
/index.html
/KNOPPIX/KNOPPIX
/KNOPPIX/KNOPPIX-FAQ-EN.txt
```

—GURURAJ RAO

► Keep Laptop Temperature under Control

I work all the time with a laptop, and as you all know, from time to time laptops can get hot. When you're actually using it as a "lap"-top, or when you're close enough to hear the fans, you know when it's heating up. But, when the conditions are such that you don't realize it's heating up, your laptop can get pretty hot. And, you've all heard the stories about laptops catching fire.

The following script monitors the temperature and slows down your system when it gets too hot. The script should be run as root from cron every minute or so. You need to install cpufrequtils to get it to work:

```
#!/bin/bash
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/local/bin:

# Get the temp of the core 0
core_0=`acpi -t | awk {'print $4'} | head -n 1`

# Get the temp of the core 1
core_1=`acpi -t | awk {'print $4'} | tail -n 1`

# Round the result of core_0
convert_0=$(echo "scale=0; $core_0/1.0" | bc)

# Round the result of core_1
convert_1=$(echo "scale=0; $core_1/1.0" | bc)

# Set maximum permissible temperature.
max=90

# Set temperature at which the CPU frequency can
# be increased again (if needed).
min=68

if (( $convert_0 >= $max )) ; then
    # Too hot, slow down to 800MHz.
    cpufreq-set -f 800
    echo "CPU temp higher than desired!!!" | \
        mail -s "CPU temp too high, set frequency to half" root
elif (($convert_0 <= $min)) ; then
    # Cooled down, allow frequency to increase again if needed.
    cpufreq-set -g ondemand
fi
```

As you can see, in the script, I actually use the temperature only of core 0, because I know that this core tends to overheat before core 1.

—ALBERTO

LINUXTM JOURNAL

Linux News and Headlines Delivered To You



Linux Journal
topical RSS feeds
AVAILABLE

http://www.linuxjournal.com/rss_feeds



KYLE RANKIN

Bill and Kyle vs. LJ Readers

This month, Kyle and Bill decided to do something a little different—they're taking on some of *your* favorites in the 2010 Readers' Choice Awards (see page 52 for the article).



BILL CHILDERS

READERS: best Linux distribution: Ubuntu; honorable mention: PCLinuxOS.

BILL: I've got to give kudos to Ubuntu. These folks continue to be the one to beat, year after year. I've been using Ubuntu since it was no-name-yet.com, and it's just gotten better with each release.

KYLE: I have to agree with the readers on this one too. Despite the hype and anti-hype, Ubuntu still is a fine overall distribution both on the desktop and as a server, and it's the one I generally recommend.

READERS: best mobile OS: Google Android; honorable mention: MeeGo.

BILL: Android really is the heavy hitter in the Linux mobile space today. More and more phones are shipping with Android. I never thought I'd see the day when my sister used a Linux device daily—pretty cool stuff.

KYLE: Even though I'm not a huge fan of Android myself, I do have to agree with the readers here at least on the point that Android has gotten Linux into way more people's hands (and pockets) than just about anything else. That said, I still prefer the extra flexibility and power that Maemo (and hopefully MeeGo) brings to mobile devices.

READERS: best desktop environment: GNOME and KDE (tie).

BILL: GNOME wins here for me, hands down. Part of it is just sheer muscle memory, but the simplified elegant interface of GNOME appeals to me more than all the knobs and switches of KDE.

KYLE: I thought it was very interesting that this landed as a tie this year, and honestly I think I'm with the readers on this, except that to me, the actual desktop environment doesn't really matter very much anymore. I do use GNOME at the moment, but mostly because it was the default desktop environment on my

system. I generally disable or just plain don't use most of the GNOME features on my desktop. I use terminals and a Web browser for everything I need, so if Ubuntu decided tomorrow it would default to KDE, I suppose I'd be a KDE user then.

READERS: best Web browser: Firefox; honorable mention: Chrome.

BILL: I've got to side with the crowd here, Firefox still beats Chrome and every other browser for me. The plethora of plugins, ubiquity and the fact that I've been using it forever make it my go-to browser.

KYLE: It turns out I agree with Bill here. I still prefer Firefox myself. I know that vimperator-like plugins do exist for Chrome; however, I not only like the wide variety of plugins that Firefox has, but I also think it's important for a company like Google to have a viable open-source competitor, especially when you consider all the valuable marketing data that can be had in a user's browser history.

READERS: best e-mail client: Thunderbird; honorable mention: Gmail Web client.

