

# LINUX

www.linuxformat.co.uk

# FORMAT



**WINNERS  
REVEALED!**

**SCO SUES IBM OVER LINUX CODE...**

# \$1,000,000,000 LAWSUIT!

...but does it add up? Details, comment  
and analysis inside **PAGE 12**



**NEW SERIES**

## BLENDER MADE EASY!

Create 3D modelling  
masterpieces in minutes **p84**

## GIGABYTE SR113

Great value 1U Pentium 4  
server on test **p30**



## LINUX ON THE MAC

Linux on Apple hardware from  
a Mac user's perspective **p50**

## NO MORE BIOS! p58

DVD issue also available

**LXF40 MAY 2003**

Printed in the UK

**£5.99**

**future**

magazines



(\$5.99 outside UK and Republic of Ireland)

**THE UK'S BEST-SELLING LINUX MAGAZINE**

# How much?!?

**Y**ou read it correctly, one billion US dollars. That's the amount SCO are hoping to get out of IBM for unlawful use of trade secrets. The full facts of the story, plus some useful notes on the actual lawsuit and plenty of comment can be found in our extended news coverage this issue, starting on page 12. Obviously we are not the judge and jury in a court of law, but certain elements have to be cleared up to stop this story being used as a weapon of FUD against Linux in general. So let's just make this extra clear, the lawsuit is currently only against IBM, and SCO have claimed that they wouldn't take action against Linux vendors.

Of course, on the sidelines of this story is the fact that SCO are one of the major player in the UnitedLinux partnership. They have been quick to release statements that their Linux distributions are in no way affected, and they will continue to develop for Linux. Unfortunately, it seems to have escaped their notice that you can't call Linux developers (and I'm paraphrasing here) unco-

ordinated morons and still expect them to foster warm feelings towards you. Described by many in the Linux community as a 'suicide note', the preamble text of the lawsuit poors scorn on the capabilities of Linux and the efforts of its developers, and certainly removes SCO even further from the community. One can only conclude that they don't 'get' the whole Linux thing, and like those before them who tried to make money from Linux but failed to grasp the ideas of Open Source and Free Software, their days as a serious player in the Linux market could possibly be numbered.

But on to happier things. There are a fair few reviews crammed into our pages this issue, including the latest SuSE Enterprise Server (built on UnitedLinux), Zend Studio and a terrific value entry-level server.

You'll no doubt be anxious to find out the winners of the *LXF Awards* (p54) and check out our brand new *Blender* tutorial starting this issue! There's plenty more inside – get reading!



**Nick Veitch** EDITOR

One billion dollars? Find out what it's all about, and what may happen (in our opinion) **p12**

The great and good are honoured in our awards feature **p54**

Now open-sourced, *Blender* is an excellent way to get into 3D **p84**



## AIMS OF THE MAGAZINE

**Linux Format** is a magazine dedicated to Linux and the Open Source community. We aim:

- To provide the most accurate, unbiased and up to date information on all things Linux.
- To promote the use of Linux in business and the home, for servers and on the desktop.
- To support the Open Source community by providing a resource of information, and a forum for debate.
- To help all readers get more from their Linux experience by providing insightful and useful tutorials.

## MEET LINUX FORMAT'S TEAM OF WRITERS...



**Andrew Channelle**  
The *LXF* resident newshound has also penned our series on mangling (surely 'improving') Mozilla.



**David Coulson**  
Our Answers guy is a networking and security guru with plenty of sysadmin experience.



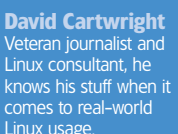
**Paul Hudson**  
Formerly our PHP guru and a web designer, Paul has joined us as our Reviews Editor. Be gentle with him...



**Jono Bacon**  
Jono is a core KDE developer, web developer and writer. Jono is also a musician and sound engineer.



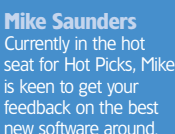
**Maurice Kelly**  
Our Norn Iron representative has been busy this month coding network applications.



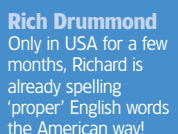
**David Cartwright**  
Veteran journalist and Linux consultant, he knows his stuff when it comes to real-world Linux usage.



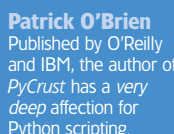
**Hoyt Duff**  
Now an acclaimed author, Hoyt is our resident expert on all things Red Hat. And fishing.



**Mike Saunders**  
Currently in the hot seat for Hot Picks, Mike is keen to get your feedback on the best new software around.



**Rich Drummond**  
Only in USA for a few months, Richard is already spelling 'proper' English words the American way!



**Patrick O'Brien**  
Published by O'Reilly and IBM, the author of *PyCrust* has a very deep affection for Python scripting.

## CONTACT US

Letters for publication:  
[linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

Subscriptions/back issues:  
[subs@futurenet.co.uk](mailto:subs@futurenet.co.uk)

Technical help/Ask the Experts:  
[linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

Disc problems:  
[linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

General enquiries:  
[linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

Website: [www.linuxformat.co.uk](http://www.linuxformat.co.uk)

Or send your letters to:  
**LINUX Format, Future Publishing,**  
30 Monmouth Street, Bath, BA1 2BW  
Phone: 01225 442244  
Fax: 01225 732295

More contact info on p114

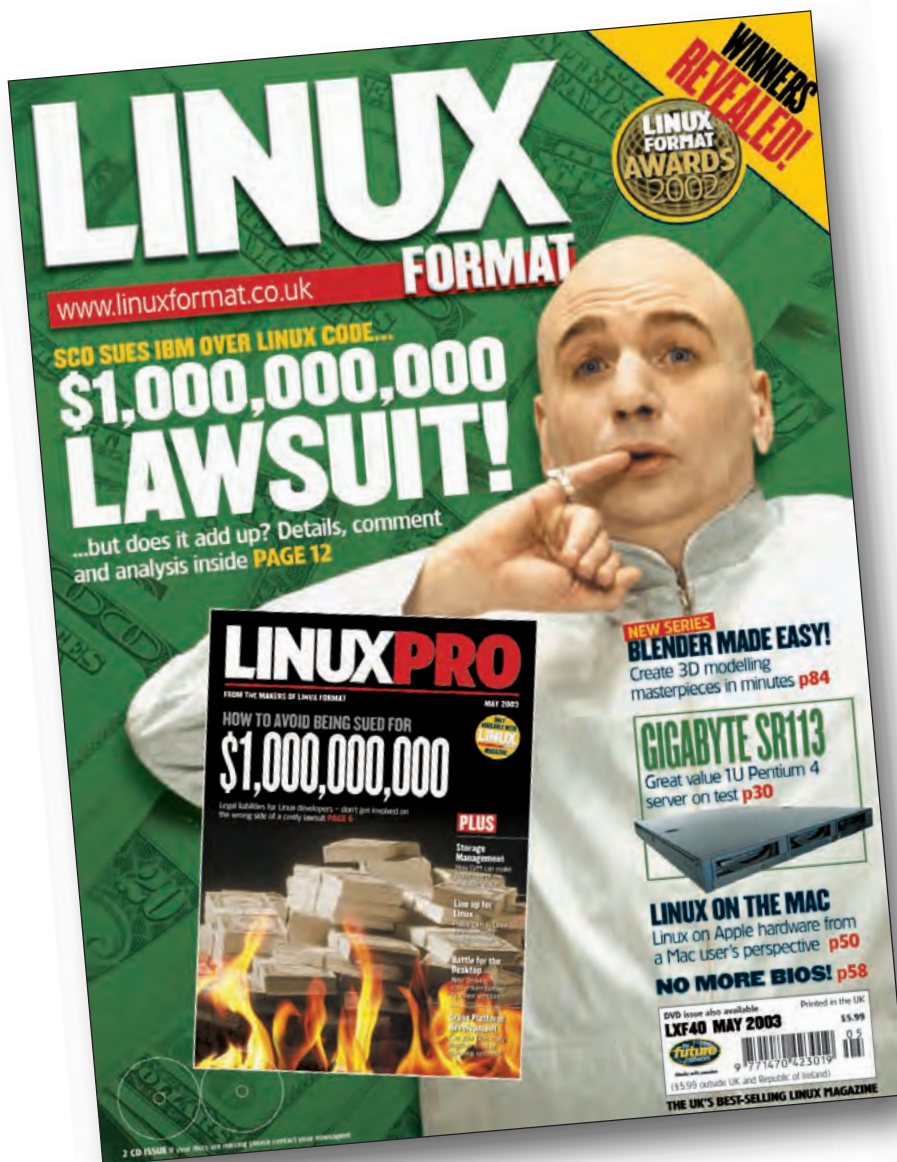


LXF40 May 2003

# Contents



Welcome to another jam-packed issue of *Linux Format*, your guide to all things Linux!



## COVER HIGHLIGHTS

# \$1,000,000,000 lawsuit!

12

SCO sues IBM over alleged misuse of Unix code: analysis and comment

## 84 Blender made easy

New 3D tutorial series starts this issue!

## 32 Gigabyte SR113 server

Bargain price, and a mean performer

## 50 Linux on the Mac

The whys and hows of Linux for Mac users

## 58 No more BIOS!

How EFI could change the face of computing forever!



'Since I'm writing about my love for **Python**, the letters that **make up** this article must be **love letters**'

**80** Passionate about Python

'Even today, operating systems still need to make **real mode calls** to the BIOS, which are **incredibly slow**'

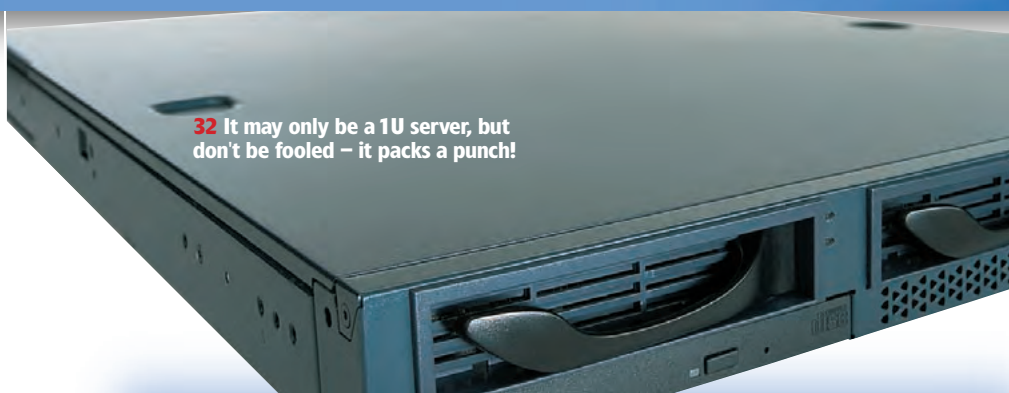
**58** The rationale behind EFI

## LINUXPRO

**Legal Liability** Don't get sued for \$1 billion!  
**LVM** solve storage woes  
**Desktop Linux Consortium** what and who  
**Cross-platform GUIs** Qt v GTK **VIPER**  
Linux helps the police with their inquiries



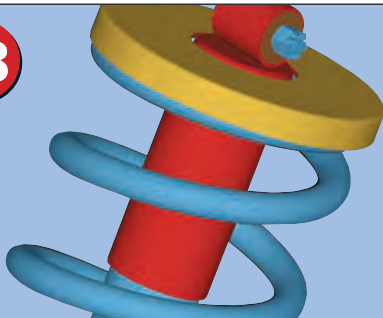




**32** It may only be a 1U server, but don't be fooled – it packs a punch!

- |   |  |
|---|--|
| <p><b>6 News</b><br/>Analysis of the latest Linux stories</p> <p><b>17 Infosecurity Preview</b><br/>What's going on at Infosec 2003</p> <p><b>20 Mailserver</b><br/>Share your views in print</p> <p><b>28 VariCAD review</b><br/>Mechanical CAD at its best</p> <p><b>32 Gigabyte SR113 review</b><br/>Bargain-price 1U server on test</p> <p><b>36 IP*Works! review</b><br/>Proprietary network libraries</p> <p><b>40 IEMS review</b><br/>Easy to configure mailserver</p> <p><b>42 Book reviews</b><br/>Three new tomes digested</p> <p><b>45 Hot Picks</b><br/>The best new free software</p> <p><b>50 Linux on Mac</b><br/>Installing Linux for Mac users</p> | <p><b>54 LXF Award winners</b><br/>Your votes have (been) counted</p> <p><b>58 What on Earth is EFI?</b><br/>BIOS replacements explained</p> <p><b>64 Mozilla tutorial</b><br/>Packaging up your work of art</p> <p><b>68 GIMP tutorial</b><br/>Non-destructive image editing</p> <p><b>72 PHP tutorial</b><br/>Creating GUIs with GTK</p> <p><b>80 Python tutorial</b><br/>Lists, strings and tuples explained</p> <p><b>84 Blender tutorial</b><br/>Starting out with Blender 3D</p> <p><b>88 Answers</b><br/>Your Linux problems solved!</p> <p><b>98 Coverdisc pages</b><br/>Get the most from your discs</p> <p><b>110 User Groups</b><br/>Local support and fun!</p> |
|---|--|

**28**



**54**



**58**



**84**



## COVERDISCS

A DVD or 2 CDs packed full of the latest Linux goodies **98**



### CDS A AND B

**Runtime Revolution 1.1.1** full commercial software – create Linux apps the easy way; **Vector Linux** complete, streamlined slackware-based distro; **Mini distros** three more specialist distributions; **Mozilla 1.3** final release version, includes many new features and improvements; **MySQL 4** production, mission critical ready version; PLUS **EcliptRoaster**, **IceWM**, **Kino** and plenty more!



### DVD

**ALL OF THE ABOVE PLUS:** **MisterHouse** Perl-based home automation; **XNAME** arcade emulator; **MondoRescue** recovery software; **Lopster** P2P networking and more!

Please read the coverdisc instructions on page 100 and page 107 before installing from coverdiscs!



**SAVE MONEY!**

**SUBSCRIBE TO LINUX FORMAT**

See page 96 or phone 0870 4448645





# Newsdesk

MS marketshare 'unrepresentative'; HOAP2; vertical OpenOffice.org; Textmaker; Linux viruses; HP in bed with RH; Sun refocuses Linux efforts; MySQL 4; Intel/Microsoft 'protectionism'?

## DISTRO WARS EPISODE II...

# The great leap forward?

**N**ot content to sit back and reflect on their successes, the major distribution vendors have packaged up the latest desktop and server advances into their next generation of releases. And in the case of Red Hat – which celebrated its 10th birthday in March – the company regard the release as significant enough to bypass the usual point numbers and jump straight to version nine. In a break from the usual distribution process, the company offered Red Hat Network subscribers one week's exclusive access to download sites, before the six ISOs (three each for binary and source) became generally available on April 7th 2003.

Based around recent KDE and GNOME desktops, with the latter as the default, Red Hat 9 keeps its controversial Bluecurve theme which attempts to 'unify' the two

environments and, again, ships with no MP3 software due to Red Hat's reading of the Fraunhofer licence restrictions on the format. Other notable additions include better support for wireless networking – although one early review reported problems with Wireless Equivalent Privacy (WEP) when moving from one access point to another – and CD burning. When it comes to the latter, inserting a blank CD into the drive opens a new window called '///burn' for dragging, dropping and burning files. Bluetooth is also included as standard via the *Bluez* library.

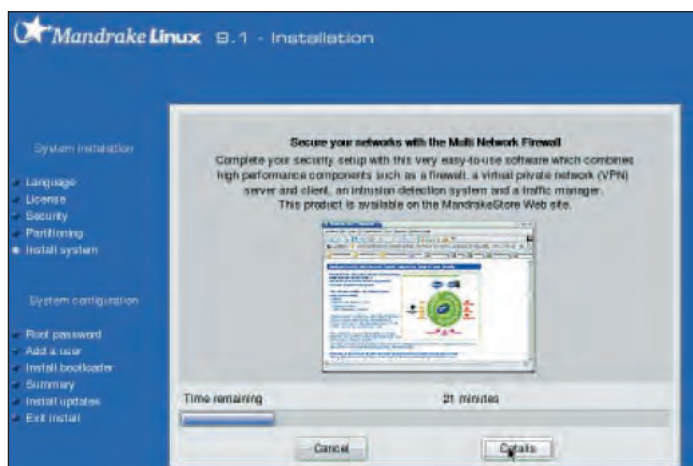
Mandrake's big selling point for 9.1 (apart from the updated desktops) is a brand new installer which reduces the number of inputs needed by the user to a bare minimum and has full support for resizing Windows XP's native NTFS file system. Though Mandrake picks KDE over GNOME as the default desktop, the developers

have taken a leaf out of Red Hat's book by attempting, as far as is possible, to make the user experience similar across the environments with the new Crystal-esque Galaxy theme.

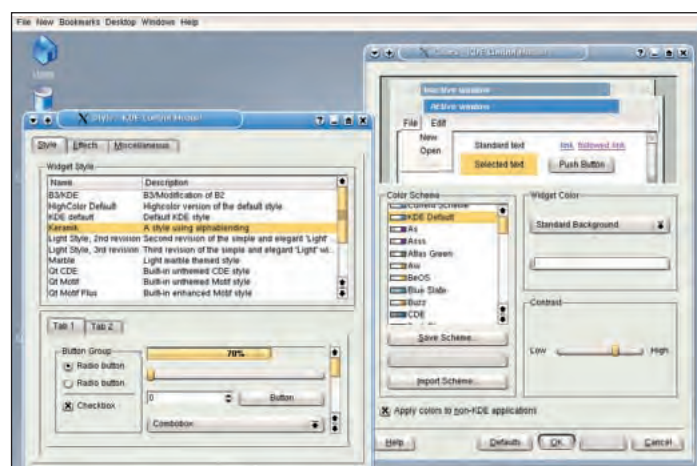
Other notable inclusions are 'Zero configuration' networking, support for WiFi, dynamic access to USB devices such as digital cameras and ACPI power management support. With an eye on their recent cash-flow problems, Mandrake 9.1 is the company's most important release. It is also the first where beta testers have helped determine the content of the



SuSE 8.2 is the company's first distribution to target home users.



Mandrake 9.1 will be vital in boosting the company's financial fortunes.



KDE 3.1 forms a core part of all three of the major distributions.

## YellowDog 3.0

Linux for Apple Macs

Though x86 users get a choice of new releases this month, Apple users are not left out with the introduction of Yellow Dog Linux 3.0. Based around Red Hat and featuring the same Anaconda installation routine, YDL3.0 now includes hardware support for the eMac, G-Force 4MX-based PowerMacs and the recently launched 12-inch Powerbook. As is becoming standard now, the new distribution also includes

its own unified (and Bluecurve-alike) desktop, called *Wonderland*.

In among the 1,300 applications and packages is the latest version (0.9.68) of Mac-On-Linux which is capable of running MacOS-X 10.2.

In addition to the PowerPC distribution, YDL's developer Terrasoft also supplies pre-loaded systems and clusters built around Apple's Xserve hardware.



YellowDog Linux joins the unified theme brigade with *Wonderland*.

distribution. President Jacques Le Marois said the new packages consolidated Mandrake's reputation as one of the most innovative companies in the Linux sector. "It is important to note that most of these features have been requested directly by our users, especially in the enterprise area," he said. "This new release truly has the potential to greatly extend Linux adoption throughout the world."

As ever, Mandrake is available in Standard, PowerPack and ProSuite boxed packages as well as the basic download edition and once more Mandrake is encouraging users to support the company through membership of the Mandrake Club.

### SuSE

Not to be outdone, SuSE's latest release – which for the first time has an edition aimed explicitly at home users – has been certified LSB-compliant. Based around the twin marvels of KDE 3.1 and GNOME 2.2 and the usual sink-full of applications, SuSE 8.2 is also the first distribution to feature a full version of *MainActor*, the professional video editing and

compositing application. *YaST 2* has been tweaked once more and now provides access to Wireless networking tools, including a profile manager intended to make moving between WLANs (and getting access to networked printers and scanners) a single click process.

As well as aiming for the desktop market, SuSE has announced a version of the software specially tailored for Intel's Itanium 2 processor as well as the first big order for the system from America's National Science Foundation. The NSF project will be built in tandem with IBM's Global Services division and consists of clusters at four separate sites with a combined storage capacity of 600 terabytes.

Intel's Lisa Graff said the release of SuSE 8 for Itanium 2 chips would drive the development of 64bit Linux applications. "The selection of SuSE for the massive Teragrid system demonstrates the stability, scalability and performance of SuSE Linux Enterprise Server and Itanium 2-based systems for complex implementations."

## NEWSBYTES



■ *Half-Life* developer **Valve** has announced a 64-bit Counterstrike server which brings a 30 per cent performance gain over its 32-bit counterpart. The server software runs on Linux and AMDs Opteron processor. Obviously pleased with his work, Valve MD Gabe Newell said "Every PC developer should be looking to get their server code and development tools running in 64-bits right away."

■ Web servers continue to get smaller. **Lantronix** have managed to squeeze a fair bit of network connectivity into a device not much bigger than the size of a standard RJ45 connector.

■ **Apple** are hoping to grab a slice of the clustering market, currently dominated by Dell Computer, by extending their Xserve lineup. The first product off the production line costs \$3,000 and sports a pair of 1.33GHz PowerPC chips, but omits a CD-ROM and video card and only has a single Ethernet port. The case also lacks space for additional hard disks. Apple have also killed the original iMac.

■ **Ericsson** and **Intel** are attempting to improve the productivity of mobile workers by creating a pub-based wireless network called, with not a little irony, *The Cloud*. Pubs, the venture's backers say, make more sense for WiFi access points, as the current trend for Coffee Shop access will run out of suitable venues soon. A spokesman said: "Not all cities have a branded coffee shop. In the UK, pubs are very good places to do business." The network should eventually cover over 30,000 locations, each one supporting 32 concurrent users.

■ **American Megatrends** is targeting small and medium sized enterprises with a new Linux/ATA based NAS storage system which, it is claimed, boosts performance by 30-40 per cent while significantly lowering the TCO. Packages, starting at \$2,300, can handle 1TB of storage, 3GB of DDR memory and up to four ATA 100/133 drives.

■ A vulnerability in **Samba's** main *smbd* code could allow an attacker to anonymously acquire super-user privileges on a *Samba* server. The problem affects all releases between 2.0.x and 2.2.7a and users are advised to update to 2.2.8 immediately or block access to TCP ports 139 and 445.