BILL: Thunderbird wins—awesome. I don't even see mutt on the list.

KYLE: I won't rehash our past column on mutt vs. Thunderbird, but I have to admit I was disappointed not to see mutt make the list. I mean, I don't even count Gmail as a Linux e-mail client any more than I would count Yahoo or Hotmail. It should be no surprise to anyone who has read my column that I think mutt beats all of these clients hands- (and mice-) down.

READERS: best IM client: Pidgin; honorable mention: Skype.

BILL: I'm glad to see Pidgin beat Skype. I've got an allergy to Skype—Kyle's seen it: hives, itching, the whole bit. Seriously though, Pidgin's open-source, cross-platform, modular approach to IM makes it a

winner for most of us.

KYLE: Skype, the Sun—Bill has a lot of allergies it turns out. I understand why Pidgin got the win in this category, but I have to say I prefer BitlBee for all my IM needs. Really nothing compares to being able to treat IM like just another IRC channel inside my favorite IRC client (more on that next).

READERS: best IRC client: Pidgin; honorable mention: XChat.

BILL: Here's where I disagree with the crowd. Pidgin's IM-style interface really gives me fits when I try to use it as an IRC client. I'd much rather use XChat or Irssi, and I do, in fact, use both, although I spend more time on Irssi as I can leave it running in a screen session.

KYLE: Wow, Bill agrees with me on a console application over a GUI one! Like with mutt, once you get over the initial learning curve, Irssi really is the best IRC client. Like Bill, I leave Irssi running inside a screen session, so I can reconnect to it from wherever I happen to be. The fact that an IM client like Pidgin can connect to IRC is neat, I guess, but I think all the other straight IRC clients do it better.

READERS: best audio player: Amarok; honorable mention: RhythmBox.

BILL: Where's the love for XMMS? Seriously, the last time I actually listened to music on the computer instead of a mobile MP3 player, I was using XMMS. I guess I'm becoming a greybeard in some things.

KYLE: I'll be honest, I don't really listen to music from apps on my computer anymore. For years, I've used either a portable media player (which is now my N900) or possibly my home XBMC machine to play music. That being said, despite its complexity, I still prefer Amarok over RhythmBox. I keep a very tidy nested directory structure for my music, and I like it when it's simple to browse an actual directory structure instead of relying on fickle ID3 tags.

READERS: best media player: VLC; honorable mention: MPlayer.

BILL: I side with the MPlayer crowd here. I've traditionally just had better success with it, and it's kind of the Swiss Army knife of media.

KYLE: VLC always has seemed needlessly complex when you want to do more than play an .avi. It's nice to have around as a sanity check if a video doesn't play on something else, but to me, it's hard to beat the power and speed of MPlayer. It plays just about anything I've thrown at it and always seems to make the most of my resources.

READERS: best on-line collaboration tool: Google Docs; honorable mention: wikis.

BILL: I'm torn here. I like the cloud-based stuff, but I'd never dream of putting my business' documentation on Google Docs. I'd have to make this a split decision: Google Docs for personal documentation, and wikis for business documentation.

KYLE: Wow, this category made Bill question his long-held love affair with the cloud. I suppose I prefer wikis over Google Docs. I just don't use a lot of cloud offerings, Google or otherwise, so I'd prefer to use tools that I can conceivably run and control 100% myself.

READERS: best cloud-based file storage: Dropbox; honorable mention: Ubuntu One.

BILL: I'm a huge fan of Dropbox and use it daily. So for personal use, I'd definitely say it's the best solution. However, if I were running a business and needed cloud storage, I'd look at Amazon's S3.

KYLE: My favorite cloud-based file storage is my own file server. If you have Linux and a few disks, it's trivial to rsync files around to whatever machine you are in front of. Granted, some of these other offerings might have more automated features, but I'd like to see the Open Source community provide something that made it easy for people to have these same capabilities on their own machines without relying on some third-party outside their control.

READERS: best game: *Frozen Bubble*; honorable mention: *Doom* and *Battle for Wesnoth* (tie).

BILL: This is a tough one, as I don't play many games, but I'd have to side with *Doom*, simply because of nostalgia.

KYLE: This is a tough one for me too, because I'm torn between *Frozen Bubble* and *Quake 3*. *Doom* is fun, but these days, it's become the game version of Hello World that everyone ports to a new Linux device. I guess I'll go with *Quake 3*. Yes, it's an old FPS, but it's still my favorite.

READERS: best database: MySQL; honorable mention: PostgreSQL.

BILL: MySQL wins for me. It's just too easy to get a database up and running with it.