### Hoyt Duff

The author is one of 800 Hoyts living in the USA and runs a little fishing pier when he's not dabbling with his computers.



### COMMENT

## Be a part...

“ Even with recent improvements, the state of Linux documentation remains abysmal. Many documents found on the Internet are woefully out-of-date and poorly written. Full of dense jargon, they are often distro-specific or relate to a particular version of software; they are not easily generalised. One must master higher-level Googling skills to wade past useless info.

The much vaunted source code is, as a practical matter, almost as useless unless you've experience with coding and understand how Linux apps actually work. Besides, the source code is typically not installed by the average user, so they don't even know it can be used as a resource. The kernel docs are in a likewise fetid state.

The Linux Document Project is to be commended in reviving itself and supporting its authors in revising and updating the HOWTOs. Yet these docs too could use some serious grammar, syntax, and style.

A poorly documented prog is a poor program, especially in an OS that relies on others to expand and improve the code. Coders need to document their own code in more detail so that if they can't write the docs themselves, others who have the writing skills can do a better job. Conversely, those who have editing skills need to contribute their efforts to make the existing docs clearer and, hence, more useful to everybody.

If you have never written documentation, you can still help. Write a tutorial about your experience or examine an existing tutorial. Edit it to eliminate jargon, try and make it applicable to more than one distribution (so as to generalise it), correct grammar, syntax and spelling. Make it useful. Learn about Linux. Become a part of the solution. ”



## XFREE86 ACCOUNTABILITY?

# X, Lies and Videocards

**A** vociferous and potentially damaging spat has erupted on the future of the XFree86 project after core member Keith Packard was accused of 'actively (but privately) seeking out support for a fork of XFree86 that would be led by himself. He is also said to have held private discussions with 'vested interests' to highlight his concerns about the future of X. As a result Packard (largely responsible for the Xft font architecture) was booted out of the core development team.

He responded with a point-by-point explanation of his concerns, which mainly centred around a lack of overall governance and accountability from the XFree86 management team, claiming that community-led projects like KDE and GNOME provided a better model for developmental democracy.

"While the XFree86 Board of Directors is nominally in charge of

XFree86, they have absented themselves from governing the project," he wrote. "Decisions appear to be arbitrary and are not seen to reflect the will of the community. The leadership has no accountability to the community: thus community members have no ability to change project direction and the Board has little incentive to do so."

The Board of Directors, writing to the XFree developers mailing list, stated that discussing the future of the project within an invitation-only group ran "contrary to the spirit or openness and community involvement that many of those vested interests claim to espouse." As a response, the BOD also initiated a consultation forum inviting developers and users to express their thoughts, opinions and concerns.

David Wexelblat, one of the trio of developers that began the project, said he thought Packard's outburst was one

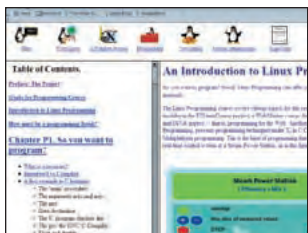


**Keith Packard's actions have caused much soul-searching about the future of the XFree86 project.**

of the most "unprofessional and tactless" things he'd experienced. 'While still a member of the XFree86 Core Team, he has explicitly attempted to subvert XFree86 by soliciting individuals and corporations to create an alternative to XFree86. Including inviting certain Core Team members to join him'.

In his long take on the issue (archived at <http://xfree86.org/pipermail/forum/2003-March/000128.html>) Wexelblat also accused Packard of lying about his intentions and said if he had genuine concerns he should have resigned from the core development team.

## Linux Web Watch/



**The place to start if you don't know your ALSA from your elbow.**



**Perl becomes essential once you get involved in Web development.**



**The future is XML (probably). Start boning up on it now.**



**Come to our Java tutorials late? Sort out the basics here!**

## Make your own Linux apps

While having a decent book to hand is desirable, there's no doubt that the best resource for programmers of every hue is the Internet. Constantly updated, tended to by devotees, searchable, (often) free. What's not to love?

Programming is not something you can do lightly as even the simplest, most humanly readable languages are never as straightforward as you'd imagine from listening to experts. So where do you begin? Well, you could do worse than visit [www.ftlinuxcourse.com](http://www.ftlinuxcourse.com) which was launched in 1998 with the intention of housing courses on every aspect of running a Linux systems. The base course is useful (though it should

be updated soon) covering many aspects of Linux use, but the programming courses are where you should head. These cover everything from web development to X programming to more complex languages. It's a very good start.

If you are attracted by talk of 'elegant one-liners' you need to investigate Perl, the 'Swiss army chainsaw' of programming languages.

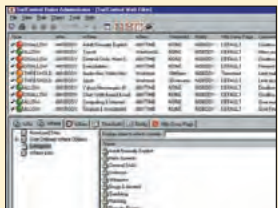
One of the better sites we've seen is [www.perl.com/pub/a/2000/10/begperl1.html](http://www.perl.com/pub/a/2000/10/begperl1.html) which is a six part tutorial aimed at absolute beginners.

XML is widely regarded as the future of, well, everything by some people so if you're chasing the money, this may be the way to go. The tutorials at [www.w3schools.com/xml/default.asp](http://www.w3schools.com/xml/default.asp) cover

everything from the differences between HTML and XML to the creation of applications.

If Java development is more your cup of tea, Sun have a decent range of tutorials at <http://java.sun.com/docs/books/tutorial/>. Some of these have appeared in various books, but there are also a number of online only efforts covering things such as drag and drop and networking.

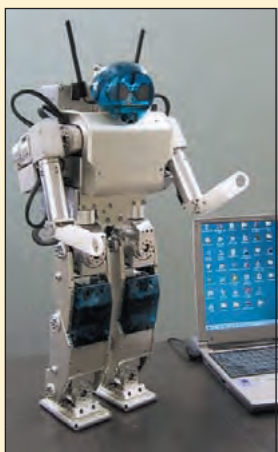
## NEWSBYTES



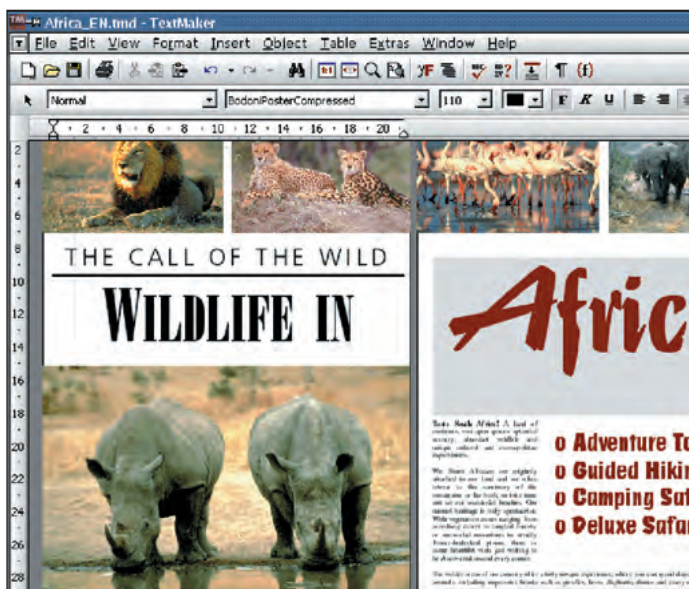
■ Sysadmins now have a new weapon in their bid to prevent itinerant workers downloading viruses, spyware and images of questionable taste in the form of **SurfControl's** upgraded filtering software. The system, running on Linux, can filter on groups or individual IP address and also has control for web annoyances such as popups, banner ads and media streams. A downloadable demo is available on the Internet from [www.surfcontrol.com](http://www.surfcontrol.com)

■ **VMware** has released the latest version of its virtualisation software which includes support for a greater number of guest systems as well as improved sound and graphics card drivers. One of the more interesting innovations, though, is the ability to drag and drop files from one virtual machine to another or set up shared folder without needing access to an external network. Recent purchasers of **VMware 3.2** may be entitled to a free upgrade.

■ The **Mozilla** project has spawned a *Phoenix*-like mail client called *Minotaur* developed for users unsatisfied with the do-it-all attitude of the main browser suite. The project is currently searching for members and a default theme.



■ **Fujitsu** has given birth to a successor to the Humanoid Open Architecture Platform (HOAP) 1 biped robot. HOAP-2 is built around a 700MHz Pentium III CPU with USB internal networking, optional WiFi link and a version of Linux optimised for real-time operations. Whether it will 'get jiggy' with your vacuum cleaner and microwave like in the TV advert for Australian lager remains to be seen.



**SoftMaker's new WP** is aimed at both home and corporate users looking for speed and compatibility with a small footprint and low cost.

## NEW WORD PROCESSOR

## Taking on Microsoft and OpenOffice

The **Linux word processor market** is about to get a little more crowded with the announcement of a new project from German developer SoftMaker which has just entered its public beta phase.

*TextMaker* is a \$49 application designed, the company says, to bridge the gaps between Free Software which takes forever to load, quick compact apps that are light on features and proprietary applications that cost an arm and a leg. The company confidently claims its word processor

will open and save files for *Microsoft Word 6/95/97/2000/XP* files without losing any formatting or content.

The natively developed application is the first element of a planned office suite which should include a full complement of groupware services.

The timed beta is available from [http://www.softmaker.de/tml\\_en.htm](http://www.softmaker.de/tml_en.htm) and the full release should be available towards the middle of the year. Look out for a review in *LXF* and maybe the inclusion of a free version on the coverdiscs...

## OpenOffice.org gets vertical

## Version 1.1 new features

The beta release of Open Office 1.1 has hit the mirrors promising a raft of new features including long-anticipated support for vertical languages such as Thai, Hindi, Arabic and Hebrew and the ability to output both PDF and Flash (SWF) files.

As well as new features, some of the UI annoyances have also been fixed making for a more consistent and configurable experience. Developers also get XHTML tools and the facility to output flat XML files, while

improvements in bitmap handling means Linux/UNIX users can now copy and paste bitmapped graphics. Progress has also been made in the area of MS file compatibility with the Excel importer now capable of dealing with sheets formatted with bitmaps, textures and hatchings.

You can download the beta (currently only in English but other binaries will be available 'soon') from [www.openoffice.org/dev\\_docs/source/1.1beta/](http://www.openoffice.org/dev_docs/source/1.1beta/)

## David Cartwright

David Cartwright is an IT consultant who specialises in providing Linux systems and solutions.



## COMMENT

## Updating

“Those of us using Linux distributions such as Red Hat or Debian are used to being able to run a simple “Go and fetch all the recent updates” routine – *apt-get* on Debian, or *Up2Date* on Red Hat. It's a really cool way to go about getting the latest versions of software packages without the hassle of visiting the various software makers' sites to keep up-to-date with developments and bugfixes. OK, there's usually a bit of a lag between the software people fixing the bug and the new package appearing on the distribution update system, but the delay's not usually unreasonable.

Hands up, then: how many of us actually use these tools regularly? I do, not for any particular reason except that I can – I've got always-on half-MB ADSL and it costs nothing to kick off *Up2Date* of an evening, even if there's a new squillion-megabyte kernel version to download. I tend to be a bit cautious on my production Web server (Red Hat 7.2) and make sure that I'm at least in the same town as it just in case something goes pear-shaped and it won't reboot properly, but with the number of bugs and security issues that are resolved from time to time in key packages like *Apache* or mail software, I'm always wary of running an old version so I like to keep stuff updated if I can.

As long as one can be confident that the updater checks interdependencies between packages (which is generally true) and that if the new kernel isn't happy you can roll back to the previous one (which is certainly true on my Red Hat 7.2 box at home) these tools are a godsend. So make the most of them, and let your machine handle all the tedious grunt work of keeping its software up-to-date.”



## LINUX HARDWARE SUPPORT

# Hewlett Packard loves Red Hat – official

**H**ewlett Packard and Red Hat have jointly announced an 'expanded' relationship, making the former a preferred vendor for "delivery of single-point fulfilment and service of the complete Red Hat Enterprise Linux product line." IDC analyst AI Gillen said the partnership could

improve Linux's standing in Enterprise across the board, stating that "The combination of world-class support and industry-standard hardware with the price and performance advantages of Linux will appeal to enterprise customers."

HP's Peter Blackmore said the deal would cover the entire Red Hat

Enterprise Linux Server line on both 32-bit and Itanium architectures and provide a single point of contact for support issues. "Today's announcement builds on our \$2 billion in Linux-based revenue in 2002 and our decade of commitment to the open source and Linux communities," he said. Linux accounted for 15 per

cent of HP's pre-installed server business last year, and the cementing of the relationship with Red Hat should lead to a more concerted push to get the open source operating system evaluated as a viable, cost-effective alternative to Windows or proprietary UNIX systems in areas of the IT world traditionally distrustful of Linux.

## Embedded Linux News

● **Motorola** says that development for its next generation of mobile phones will concentrate on Java rather than their underlying Linux OS. Spokesman David Rudd said there were security issues that needed to be addressed before the Linux portion of the pairing would be exposed to end users. "The main issue here is that a phone's operating system must be made secure so that, for example, badly written or malicious code cannot power up the modem and rack up charges on your bill!"



● It has been a long-time coming, but the **YOPY PDA** is now available to buy in the UK. Built around a 206MHz StrongARM processor, 128MB RAM and a 240x320 true colour screen, the YOPY comes in a choice of two models, priced at £375.90 and £431.90 (inc VAT). The more expensive model features a CompactFlash Type II slot, as well as the usual array of connections. Both models come with a complete range of applications including Hancor's highly regarded productivity suite. <http://www.yopypda.co.uk/>

## MADHATTER CANNED

## The Sun ain't gonna shine anymore...

**Before the CDs were even pressed,** Sun has signaled the demise of its own branded Linux distribution. In what Jonathan Schwartz calls the "single biggest shift in [the company's] software strategy in a decade" Sun will instead concentrate on forging links with the major Linux vendors, and as Madhatter – the now-dead Sun Linux project – was a tweaked version of Red Hat, it would seem inevitable that Red Hat will receive the bulk of their attention. In its place, Schwartz suggested that Solaris on x86 (itself a formerly dead project) would gain the ability to run Red Hat Advanced Server and UnitedLinux binaries, and Sun would concentrate on integrating the entire SunONE stack into Solaris and "reducing complexity."

John Loiacono, Vice President of Operating Platforms, said the company was responding to customer feedback. "Our Sun Linux distribution is essentially Red Hat Linux with a few minor tweaks. But our customers told us they didn't want a standard distribution that had some tweaks, so I decided to fix the problem by simply supporting between two and four standard Linux distributions." No decision has yet been made on which distributions will be favoured, though Loiacono said he was in negotiation



**Jonathan Schwartz says Sun will be refocussing its efforts on Project Orion, which will bring limited Linux compatibility to Solaris.**

with all the main players. Schwartz said that such a major change would inevitably lead to license changes with all software moving to a single platform and three licenses: traditional, predictable and metered. This, he said, was "less about being a race between us and the major Linux vendors, and more about the combined effect it has on Windows."

## MARKETING V ENGINEERS

## Centrino confusion

**Chip giant Intel is facing criticism** over the launch of the new Centrino mobile CPU which combines improved battery life, speed gains and integration of 802.11b WiFi technology. Michael Roberts, Lindows CEO has claimed that Intel is "locking out" Linux vendors due to marketers favouring Windows while the engineers are keen to allow Linux users access to the technology. Roberts praised the Intel engineers, saying they were valuable contributors to the Linux community and did "an excellent job of ensuring that the latest chips and motherboards have solid Linux support... the marketing-minded individuals are more worried about the risk of upsetting Microsoft" Roberts points to two events to support his claims; Intel pulling out of the Desktop Linux Summit and the denial of a place for Lindows.com on an Intel product roadshow (coincidentally?) sponsored by Microsoft.

Intel claims, in fact, to have Linux drivers available for the Centrino, but spokesman Scott McLaughlin said consumer demand would dictate whether these drivers are released.

Some pundits have speculated that the delay of Linux Centrino drivers is another aspect of the increasingly protectionist relationship between Intel and Microsoft which, it is claimed, has also seen Windows support for AMD's 64-bit processors constantly postponed.

## FIDDLER FIGURES?

## Marketshare in the air...

## Nicholas Petreley's interpretation

of the latest figures from Evans Data Corp on the state of the Linux market suggests that the commonly held belief that the operating system's gains in market share has been at the expense of proprietary UNIX is wrong. In fact, he claims that developers in particular are favouring Linux over Windows and that, of Linux developers surveyed, 50 per cent formerly worked primarily on Windows. Moreover, Petreley says that behind this figure is a broad trend towards Linux.

"This year, Windows commands attention of 50 percent of the developers. Roughly 40 per cent focus primarily on Linux. These priorities will switch places almost number-for-number next year."

The divergence between myth and reality happened, he says, due to the way market share for Windows is estimated. "Windows market share is usually estimated by the units of Windows Microsoft claims to have shipped," he says. But these numbers are suspect as they include all the

unsold boxes still sitting on shelves and every PC that consumers buy that are pre-installed with Windows. "When someone purchases a PC with Windows pre-installed, and then overwrites that pre-installed Windows with Linux, nobody subtracts 'one' from the installed base of Windows and then recalculates the Windows market share." Upgrading from one Windows version to another, Petreley says, is also counted as a new installation where a counterbalancing uninstall is not counted.

## LINUX UNDER ATTACK

## Coming soon: virus invasion?

Linux will soon become a major target for virus writers as its popularity grows, according to a new study by anti-virus vendor Central Command. The company says that the advent of user-friendly distributions such as Lindows, Xandros and mainstream sellers such as Red Hat and Mandrake targeting home users will make the platform an increasingly attractive target for virus writers.

Steven Sundermeier, Product Manager, said that many users may feel that, as the overwhelming majority of viruses exist on Windows, simply installing Linux is enough to protect them. "Promoters that claim the Linux OS is virus-safe fail to understand that

the user's data is far more valuable than the Linux operating system," he said. While the core of the OS may be ring-fenced, data tends to be stored in user-accessible directories. "A virus or malicious program might not be able to access the core operating system components or gain 'root' privileges but malicious programs only need the current user access to do damage to that person's data."

Inevitably (some would say) the results of this study have been revealed in tandem with the latest version of Central Command's virus protection solutions for Linux servers and workstations.

[www.centralcommand.com](http://www.centralcommand.com)

**CENTRAL COMMAND**  
**Virus Protection for the Real World.**

Home | Buy Now | Products | Support | Downloads | Updates | Contact | News | About

**Protection Solutions**

- Windows
- Linux
- FreeBSD
- OpenBSD
- Sendmail
- Qmail
- Postfix
- Exim

**Top Virus Threats**

- Worm/Klez.F & G
- Worm/Avil.A
- W32/Yaba.E (Lentini.F)
- Worm/BugBear
- Worm/SuBig.A

**Recently Discovered**

- Worm/Lovegate.G
- Worm/Lovegate.F

**Latest News:**

New: Sitebytes Internet, Largest Web-hosting Company in The Netherlands Has Selected Vexira AntiVirus Protects Over 15,000 Domains And Tens Of Thousands Of Email Users [More >](#)

Central Command Predicts Linux Viruses To

It is uncommon to find the better product is also significantly less expensive than its competitors. But, I discovered that the better and less expensive product also comes with the great support. Customers should feel very confident in their choice of critical antivirus software, unless they decide they don't. I provide great support.

George Candiger  
Lemon Pesto Smoothie  
[More customers >](#)

**EDU Sale@School - Special Educational Pricing**

[Submit a New Virus](#) [Join@CentralCommand.com](#)

[Free Subscription to Central Command](#)

Central Command has protection solutions for Linux, Windows and FreeBSD.

## DATABASE

## Greenlight for MySQL 4

## The long-awaited fourth

generation release of the leading open source database system MySQL has finally achieved 'production' status. MySQL confers the production label after 'rigorous' testing to ensure there are no fatal bugs and after a three-month period of battle testing in live environments.

Michael Widenius, CTO and co-founder of MySQL AB said that, though the application had been available for over a year, many users would be waiting for it to 'go production' before upgrading. "Now that it's been thoroughly tested by thousands of users, with no major bugs having been found in many months, we are confident that version 4.0 can be used reliably in any MySQL implementation - whether it is in the enterprise, embedded in third-party software or in a large Web application," he said.

Along with the production release, MySQL AB also announced the completion of an optimisation process for Intel's Itanium 2 chip bringing the application's world-class speed and stability to large databases, datamining and other enterprise applications.

MySQL is the the world's most popular Open Source database server with more than four million installations powering customers such as Yahoo! Finance, MP3.com, Motorola, NASA, Silicon Graphics, and Texas Instruments.

## Jono Bacon

The founder of UK Linux, KDE developer and all-round nice guy, Jono Bacon is studying at Wolverhampton University.



## COMMENT

## Audio Spectra

“An area begging for Linux apps is high-end audio that recording engineers and home recording buffs could use.

In my house I have one Windows machine; it's location is up in my studio. It has no Net access and only really runs two apps that I use for my recording. Although my recording software is awesome, the instability of Windows 2000 is an issue, and one that is frequently complained about on boards like [homerecording.com](http://homerecording.com).

The problem we have here is the age-old one; device support. We have support of many cards such as Soundblasters and such, but unfortunately many of these cards are not used in studios and more specific equipment is used.

Another issue is the software support. There are indeed projects to develop a multi-tracker for Linux, but the common supported VST framework for plug-in audio facilities has had little success being ported over to Linux due to licensing issues. It is an unfortunate situation as many engineers are relying on VST plugins more and more over outboard equipment and as such the vast array of plug-ins would be useful for Linux.

Though Linux is not as feature-full for audio processing/recording presently, the nature of Open Source means that it may well be some day soon; this is an exciting time for musicians and sound engineers as Linux could provide a good recording environment and also the stability and reliability that we have got so used to.

It is an unfortunate fact that the legal issues of technology get in the way for the adoption of a particular standard; VST has proven to be reliable and flexible and the adoption of the standard on Linux would certainly be of benefit to its inventors.



# £1,000,000,000

SCO have filed a suit against IBM, alleging misuse of proprietary UNIX secrets in their Linux contributions. What does it mean for Linux and open source? What does it mean for UnitedLinux? Can IBM fend off this attack and permanently sever ties between proprietary UNIX and Linux? *Linux Format* investigates...

cover feature



**O**n March 7th this year, SCO Group, the current owner of the UNIX operating system, filed a legal action against IBM for more than \$1 billion, alleging that IBM "misappropriated" parts of SCO's proprietary UNIX technology and gave it to the Linux development community. SCO, formally Caldera International, have further alleged that IBM engaged in unfair competition, breach of contract, and interference with SCO's business.