KYLE: Although MySQL is nice, I prefer PostgreSQL both for its power and the fact that it's still nice to have *some* sort of alternative to Oracle. These days, it really isn't any more difficult to manage MySQL than PostgreSQL.

READERS: best backup solution: rsync; honorable mention: tar.

BILL: I have to go with tar on this one, because when I hear "backup", I think "tape", and tar stands for Tape Archive. rsync's got its place, and I use it a lot to copy data from point A to B, but it feels like more of a copy utility rather than a backup solution.

Sure, some phones have more processing power these days, but it's hard to beat the openness and variety of development languages that the Maemo platform has on the N900.

KYLE: I'm going to have to go with BackupPC here. It's relatively easy to set up, easy to manage and works well. Plus, you have the option to use either rsync or tar for your backups. It seems like a win-win to me.

READERS: best virtualization solution: VirtualBox; honorable mention: VMware.

BILL: I did a virtualization shootout last year, and VMware came out on top at the time. Now, I'm not so sure. VirtualBox has come a long way. I'm split on this one.

KYLE: I think the main thing VirtualBox has going for it is the ease at which it installs within distributions like Ubuntu. I just hope Oracle doesn't kill it. On the enterprise end, I favor VMware's enterprise products first and probably KVM second.

READERS: best content management system: WordPress; honorable mention: Drupal.

BILL: WordPress: it's ubiquitous, easy to install and maintain, and it gets out of your way and lets you create content. What can be better than that?

KYLE: I'm with Bill on this one. Drupal is powerful but has a significant learning curve. It's dead simple to set up a site on WordPress, and plenty of support and packages are available to make it easy to maintain long-term.

READERS: best Linux laptop vendor: Dell; honorable mention: Lenovo.

BILL: I own quite a few laptops, both Dell and Lenovo, but when I spec a new machine, it's always a Lenovo system. The build quality still is a cut above Dell, despite the leaps and bounds that Dell has improved in the past few years. Besides, how many laptops have holes in the lower case to allow Mountain Dew to drain out?

KYLE: This is tough, because although Lenovo doesn't do the best job of championing Linux as a pre-installed option like some of the other vendors, I've found its Linux support (especially for the ThinkPad series) is very solid. I've said for a while now that if you want good Linux support, you should get the same hardware the developers use, and it seems there are plenty of Linux developers on ThinkPads.

READERS: best brand of video chipset: NVIDIA; honorable mention: ATI.

BILL: It's rare that I do any 3-D at all, so graphics simply need to be there, have stable drivers and not use a lot of power. That's why I like the Intel graphics chipsets.

KYLE: Although like Bill, I can handle my 3-D acceleration needs with my onboard Intel graphics card, the VDPAAU support in the NVIDIA cards that allows you to offload decoding of HD video content to your graphics card pushes NVIDIA to the top of my list.

READERS: best Linux smartphone: Nokia N900; honorable mention: HTC Nexus One.

BILL: I've spent time with the Motorola Droid, and it was a very pleasant experience. I don't run a Droid as my daily phone, but if I were stuck with one, I wouldn't mind terribly. It's a nice unit.

KYLE: The readers are right on with this one. The N900 is the phone I carry with me, and I'd argue is definitely the best Linux smartphone out there. Sure, some phones have more processing power these days, but it's hard to beat the openness and variety of development languages that the Maemo platform has on the N900.

READERS: best Linux-based gadget: Amazon Kindle; honorable mention: TomTom Navigation System.

BILL: The Kindle is the gadget to beat in this space. They're very nice, though the unmentioned Barnes & Noble's Nook is a good alternative (see my review of the Nook on page 48).

KYLE: This is tough for me, as I look at all the Linux-based gadgets around my house. I think the Spykee, the Pogoplug, the N900 and the Parrot AR drone are all very interesting Linux-based gadgets. But, I still think I'm going to have to go with my N900.

READERS: product of the year: Android; honorable mention: KDE.

BILL: I have to agree with the readers. Android deserves product of the year. Not only are countless people now running Linux, but also Android's technical achievements can make their way back into mainline Linux over time and improve the entire Linux ecosystem. I'm looking forward to seeing ARM-based, Android-powered Netbooks in the next year.