"It is clear from our stand point that we have an extremely compelling case



against IBM. SCO has more than 30,000 contracts with UNIX licensees and upholding these contracts is as important today as the day they were signed", said Darl McBride, president and CEO of SCO.

The initial announcement sparked a flurry of comment, and certainly

generated an amount of anti-Linux FUD (Fear, Uncertainty and Doubt). Many were concerned that the action, while initially filed against IBM, could be applied to many Linux vendors and projects, though in subsequent comments to the community, SCO have outlined that this is not the case. So what is the beef with IBM...?

owned by an AT&T subsidiary, USL, which was then bought by Novell, which was in turn acquired by Caldera. This Bell Labs UNIX is the ancestral codebase of modern UNIX systems such as Solaris, HP-UX, and IBM's AIX, and the term UNIX includes all those operating systems, as well as SCO's UNIXWare.

Sun, HP, and IBM all own perpetual licences to the UNIX source code purchased from SCO or its predecessors in interest, and all three modified that code to work on their own systems.

Starting with paragraph 1, SCO state that "UNIX and SCO/UNIX are widely used in the corporate, or 'enterprise,' computing environment" SCO continue alleging their market dominance in paragraph 23, "Except for SCO, none of the primary UNIX vendors ever developed a UNIX 'flavor' to operate on an Intel-based processor chip set"

These particular paragraphs have been a sticking point for many reading it, because paragraph one is written in such a manner as to imply that SCO UNIX is widely used in the corporate computing environment, whereas it simply says that *UNIX* is widely used - and the large majority of those UNIX systems are not provided by SCO.

Indeed, for their own part, SCO's Securities and Exchange Commission (SEC) filing for this year states "Our business is focused on serving the needs of small businesses, including replicated site franchisees of Fortune



Sam Palmisano, President of IBM, revealed his company's ambitious Linux strategy at LinuxWorld NYC, January 2001

## Anatomy of the suit

The basic breakdown of the suit is a series of claims that IBM gained UNIX technology through licences and partnerships with SCO, and then unlawfully released or made available this technology in the form of code for Linux.

While specific claims in the lawsuit have received mixed responses from the development and enterprise communities, it seems that few people believe that IBM are worried by the case, mostly because of the wording, which many who have studied it say contains somewhat vague and uncertain language.

The 30-page filing contains four core parts: "Misappropriate of Trade Secrets", "Unfair Competition", "Interference with Contract", and "Breach of Contract". However, the most interesting parts lie in the section "Background Facts", where SCO puts forth 103 paragraphs of statements regarding the basis of the case.

In order to better understand the suit, it's important first to understand the history of UNIX. Originally owned by Bell Labs/AT&T, UNIX was later

# lawsuit!

1000 companies" – not what most people would describe as an enterprise computing environment.

The claim in paragraph 23 that no other "primary" vendors ever developed UNIX to work on Intel has been said to be even more spurious, partially because there is historical proof on Usenet that Sun had Solaris working on 386s as early as 1993 and that IBM announced a project to port AIX UNIX to the 386 back in April 1987. Users of FreeBSD were particularly surprised, given that many users have been running FreeBSD, which derived from BSD Unix, on Intel hardware since early 1993.

Some have said, though, that the most damning evidence against paragraph 23 is a post made to the newsgroup comp.sys.intel by Tim Ruckle of SCO on September 25th 1991, in which he said "For the benefit of the entire user base, as well as the industry as a whole, SCO encourages all UNIX System vendors for Intel processors to join SCO, USL, Intel, ISC and OSF in supporting the IBCS-2 standard for x86 applications."

Paragraphs 82 to 86 change the tone of the suit from laying down SCO's version of the state of UNIX software to alleging that the development of Linux was not likely to

get where it is today without IBM's involvement, or more specifically, without the misuse of UNIX intellectual property.

Paragraph 82 states "Virtually none of these software developers and hobbyists had access to enterprise-scale equipment and testing facilities for Linux development."

The primary rebuttal from the community on this front is that Linux shipped an SMP (Symmetric Multi-Processor) capable release as early as version 2.0, and it was stabilised by 2.0.36. SMP is a core feature of so-called enterprise scale equipment, however the key point of interest here is that the hardware to develop SMP was donated to Alan Cox back in 1995 – by none other than Caldera!

Paragraph 83 fires a broadside with: "As long as the Linux development process remained *uncoordinated and random*, it posed little or no threat to SCO, or to other UNIX vendors". Given that IBM didn't join the Linux scene until around 1999, many are perhaps quite rightly upset at having eight years of community development written off as "uncoordinated and random". By 1999, Linux development was already streamlined towards productivity, and Linux itself was making drastic inroads into the

**"SCO is the thief who puts a gun to his own head and says give me your money or I'll shoot... The claims I've heard are specious, and not enforceable in court. Why, then, is SCO doing this? They want to be purchased."**

– BRUCE PERENS



middle-range server market.

While few can doubt that IBM have brought a certain degree of respectability to Linux, and also, of course, that they have submitted a variety of excellent patches to the community, IBM have certainly not changed the way Linux is developed – as a group of the IBM kernel hackers said in a Slashdot interview, "Linus himself is wonderful about accepting patches on technical merit alone. He doesn't 'grade' them differently if they come from ibm.com or mit.edu. We submit patches the exact same way that everyone else does: append the patch, mail to Linus and CC linux-kernel. If it's good, it gets in. If it sucks, you get flamed."

However, if the previous statements were decried as being dubious, paragraph 84 had even more to offer: "Prior to IBM's involvement, *Linux was the software equivalent of a bicycle. UNIX was the software equivalent of a luxury car*. To make Linux of necessary quality for use by enterprise customers, it must be re-designed so that Linux also becomes the software equivalent of a luxury car. *This re-design is not technologically feasible or even possible at the enterprise level* without (1) a high degree of design coordination, (2) access to expensive

and sophisticated design and testing equipment; (3) access to UNIX code, methods and concepts; (4) UNIX architectural experience; and (5) a very significant financial investment"

As we have already seen, the Linux development has had a high degree of co-ordination for a long time now, owing largely to its streamlined, distributed development process. Furthermore, we have seen that Caldera donated hardware to Linux developers in order to speed along production of an enterprise-level kernel. With regards to the requirement of "a very significant financial investment", surely the opposite is true – open-source development seems to do rather well for itself despite operating on a shoestring budget.

It is, however, item 3 in the list that strikes some as peculiar, because SCO/Caldera make available online "for UNIX enthusiasts" the copies of the source code for 5th, 6th, and 7th Edition UNIX, although to be fair these are covered by a licence restricting their use. However, BSD is one of the most popular UNIX systems available, and it is of course entirely Open Source. Ultimately, it could prove difficult for anyone to conclusively prove what code is really owned by whom.



## The key claims

### The alleged wrongs SCO wants put right

- IBM misappropriated SCO's trade secrets and confidential information.
- IBM engaged intentionally and foreseeably calculated to undermine and destroy the economic value of the UNIX source code "anywhere and everywhere in the world".
- IBM contributed trade secret protected software code for incorporation into one or more Linux or other free UNIX-like software releases.
- IBM induced and encouraged others to violate confidentiality provisions and to misappropriate trade secrets and confidential information.
- IBM subjected SCO's UNIX trade secrets to unrestricted disclosure, unauthorised transfer and disposition, unauthorised use, and has otherwise encouraged others in the Linux development community to do the same.
- IBM's misappropriation was willful, malicious, and in reckless disregard
- SCO stands at imminent risk of being deprived of its entire stream of all UNIX licensing revenue in the foreseeable near future.





IBM have been coincidentally been investing \$1billion a year in their Linux strategy.

« However, as SCO have stated that paragraph 84 is a fact, they build upon that fact in paragraph 85, saying: "For example, Linux is currently capable of coordinating the simultaneous performance of 4 computer processors. *UNIX, on the other hand, commonly links 16 processors and can successfully link up to 32 processors for simultaneous*

*operation...*The ability to accomplish this task successfully has taken AT&T, Novell and SCO at least 20 years, with access to expensive equipment for design and testing, well-trained UNIX engineers and a wealth of experience in UNIX methods and concepts."

This particular paragraph will prove particularly interesting once the case

gets to court, for two reasons. Firstly, Linux is well-known to run on systems with many more than four CPUs. For example, according to the Linux Documentation Project, UltraLinux (Linux ported to UltraSPARC) runs on 14-CPU SPARC-based machines, all the way up to 24-CPU Starfire E10000s. Additionally, Peter Rival (of Compaq/DEC at the time) uploaded the boot messages of a 31-CPU AlphaServer booting up Red Hat Linux with no problems. By this point, he was actively working on Linux support for the Alpha system.

Secondly, SCO's own release of UNIX, SCO OpenServer, cites "Support for systems with up to 4 CPUs", so it's almost like SCO have got this claim entirely backwards!

Paragraph 86 pretty much sums up SCO's position: "*It is not possible for Linux to rapidly reach UNIX performance standards for complete enterprise functionality without the misappropriation of UNIX code, methods or concepts to achieve such performance, and coordination by a larger developer, such as IBM.*" (emphasis added)

As seen above, it has been argued by many that it was indeed possible for Linux to advance as fast as it has done, mostly because community support – and increasingly

commercial support, but not necessarily from IBM – has been behind kernel development.

## The Suit Against IBM

To this point, the suit was aimed pretty squarely at attempting to prove that Linux is not ready for enterprise use. However, there are five paragraphs towards the end of the suit that are aimed at proving IBM was directly behind Linux's speedy development cycle, and that they wanted to help Linux in order to destroy SCO's UNIX business.

At first, many Linux users didn't bother commenting on these parts of the suit, particularly because of their

## UnitedLinux

### Keep your friends close and your enemies closer...

SCO are one of the founding members of UnitedLinux, along with Conectiva, SuSE, and Turbolinux. As SCO have been so roundly criticised over this lawsuit, will this affect the UnitedLinux partnership?

SuSE are, paradoxically enough, also close partners with IBM, which seems to leave them CAUGHT in between two warring parties. Sources from inside SuSE have said that there have been high-level discussions taking place recently to try to resolve the potential conflict of interest here – it will be interesting to see where those high-level discussions lead.

In the meantime, SuSE's official statement is as follows:

"We at SuSE were greatly disappointed to learn of the SCO Group's recent actions. While we agree that SCO has every right to enforce

their intellectual property rights, and while we strongly believe that this does not impact Linux (as even SCO has made clear), we are concerned that these actions are not in the best interest of customers, partners and the Linux community. Accordingly, we are currently re-evaluating our relationship with the SCO Group. That said, we want to very clearly and unequivocally voice our support of the ideals and goals of UnitedLinux and the Linux community."

The other major partner in United Linux, Conectiva, also have reservations about SCO's actions. In an interview given to MozillaQuest, Conectiva's Gordon Ho has criticised much of the logic in SCO's preamble, asserting that various versions of Linux were enterprise ready before IBM were even interested in the OS.



Darl McBride, President and CEO of The SCO Group.

focus against IBM. However, more and more people are taking the opinion that IBM, despite being a very large company, deserve all the support they can get. It's also here that the case starts to stray onto particularly hazy ground – it attempts to prove that IBM has taken code from the proprietary UNIX codebase and dumped it directly into the Linux source code.

At first, it might seem fairly cut-and-dried to prove code copying – simply compare two lots of source code and look for an undue amount of similarity between the two. However, many people in IBM who previously worked on their AIX UNIX system were moved to Linux development at a later date, so it's quite possible that code designed to accomplish a particular task will naturally look fairly similar – after all, it may well have been the same programmer writing the code.

A key paragraph in the case is number 92: "IBM quote: Linux cannot fill that need today, but over time we believe it will. To help out we're making contributions to the open source movement like the journal file system... We're willing to open source any part of AIX that the Linux community considers valuable"

AIX is IBM's port of the UNIX code to its own platforms, originally AIX would have been almost entirely proprietary UNIX code. However, IBM apparently also built AIX on BSD code (not covered by UNIX intellectual property) to a great extent, and changes would have been made by them to add support for new features, optimise slow parts of the system, remove old features, etc.

Few outside of IBM are likely to know how much of the original UNIX source code remains in AIX, and this is where SCO are unhappy: if there are parts of AIX which do retain the original UNIX code, then it would be a breach of trade secrecy if IBM were to give such code away.

IBM's eagerness to help Open Source software may cause problems, as the suit highlights two quotes from IBM executives that in hindsight – particularly if it is found that AIX still contains UNIX code – would look particularly embarrassing. In paragraph 95 of SCO's complaint, we see "IBM's AIX contributions were integrated into the standard Linux source tree, a win for open source", then, in paragraph 98, "IBM will

exploit its expertise in AIX to bring Linux up to par with UNIX"

### Carrier-grade Linux

One of the last claims made in the suit is in paragraph 102c, where it is alleged that "A carrier-grade Linux project has been undertaken to use UNIX code, methods, concepts, and know-how for the unlawful purpose of transforming Linux into an enterprise-hardened operating system"

Bill Weinberg of Montavista suggested that as a company, they were 'unconcerned' with SCO's action, as their work on hardened Linux was not based on UNIX code. As a founder member of the Open Source Development Labs Carrier-Grade Linux Working Group, Montavista helped create the spec for *Carrier Grade Linux* and are currently the only ones shipping a product that complies with this specification, and so regard the claim that their carrier-grade functionality owes something to SCO with a certain amount of derision.

### What are the possibilities?

If IBM were to lose this case, where would it leave Linux, and indeed what would it mean for the Open Source community as a whole?

If it is found that Linux does indeed contain 'contaminated' code, then the method of resolution is likely to be entirely down to SCO – they can force the removal of offending code and claim extensive damages from IBM, or they allow the community to retain the code and still claim extensive damages from IBM. However, from discussions on the kernel developer lists and elsewhere, it seems that IBM has contributed little code that could fall under remedy terms of this action to the kernel itself.

In this event, Free Software developers across the world would need to be that little bit more careful when accepting code from companies that have potentially conflicting interests, in order to ensure this situation is not encountered again. As far as the community is concerned, this kind of lawsuit can only be bad – potential users may see proprietary developers such as Microsoft as being unaffected by these legal issues, and thus potentially a better bet.

However, if IBM comes out as the winner in this suit, it should hopefully sever any potential legal ties perceived

## COMMENT

Nick Veitch, Editor – *Linux Format*

Whether or not IBM have acted unlawfully, which is obviously a matter for the court to decide, it seems that this case will change things. It won't change Linux. It seems that any exposure to 'tainted' code would be very limited. The likelihood of a successful SCO following up this action with other claims against Linux developers or vendors is slim.

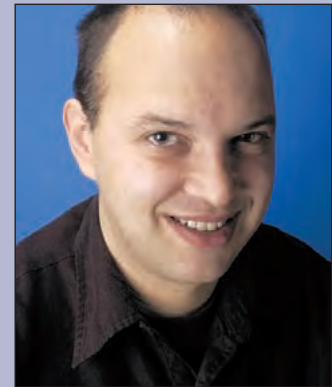
It won't even change IBM's relationship with Linux. Linux is not just a convenient part of their strategy, in many ways it *is* their strategy, and they have certainly invested much more money in Linux than they stand to lose.

From what we have seen, most of SCO's preamble about Linux, in my opinion, doesn't really stand up to much scrutiny, whatever the validity of their claims about IBM. It is important though, because it forms the basis of their damages claim.

The law in cases like this can be unpredictable. If it does go to a jury trial, it could be even more unpredictable. But maybe a trial isn't part of SCO's plan at all. The worst-case scenario for them would be if it went to court and they lost. There are other options – they could hope that the unpredictability of the outcome forces IBM to settle. Or they could hope for a buyout. With their main assets including

a fair amount of intellectual property, one logical purchaser might be the largest single holder of technology related intellectual property – IBM.

Win, lose or buyout though, it seems with the unnecessary degrading terms used to describe Linux, SCO has burnt its bridges with the Linux community as a whole. By implying that the whole Linux community is a bunch of uncoordinated morons incapable of creating 'enterprise' software without stealing code, SCO will find themselves crossed off quite a few people's Christmas card lists. It also puts their Linux work in jeopardy, and further strains the relationship with the rest of the United Linux consortium.




**"SCO has burnt its bridges with Linux...by implying that the Linux community are morons incapable of creating 'enterprise' software."**

between proprietary UNIX and Linux, keeping Linux free for the masses.

The suit was filed in Utah, SCO's base of operations, but IBM have recently asked for it to be moved to federal courts. The process of filing suit in one's own state is such a common tactic in the US that it has its own name, "home-towning". Having a case conducted in your own state – where the judges need to win re-election regularly – is often thought to be biased towards the company making the filing, on the grounds that the presiding judge may favour local companies. Federal court judges, however, have life-long

positions, and so are not considered to be easily swayed.

Regardless of where the suit will be heard, SCO have already done well for themselves. From a low of just above \$1 a share part-way through February, SCO stock is currently selling at \$2.84, with particularly heavy trading taking place (and a corresponding share price jump) immediately after the announcement.

How the suit will pan out remains to be seen, however it has already sent shockwaves through the community, with the promise of much more to come. 



## PREVIEW

# Infosecurity Europe 2003

The Infosecurity event is the biggest security show on the calendar for the UK. This year it will be bigger than ever, with an unprecedented number of new product launches and more Linux-related security solutions than ever before.



**Tripwire** will be demonstrating how organisations can keep up with today's risk factors, with the launch of Tripwire for Servers 4.0, which instantly detects changes to server data wherever it originates from, enabling organisations to lock down vital systems and protect its information.

**SSH** will be there, exhibiting the proprietary software built on top of secure shell technology, which includes IPSec and authentication software for the desktop and enterprises.

**IT Security Audit** is offering visitors the opportunity to attend a no-cost attack evaluation surgery. Using their unique 'threat assessment engine', security consultants will identify areas where a business may be at risk. To encourage businesses to review their IT security measures, IT Security Audit is offering a no-cost attack evaluation surgery.

**Computer Associates** are Launching their new eTrust Antivirus 7.0. Unique Features include: Digitally Signed Signature Updates; Roaming technology; Free Signature Updates; Signature Updates Distributed Daily; and an Uninstall Utility

## Education Programme - London room

### TUESDAY 29TH APRIL

**Security: Corporate governance**  
**Deloitte & Touche**  
**14:00 – 16:00**

Yag Kanani will discuss Deloitte & Touche's experience and the business case for IT Governance. Vernon Poole will describe the effective tools that exist to establish a sound framework of information security management. Followed by an Information Security Risk Workshop to assess how attendees rate their organisational security risk profile using the governance maturity model to establish rankings.

### WEDNESDAY 30TH APRIL

**Security in a connected world**  
**Check Point Technologies Ltd**  
**10:30 – 12:30**

Mik Stevens, Solutions Evangelist, responsible for EMEA region, will identify the growing risks to organisations and demonstrate the type of solutions that should be deployed to ensure the security policy is implemented correctly.

**Best practices in secure messaging – Creating corporate policies for email usage**  
**Clearswift**  
**14:00 – 16:00**

Delegates can ask specific questions regarding regulatory compliance and legal liability and how to establish and implement policy.

## Education

One of the best things about the show is its free education programme that has something for technical experts, general management and those at board level. There will be six keynotes and two distinct streams of seminars in 2003, one covering technical issues and the other business strategy.

The exciting keynote line up starts with an opening keynote by Dr Stephen Marsh director for security policy, Office of e-Envoy. The e-Envoy, is leading the drive to get the UK online, ensuring that the country, its citizens and businesses derive the maximum benefit from the knowledge economy. He co-ordinates the government strategy, ensuring e-access and training, galvanising UK business and driving the e-agenda through government.

Another keynote is, *Policing The Digital Frontier* by Detective Chief Superintendent Len Hynds.

A full timetable is featured here. On the page overleaf we also have a short comment from exhibitors Wick Hill...

## Education Programme - Theatre Room

### TUESDAY 29TH APRIL

**Opening address by Office of eEnvoy**  
**10:30 – 11:00**

Dr Stephen Marsh, Director for Security policy within the Office of the e-Envoy, will give an overview of the Government stance on today's e-commerce situation.

**The European initiatives on network and information security**  
**11:30 – 12:30**

Frans de Bruine, of the European Commission will explain how the exposure of Society to new vulnerabilities and threats to security has led to the EU proposing a number of regulatory measures and policy initiatives.

**Is information security a legal requirement or just good business practice?**  
**12:45 – 13:30**

David Griffith, Partner, eCommerce, Clifford Chance will take a critical look at the assumption that businesses are under a legal duty to take steps to protect the security of their information.

**Calculating the business case for security**  
**14:00 – 16:00**

Nick Coleman, Chairman of the Security Alliance for Internet and New Technologies, will present the findings of a survey on Investing in Security in today's market. The panel session entitled "Investing in information security and infrastructure" explains how to sell the information security to executives.

### WEDNESDAY 30TH APRIL

**Policing the digital frontier**  
**10:30 – 12:30**

Len Hynds, Head of the National Hi-Tech Crime Unit (NHTCU) will open the session outlining the Unit's "Hi-Tech Crime Strategy and the Threat facing UK businesses". John Lyons and Tony Neate, will discuss the results of the NHTCU/NOP survey "Hi-tech crime – the impact on UK business".

**The proliferation of wireless technology**  
**13:00 – 13:30**

John Doody, Head of Customer Services for CESG, will address the various types of wireless technology and provide solid guidelines and recommendations for the safe and secure deployment of wireless devices and networks.

**IT security expert advice clinic**  
**14:00 – 16:00**

An advice clinic of six experts hosted by [www.itsecurity.com](http://www.itsecurity.com), one of the largest single sources of security knowledge on the Internet, will answer any technical question posed to the panel from the audience of security professionals and end users. Areas covered include data protection, content security, human rights, spam and spyware.

### THURSDAY 1ST MAY

**Strategy, delivery and assurance**  
**KPMG**  
**10:30 – 12:30**

A panel of KPMG partners will describe how senior management can better understand the triumphs and pitfalls in dealing with security. Subjects covered include: the latest security risk, threats and strategies; how outsourcing security can affect an organisation's security strategy; wireless intrusion detection and the need for application security testing.

**Critical national infrastructure defence – Hacking Panel**  
**14:00 – 16:00**

A Live hacking panel debate moderated by Bob Ayers, Independent Consultant, will highlight the critical issues concerning National Infrastructure Defense Strategy and the current UK Government objectives. The panellists will look at the perception of the private sector and what you should be doing about it.