KYLE: As I mentioned previously, I'm not a big Android fan personally, but I do agree that it should be product of the year. It has dramatically increased the number of people using Linux, even if they don't know it, and it has great potential to increase Linux's reach in the future. ■

*Kyle Rankin is a Systems Architect in the San Francisco Bay Area and the author of a number of books, including *The Official Ubuntu Server Book*, *Knoppix Hacks* and *Ubuntu Hacks*. He is currently the president of the North Bay Linux Users' Group.*

Bill Childers is an IT Manager in Silicon Valley, where he lives with his wife and two children. He enjoys Linux far too much, and he probably should get more sun from time to time. In his spare time, he does work with the Gilroy Garlic Festival, but he does not smell like garlic.

Polywell Industrial Mini-PCs

Cost Effective Embedded PC For Appliances



NVIDIA ION™



ITX-10A



ITX-30A with PCI Riser



ITX-30G with NVIDIA® ION™ Graphics
Barebone system **\$199**



ITX-50 Series



ITX-40A with Slim Optical Bay



VESA / Wallmount option



Full Height Riser or Low-Profile Add-on Slot
up to 8 x 2.5" or 4 x 3.5" Hard Drives



ITX-1000C with 4LAN
and WiFi Option

Over 250 Mini-ITX Models Available:

- NVIDIA® GeForce 8200/8100 with AMD® Athlon/Phenom Processor
- NVIDIA® GeForce 9300/9100M/7050 with Intel® Core 2 Duo Processor
- PCI, PCIe, MiniPCIe Slot for TV Tuner or Industrial Add-on
- Custom Design Chassis for Small to Mid Size OEM Project

888.765.9686

linuxsales@polywell.com

polywell.com/us/Lx

■ 23 Years of Customer Satisfaction

■ 5-Year Warranty, Industry's Longest

■ First Class Customer Service

Polywell Computers, Inc

1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

NVIDIA, ION, nForce, GeForce and combinations thereof are trademarks of NVIDIA Corporation. Other names are for informational purposes only and may be trademarks of their respective owners.



Polywell Storage Servers

More Choices, Excellent Service, Great Prices!

Netdisk 9015N



Quiet Storage NAS/SAN/iSCSI

10TB \$1,799

20TB \$2,999

30TB \$4,999

- Dual Gigabit LAN
- RAID-5, 6, 0, 1, 10
- Hot Swap, Hot Spare
- Linux, Windows, Mac
- E-mail Notification
- Tower or Rackmount

Silent Eco Green PC

The Best Terminal PC

Intel® Low Voltage Processor
Energy efficient, Quiet Platform.
starts at \$199



Mini-ITX LD-001

High Performance 5048A



5U-48Bay 96TB Storage Server

4U45JB / 4U36A / 4U24A
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 Giga / 10Gbit LAN



4U-24Bay 48TB \$8,999
4U-36Bay 72TB \$12,500
4U-45Bay 90TB JBOD

2008A / 2012A / 2012B
RAID-6, NAS/iSCSI/SAN Storage
Mix SAS / SATA, 4 GigaLAN



2008A 16TB \$2,699
2012A 24TB \$3,999
2012B 24TB \$4,899

1U945GCL2



Mini-1U Server \$499

Intel® Dual-Core Processor, 2 x 500G RAID
Dual GigaLAN, 4GB DDR2 RAM

- Over 20 Years of Customer Satisfaction
- 5-Year Warranty, Industry's Longest
- First Class Customer Service

888.765.9686

linuxsales@polywell.com

www.polywell.com/us/Storage

Polywell Computers, Inc

1461 San Mateo Ave. South San Francisco, CA 94080 650.583.7222 Fax: 650.583.1974

Intel is trademarks of Intel Corporation. Other names are for informational purposes only and may be trademarks of their respective owners.



Remembering the Future

Keeping Linux promising since 1994. DOC SEARLS



In 1993, Phil Hughes pulled together a short e-mail list to discuss the idea of starting a free software magazine. As soon as Phil got the rest of us (especially me) up to speed on the subject, he changed it. Instead of writing about free software, he now wanted to write about a new UNIX-y operating system by some kid from Finland. The general reaction on the list was, "What?" But Phil had his mind made up, and when Phil does that, there isn't any undoing it.

What Phil saw with Linux was more than just a cool new operating system. He saw a path to freedom for UNIX, and the eventual use of free and open software and hardware by pretty much everybody. He was right, and that's one reason we are now at issue 200 of *Linux Journal*.

Looking forward is a stock in trade of technology magazines, but at *Linux Journal*, we've gone to extremes from the start. (Hard not to, given the inborn promise of Linux and the constantly growing spread of its uses.) For example, in issue #2 (April–May 1994, www.linuxjournal.com/issue/2), Phil's From the Editor was a column he might post in January 2000. Here is some of what he saw—and how close he was to being right:

1) "Linux and an Internet connection in over 100 million homes worldwide."

Linux already was on its way to becoming the embedded OS of choice for countless Net-connected devices, and that last number was actually low.

2) "90% of our subscribers are now via the Internet rather than on paper."

Readers, for sure. Subscribers, not yet.

3) "Seamless ISDN connections" between the magazine's own far-flung internal systems.

Wrong on ISDN, but right on the connections, which matter far more.

4) "Formation of MoAml Semiconductor from Motorola, AMD and Intel engineers in 1994"—on whose 32- and 64-bit systems Linux would run.

Right in the sense that one standard would emerge and Linux would run on it.

5) "Linux became the most popular operating system used in computer science classes in 1995. This meant that the pool of available talent in the Linux market was huge."

Close enough with the former, and more than true with the latter.

6) Novell's purchase of USL in 1993 made Windows NT "the niche operating system".

Well, one could make the case for the long-term niche-ing of Windows OSes.

7) "Likewise, the decision of many fence-sitting vendors to go with Linux gave it the needed push that caused it to become a mainstream system."

For sure, most notably with IBM.

8) For larger vendors, "Going to Linux as their operating system both reduced their software development costs and made it easier for them to find pre-trained systems programmers for the software work they still needed to perform."

Bull's-eye.

9) Telephone directories are "free and much easier to use than a traditional phone book", which "caused more people to elect for an Internet connection..."

We call this Google.

10) "Old Linux activists and developers" are "still writing code or books", but "we don't see any who are CEOs of multi-billion dollar corporations".

Ask Bob Young. Before he founded Red Hat, Bob was the first editor of *Linux Journal*.

11) "Linux machines make up the majority of the machines connected to the Internet."

This is still off in the future, although Android looks promising for smartphones and handhelds. More important is what's happening to the Internet itself, as it divides into a vast collection of apps and services, many of them silo'd to privately managed platforms (for example, Apple's i-Everything and Amazon's Kindle). Even Android's success is compromised by all-proprietary 3G, 4G and LTE wireless data networks that continue leveraging ancient phone-company billing models, which include punishing costs for "roaming"

across national boundaries. There also remains an absence of Internet access in much of the world. All these challenges keep relevant Phil's closing words in 1994:

We need to offer Internet connectivity to everyone....In 1990 people were much less likely to know the names of their neighbors or world leaders than the names of fictitious characters on TV shows. Although Internet connectivity may not help people get to know their physical neighbors, it does help them build a community of electronic neighbors. Using the Internet is active, not passive. Whether people elect to do research or electronically talk to another person, they are now making real choices and possibly talking to real people.

Because Linux was so significant in getting tens of millions of people connected to the Internet in the past five years, and Linux machines make up the majority of the machines connected to the Internet today, I see this as a project that the Linux community should take on. In 1993 and 1994 we were all out there telling people about Linux. If today we all walked next door, introduced ourselves to those neighbors that have lived there since 1990 and then offered to help them get connected to the Internet, we could claim another huge victory for the Linux progressive movement before year's end.

This is still our job. Only now our neighbors are everywhere. ■

Doc Searls is Senior Editor of *Linux Journal*. He is also a fellow with the Berkman Center for Internet and Society at Harvard University and the Center for Information Technology and Society at UC Santa Barbara.

iX-Triton TwinBlade Servers: The Easy-to-Manage, Greener Way to Serve

► AFFORDABLE ► ECONOMICAL ► SAVINGS

The new **Triton TwinBlade Server** is the most technologically advanced blade server system in the industry, and the ideal solution for power-efficiency, density, and ease of management.



Houses **20** nodes
in 7U of rack space

The **Triton TwinBlade Server** supports up to 120 DP servers with 240 Intel® Xeon® 5600/5500 series processors per 42U rack, achieving an unmatched 0.35U per DP node. Up to two 4x QDR (40 Gbps) Infiniband switches, 10GbE switches or pass-through modules give the TwinBlade the bandwidth to support the most demanding applications.

With **N+1 redundant, high efficiency (94%) 2500W power supplies**, the TwinBlade is the Greenest, most energy-efficient blade server in the industry. The energy saved by the iX-Triton TwinBlade Server will keep the environment cleaner and greener, while leaving the green in your bank account.

Server management is also simple with the Triton Twin Blade Server.