# SECURING SOURCE CODE

**Ian Kilpatrick, chairman of the Wick Hill Group, discusses the validity of obfuscating PHP code.**

**P**HP has grown to be the most popular scripting medium for the majority of web development on *Apache*. Ease of use and the fact that it's Open Source are two of the reasons for its success. The LAMP environment – *Linux*, *Apache*, *MySQL* and *PHP* – is the working environment of choice for many in the Open Source community. However there is one area where this environment can present problems – that of securing commercial source code. Because *PHP* is so prevalent, more and more commercial code is developed using it. With so many major commercial web and Intranet applications using *PHP*, business and legal issues are starting to arise.

Companies and developers are having to address issues such as the ownership of the code, the licensing of applications, the distribution of commercially prepared code and the protection of copyright in expensively developed code. When commercially developed source code is copied by someone else, it brings up the question of intellectual copyright. If you're working full-time for an employer, if you haven't signed an agreement or contract to the contrary, then the copyright on any work you produce for that employer belongs to them. They will therefore want to protect their investment in

applications that provide them with commercial advantage.

## Commercial development

With *PHP* and *Apache*, Open Source directly benefits the whole community in that improvements are to everyone's advantage. However, with commercially developed *PHP* applications, using open source code can often directly benefit competitors.

Organisations commissioning the development of leading edge applications or great sites are proving increasingly keen to ensure that the work that they pay for is protected from being plagiarised by commercial competitors. They are clearly not going to be happy if they've spent millions of pounds developing web systems and a competitor comes along, copies their work and it costs the competitor nothing. It is this fear that leads to mistrust of Open Source.

In essence, if you're writing for commercial applications, there is a commercial and legal need to secure code. For financial organisations and the like, there is also the need to ensure that the code cannot be lifted and used to defraud customers, who could be misled into believing they are carrying out transactions with the reputable site. In some cases, senior management isn't even aware that there is an issue with unsecured code, and that they are at risk of losing their commercial advantage or leaving themselves open to fraud and other dangers.

For developers, important issues are protecting the intellectual property of their work, software license management and version management. Managing license and version deployment as well as preventing inadvertent or deliberate copying of code is clearly important for the business of developers, as well as for their reputation.

Web developers have found that if they've spent time, money and creative energy in developing applications that exceed customers' expectations, they don't want competitors copying the code and passing it off as the kind of quality they could deliver. It devalues the original developers' efforts and costs them business. A further drawback that many organisations have found in commercially deploying unprotected *PHP* applications is that users can easily modify them, adding or adjusting features.

While on the face of it this may not seem to be much of a problem and is in the true spirit of Open Source, the reality has proved to be considerably different. For a number of organisations, it has meant having to spend considerable time and effort identifying, supporting and then replacing unauthorised coding modifications.

## Security risks

Unprotected code also creates an unacceptable security risk for some organisations because their business critical code can be modified, with potentially disastrous effect. This could be a problem in certain cases such as mission critical web systems, hospital systems or the systems of financial institutions.

Many of the issues above are being addressed by suppliers such as Zend. Zend is, of course, closely involved with *PHP* and has identified the balance needed between Open Source needs and commercial requirements.

In addition to their involvement with the development of *PHP*, Zend has produced commercial code protection products including the *Zend Encoder* and the *Zend License Manager*.

The *Zend Encoder* protects commercial *PHP* applications by encoding *PHP* scripts, making it less likely to fall victim to plagiarists. Companies get the commercial benefits of knowing their code is secure and developers are able to



## About the author

**Ian Kilpatrick**

Ian Kilpatrick is chairman of Wick Hill Group, a company specialising in secure infrastructure solutions for ebusiness. Contact 01483 227 600, email [info@wickhill.co.uk](mailto:info@wickhill.co.uk), web [www.wickhill.com](http://www.wickhill.com). For further press information please contact Annabelle Brown on 0192 252 8548, email [a\\_brown@dial.pipex.com](mailto:a_brown@dial.pipex.com)

continue using a leading Open Source language, while protecting their key code for when they are ready to distribute their applications.

*Zend License Manager* lets developers manage the commercial distribution of their *PHP* applications. It generates license keys and tags files that require a license key to operate. Developers are able to concentrate on expanding their available markets, improving customer loyalty and increasing revenue and profitability.

Code sharing and commercial needs don't always go hand-in-hand. However, Zend's approach of using their revenues from solutions for the commercial use of *PHP* to fund Open Source development, seems to strike the right balance between business and LAMP. Since, there are clearly times when securing commercial code is absolutely crucial, we are likely to see this happening much more in the future. **LXF**

## Infosecurity expo

**Wick Hill Group**

Wick Hill Group are exhibiting at Infosecurity Europe, the largest and most important information security event in Europe. Now in its 8th year, the show features Europe's most comprehensive FREE education programme, and over 200 exhibitors at the Grand Hall at Olympia from 29th April – 1st May 2003. [www.infosec.co.uk](http://www.infosec.co.uk)



# Mailserver

Share your opinions, right wrongs and demand justice by writing to *Linux Format*. Drop us a line at: **Linux Format**, Future Publishing, 30 Monmouth Street, Bath BA1 2BW or email: [lxformat@futurenet.co.uk](mailto:lxformat@futurenet.co.uk)

## Linux is ready!

Having experimented with Linux from Red Hat 5.2, where I could install but not log in, (the documentation didn't think it necessary to mention your user name was root!) I find that Linux has moved an awful long way since. Is Linux ready for the desktop? Yes!

I use Linux at work to run my driving school, until very recently this was 95% Linux 5% Quicken, (Windows emulated using *win4lin*), but with the release of *GnuCash* 1.81, Windows is now history. All accounting, spreadsheets, document processing, web site design, letterhead design etc – 100% Linux.

At home scanning, digital photography, small experimental network of two desktops and a laptop 100% Linux. The only thing that was missing was a few games, so I bought a few secondhand Loki games from the Internet (connection, browsing and email 100% Linux) and installed them. All was well, the Loki installer was

superb, but my Mandrake 8.1 installation was too new for some of the games. Even so I persevered and all are now working. However not all are completely up-to-date. Is there any chance that you could put all the downloads and FAQs and demos for these games on your DVD?

I subscribed to *Linux Format* before it was released and have enjoyed every edition, have learned a great deal, but still have a lot to learn. For example I have never recompiled the kernel, and feel that a script that automatically detected your hardware and asked a few questions would make this less daunting, and allow more people to see the raw power of Linux. Perhaps an article on what all of the options in **make xconfig** explained in simple detail would help readers like me.

Can Linux overtake Windows? Hard to say, there is an awful lot of inertia to overcome, but mags like yours and the marketing power of the large computer companies might yet do it. I look forward to



At present you can search within the *Linux Format* coverdiscs.

keep learning and I look forward to seeing the version numbers reach 12 & 13 and new converts saying "I started with Mandrake 9.0, and I had to change some of my hardware, but now Linux is ready for the world."

Steven A MacIntosh, [www.learn-and-pass.com](http://www.learn-and-pass.com)

Thanks for your email – your business

is living proof that you don't have to be a multinational to make Linux work in the enterprise sphere!

## Search for progs

I have by now accumulated quite a stack of *Linux Formats* and accompanying CDs, to the extent that it is becoming a bit of a chore to search through previous editions

## ★ Letter of the month

This month's winner receives a copy of *A Practical Guide To Red Hat 8* by MG Sobell, ISBN 0-2017-0313-0

### Linux training

I am 22 years old and live and work in an area of Johannesburg called Hillbrow, South Africa. I am no stranger to GNU-Linux/UNIX and have worked a lot with both systems and really enjoy their power and flexibility. I am currently running an Education project for the underprivileged in my community for free.

A friend and I sponsor the teaching and our students receive

training free of charge. We do the work voluntarily without pay and also provide the hardware on which we train out of our own pockets.

Our training is based on Open Source technology and we currently run Red Hat 8.0 Pro on all our machines (for a .0 release its not too bad).

I just want to say a big thank you to the Linux community for making a project like ours possible and also to ask if there is any way in which we could get recognition

for our students for the training they are doing on Linux from any of the GNU-Linux bodies?

Andrew Gargan, South Africa

The most globally recognised certification is probably the Linux Professional Institute's Stage 1 and Stage 2 qualifications. The exams are comparatively cheap to take (about £60 in the UK), but not free, unfortunately. The LPI has an arrangement with Pearson VUE for doing the exams, and there is a centre in South Africa.



It is probably worth contacting the agent there (0800-995044) to see what can be arranged. It may be possible to arrange a reduction on the exam fees, or perhaps there may be some training funding available for you?

In the meantime, as a small contribution to your efforts, please accept the *Star Letter* prize this issue – and do keep in touch to let us know how you and your students get on.

to find an article or software package. I'm sure it is there somewhere, but I don't remember which issue or even the exact title. Today, for example, I was trying to install "audio-transcriber" from your March 2003 edition, and I needed some pre-requisite Tk modules from CPAN. My network connection was giving trouble and several download attempts failed. It occurred to me that the modules might well be on one of your previous CDs, but the prospect of searching through them all didn't exactly appeal! Do you have an index or keyword search facility for your magazine and/or CDs? It would be really useful.

*David Keegan, via email*

The CDs and DVDs themselves have a limited search facility – you'll find it as a Java applet on the HTML index on every disk. At the moment, this is

**strace -p pid** works fine. These commands are a hacker's delight ('hacker' with the old-fashioned meaning). They trace the calls made by a program to the kernel interface. You need a bit of knowledge to make sense of the calls, but they also give a great opportunity to learn just what programs are really doing. Here's an example of how I used **strace** recently. I was trying to do recordlocking from Perl using *fcntl* (don't ask), and couldn't get the Perl Cookbook example for Linux to work. I had it working in C, so I knew what it should be doing. I ran the Perl script under **strace** and found that *fcntl64* was being called. That gave me the clue that maybe the Perl call needed 64-bit arguments. I changed the script to allow for that factor and hey presto, it worked!

heat management is vital for modern processors – a P4 1.3GHz takes about 40A @ 1.75V, that means that about 70W of power needs to be dissipated from the chip with a maximum permissible temperature of about 70 degrees C – imagine trying to keep a 60W lightbulb at a temperature that you can still touch it when it's on!

The second point concerns the IDE buses. Prof Hatton says that you can 'tweak a bit extra performance' by using separate IDE buses for the hard disk and the CD-ROM, but in general he seems not to think it worth it. However, I think he is wrong on this point. An IDE bus will operate at the rate of the slowest interface, most CD-ROM drives use an ATA-33 interface, whereas modern IDE hard disks use ATA-100/133 – this means that if you put both devices on the same bus, the disk interface will operate at about one-third of its maximum rate. I will admit that probably most people won't notice the difference in normal usage, but as soon as a box starts swapping, you will wish your disk were faster!

Motherboards usually come with two types of IDE cable: the one with the higher density of conductors is for the high speed devices, so should be used for the hard disk, the other cable is used for the CD-ROM. The high density cables often have 'Master' and 'Slave' printed on them near the connectors. You should plug them in correctly – the master device should go at the end of the cable as that is the device that has the terminators in it, connecting the master in the middle will result in lower transmission rates and possible errors. Setting your disk drive jumpers to 'Cable Select' will make sure that the right device is the master.