Remote access is available through SOL (Serial Over Lan), KVM, and KVM over IP technologies. A separate controller processor allows all of the Triton's remote management and monitoring to function regardless of system failures, offering true Lights Out Management.

Using the **Triton's management system, administrators can remotely control TwinBlades**, power supplies, cooling fans, and networking switches. Users may control the power remotely to reboot and reset the Triton TwinBlade Center and individual Twin Blades, and may also monitor temperatures, power status, fan speeds, and voltage.

For more information on the **iX-Triton TwinBlade**, or to request a quote, visit:
<http://www.iXsystems.com/tritontwinblade>

Key features:

- Up to 10 dual-node TwinBlades in a 7U Chassis, 6 Chassis per 42U rack
- Remotely manage and monitor TwinBlades, power supplies, cooling fans, and networking switches
- Virtual Media Over Lan (Virtual USB, Floppy/CD, and Drive Redirection)
- Integrated IPMI 2.0 w/ remote KVM over LAN/IP
- Supports one hot-plug management module providing remote KVM and IPMI 2.0 functionalities
- Up to four N+1 redundant, hot-swap 2500W power supplies
- Up to 16 cooling fans

Each of the TwinBlade's two nodes features:

- Intel® Xeon® processor 5600/5500 series, with QPI up to 6.4 GT/s
- Intel® 5500 Chipset
- Up to 128GB DDR3 1333/ 1066/ 800MHz ECC Registered DIMM / 32GB Unbuffered DIMM
- Intel® 82576 Dual-Port Gigabit Ethernet
- 2 x 2.5" Hot-Plug SATA Drive Trays
- Integrated Matrox G200eW Graphics
- Mellanox ConnectX QDR InfiniBand 40Gbps or 10GbE support (Optional)



Call iXsystems toll free or visit our website today!
+1-800-820-BSDi | www.iXsystems.com



Cut Execution Time by >50% with WhisperStation-GPU

Delivered ready to run new GPU-enabled applications:

Design

3ds Max
Bunkspeed
Shot
Adobe CS5

Simulation

ANSYS Mechanical
Autodesk Moldflow
Mathematica

MATLAB
ACUSIM AcuSolve
Tech-X GPULib

BioTech

AMBER
GROMACS
NAMD, VMD
TeraChem

Integrating the latest CPUs with NVIDIA Tesla Fermi GPUs, Microway's WhisperStation-GPU delivers 2x-100x the performance of standard workstations. Providing explosive performance, yet quiet, it's custom designed for the power hungry applications you use. Take advantage of existing GPU applications or enable high performance with CUDA C/C++, PGI CUDA FORTRAN, or OpenCL compute kernels.

- ▶ Up to Four Tesla Fermi GPUs, each with: 448 cores, 6 GB GDDR5, 1 TFLOP single and 515 GFLOP double precision performance
- ▶ Up to 24 cores with the newest Intel and AMD Processors, 128 GB memory, 80 PLUS® certified power supply, and eight hard drives
- ▶ Nvidia Quadro for state of the art professional graphics and visualization
- ▶ Ultra-quiet fans, strategically placed baffles, and internal sound-proofing
- ▶ New: Microway CL-IDE™ for OpenCL programming on CPUs and GPUs



WhisperStation with 4 Tesla Fermi GPUs

Microway's Latest Servers for Dense Clustering

- ▶ 4P, 1U nodes with 48 CPU cores, 512 GB and QDR InfiniBand
- ▶ 2P, 1U nodes with 24 CPU cores, 2 Tesla GPUs and QDR InfiniBand
- ▶ 2U Twin² with 4 Hot-Swap MBs, each with 2 Processors + 256 GB
- ▶ 1U S2050 servers with 4 Tesla Fermi GPUs

Microway Puts YOU on the Cutting Edge

Design your next custom configuration with techs who speak HPC. Rely on our integration expertise for complete and thorough testing of your workstations, turnkey clusters and servers. Whether you need Linux or Windows, CUDA or OpenCL, we've been resolving the complicated issues – so you don't have to – since 1982.

Configure your next WhisperStation or Cluster today!

microway.com/quickquote or call 508-746-7341

Sign up for technical newsletters and special GPU promotions at microway.com/newsletter



OcioPuter™ 4U Server with up to 8 GPUs and 144 GB memory

1U Node with 2 Tesla Fermi GPUs

2U Twin² Node with 4 Hot-Swap Motherboards
Each with 2 CPUs and 256 GB



GSA Schedule
Contract Number:
GS-35F-0431N

Microway
Technology you can count on™