*Pete Biggs, via email*

Thanks for your CPU and ATA tips. I think it is a very good idea to stick hard drives on a different channel from a CD-ROM or CD-RW, if only to make data transfer between them happen more smoothly.

## What's the word?

You regularly print a short 'jargon buster' in the form of your *Essential disc info* page but perhaps it would help other neophytes if >>

## "Can Linux overtake Windows? There is an awful lot of inertia to overcome, but mags like LXF and the marketing power of large computer companies might do it."

limited to searching that specific disc, rather than all the back issues too, though hopefully we will be able to incorporate this either on the discs or the website too in the future.

However, since about issue 16 or so, all the discs contain index files. In older discs, these were comma separated text files, and in later discs this has been replaced by XML data (look in the .index folder). You could copy all these and just run *grep* to perform a simple search until the full search facility is available.

## Trussed up

I was interested in Elaine Wright's letter in LXF38. She mentioned truss on Solaris, and said that some Linux variations don't have this command. The command **truss** came from Unix SVR4, and it had an equivalent on SunOS: **strace**. The **strace** command has been ported to Linux, is shipped with Red Hat, and might well be available with other distros.

For all practical purposes **strace** is the same as **truss**, for example

Check out the man pages for **strace**, you will wonder how you got by without it.

*Clive Darke, via email*

Many thanks for your elaboration. There usually is a Linux answer if you look hard enough!

## Burnt goo

I have a couple of comments on the article 'Backup Server' in LXF37. It is good to see such an article in your mag: the more the insides of a PC are de-mystified the better.

Fitting the fan to the processor is more involved than is suggested in the article. Some form of thermal paste should be applied to the joint between the fan and the CPU in order that good heat transfer is obtained. Sometimes the fans come with a pad on the bottom that contains the necessary goo, but you must take the covering off the pad first (I have seen a burnt out chip and a very black-looking lump of charred plastic where someone forgot to take the protective cover of the thermal transfer pad). Good

# UNIHOST

Specialists in hosting dynamic web sites & e-commerce stores with secure SSL.

- Linux Hosting Specialists
- CGI / SSI
- PHP 4
- Perl / DBI
- MySQL
- PostgreSQL
- JSP / Java Servlets
- FTP / SSH Access
- Full Logs & Web Stats
- Huge Network Backbone
- No Data Transfer Limits
- 100 MB Storage
- Full Shell Access
- Monitored 24 x 7 x 365
- UPS
- POP3 E-mail Boxes
- Unlimited Aliases
- SSL Option Available
- Reliable Service



£250 per year (ex VAT)

**SPECIAL OFFER:**

**20% off 1st year's hosting if booked before June 2, 2003. (£200 ex VAT) Quote ref: LF503M**

sales@unihost.net

**0870 744 0721**  
**UNIHOST.NET**

[www.linuxformat.co.uk](http://www.linuxformat.co.uk)



# Mailserver



**Fiddling and tweaking is part of using *any* OS – don't let it get you down!**

« you could print a general glossary occasionally and/or be more careful about checking that abbreviations/acronyms are defined in articles which use them (CPU and RAM excepted!). In the real world, there isn't always a Linux guru round the corner to ask!

*Anthony Cummings, The Netherlands*

A fair point. In general though, we do assume that, if for example we were doing a tutorial on an advanced topic, the readers interested will have some knowledge of the relevant jargon. We don't want to be using up space just for the sake of it, but we will include

jargon boxes where appropriate. Most unusual terms are explained in the text anyway.

## Not for everyone?

Can I just say that LXF is a stunning magazine and one that most PC mags should take a close look at! I've recently taken off my dual boot of Mandrake 9 and Red Hat 8 and reinstalled WinXP. Why you may scream? Well I realised that even when I had a system up and running under Linux I still spent a vast amount of time fiddling, tweaking and generally trying to

**“I would be interested to know how much time the contributors to LXF actually spend using their programs as opposed to trying to get things to work.”**

get Linux to do the things that Windows does quite easily.

Whenever I looked at the amount of time I spent trying to get it to work compared with the amount of time I spent actually using the platform to do stuff It just didn't compare. This is where Linux falls down as a desktop platform. The average user isn't interested in fiddling with it! They just want to use it, something that Windows does do successfully (even if it does pain me to admit it!) I would be interested to know how much time the contributors to *Linux Format* actually spend using programs as opposed to trying to get things to work.

Maybe the Linux community should listen to Hoyt Duff in your March edition and concentrate on better code and making sure the systems have less compatibility problems and are easier to

administer. After all, most of us would gladly forgo functionality for stability and usability.

*Carl Maycock, via email*

At *Linux Format*, we spend most of our time trying to get things to work, but as that is a large part of our job, it isn't too surprising. Often though, the trouble is not really to do with Linux, but to do with other computer business' attitude towards it. The amount of time we've spent trying to get various bits of hardware to work beggars belief, but the real problem here is lack of manufacturer support, a situation that is, albeit slowly, changing for the better.

There's nothing to be ashamed of if you find Linux doesn't work all the time for you, it's something that most users experience, but it is helpful if readers could tell us specifically what they had problems with, so we could address these in tutorials or other features in the magazine.

# Helpdex

shane\_collinge@yahoo.com





## OpenOffice.OZ

I'm an engineering student on internship for a electrical manufacturing company. I have to run MS software as this is the office (and customers) standard. With the latest releases of *OpenOffice.org* and your reviews I felt my troubles might finally be over as I could run Linux on my laptop for work-related issues. This was not the case. I have had trouble with two areas in *OpenOffice.org* as I dual-booted to test using Linux over Windows.

**1** There is no Database software. *StarOffice* comes with this and so would not be surprised if it was left out to ensure sales of *StarOffice*. However I was wondering how the *StarOffice* equivalent rates against *Access*, and especially the portability of the files from the *StarOffice Access* clone.

**2** Most important for me are macros. I use the MS VBA language a lot in my work of analysis of trends and converting between databases and spreadsheets. However when going to load up my spreadsheets for work, I found everything working better than before, except my macros. *OOo* had commented them out, buttons to run macros didn't work and most importantly it did not understand the VB code when I uncommented it. This is very annoying. I was wondering if at some stage Visual Basic support will be given for macros so there will be the ability for portability back to *MS Office*. However I expect this will not be free, which is the most appealing aspect of *OOo*.

Sorry if there is a lot of complaining about *OOo* in this

letter. It is the best office software I have seen, far surpassing *MS Office*; however people like me can rarely be bothered to find the time to write in praising something. Its not until we find problems that we start making a noise.

*Oliver Thane, Sydney Australia*

We did cover this in the extensive Office roundup (*LXF30*) and in subsequent tutorials. The things you mention are the main differences between using *StarOffice/OpenOffice* and *MS Office*, so just to recap on them once again.

**1** There is no GUI driven database application either in *OpenOffice* or *StarOffice*. *StarOffice* does contain the adabas database server, but this is like DB2, MySQL or other database servers – it does not contain a GUI front-end for end users. The reason for including the database is that both *StarOffice* and *OpenOffice.org* can make extensive use of this data for wordprocessing etc. The reason it isn't included in *OpenOffice.org* is due to licensing issues.

**2** There is no Microsoft Basic for Linux, surprisingly enough. Macros in spreadsheets can be created using the *StarOffice/OpenOffice* macro language, but you can't run the MS macros. The reason they are commented out is to preserve them so you can use them in *MS Office* at a later date.

## Modem magic

In the letters page of your last issue, S. Ionii talked of problems getting his Conextant Modem to work. I have a similar modem and Mandrake 9.0 and the trick to get the driver is to download the source code (not the RPM) unzip it, compile it and install it with the usual steps:

`tar zxvf packagename.tgz` ➤➤



Commented out macros are preserved for future use in MS in *OOo*.

# root: /#

## £15 per month

### Linux Virtual Machines: simpler, cheaper, more reliable than co-lo or managed hosting.

- Install any software you like, configure it however you like.
- Total control over your kernel: use iptables, IPv6 or IP tunnelling to build your own network and permanent online presence.
- Preconfigured software for superior hosting: Apache, MySQL, PostgreSQL, PHP, Perl, Python, Ruby & full gcc compiler suite (among others). Choice of Linux distribution.

### Included in our service at no extra charge:

- Watchdog with SMS and email alerts, or automatic reboots. Know about server problems as soon as they happen.
  - DNS hosting: upload your domain data to our four DNS servers for resilient service.
  - Mirrored (RAID 1) storage and nightly backups of your machine prevent data loss and down time.
- ### Quality network & support
- 100Mb multi-peered, multi-homed network in Telehouse, Docklands.
  - Friendly telephone or email support, whatever your experience of Linux system administration.

### Prices per month ex. VAT

RAM	Storage	Traffic	Price
64M	3GB	5GB	£15
128M	6GB	10GB	£27
256M	12GB	20GB	£50

## Bytemark

Hosting

☎ 08707 455026  
or order online at

[www.bytemark-hosting.co.uk](http://www.bytemark-hosting.co.uk)



# Mailserver

```
cd ./packagename
./configure
make
make install
```

then run the setup program:

```
hcfpciconfig
```

then look for it under /dev/modem, not /dev/ttyHCF0, using *kppp*. Hope this helps! *Linux Format* is great, Keep it up!

P Nuttall, via email

Thanks for the tip.

## Distros tested?

I have found out what the problem was with X crashing when I upgraded MDK 8.2 to 9.0 from your coverdiscs.

The upgrade had deleted the .xinitrc scripts from my home directories. This caused the default xinitrc to call *Xsession*. *Xsession* had an entry for i18n which fails, and this caused X to crash out. Commenting this out allowed X to start up correctly. However, there are more problems...

X now starts, but I got *IceWM*, and the mouse went berserk. Despite my MS wheel mouse being correctly identified in 8.2, 9.0 thought it was a 3-button mouse and installed the wrong driver in XF86Config-4. I changed this by hand to the correct IMPS/2 driver. As an aside, I ran XFree86 -configure, which then misidentified the mouse.

The reason *IceWM* was starting was because my .xinitrc files were gone, as they contained the **startkde** instruction.

This sort of problem will put less technical and less tenacious persons than myself off. I believe at the end of the day, it is the responsibility of the distro compilers to ensure the product is fit for purpose before release. As far as I am concerned, Mandrake fell down on this one. I now believe Red Hat have probably done the right thing by re-engineering core apps like GNOME and KDE. The only positive thing to have come out of this is that I now have a much better understanding of how XFree86 is configured!

Martin Lawrence, via email

Upgrades from previous installations never seem to go smoothly, and partly it can be as a result of what you might have done with the system in the meantime. Though I think merely

**“Not only is Knoppix a complete install-free distro that can boot from CD but it can also be permanently installed onto a spare HD partition greater than 3GB.”**

having to reconfigure X isn't too bad compared with the problems some people have.

Mostly the update merely updates RPMs, but this can have some undesirable side effects in the rest of your particular Linux OS, occasionally overwriting config files, etc. This isn't really limited to one distro either, or even a full system upgrade. For some reason Red Hat's *Apache* updates always insist on re-installing the index.html page!

## Knoppix kudos

The Knoppix distro that was provided on the Cover disc of your December issue, is much more than is stated in your introductory guide. Not only is it a complete install-free Linux distro that can be booted from a CD but it can also be permanently installed on to a spare partition greater than 3GB on your hard disk. Best of all, you then have a Debian 3.0 installation with nearly all the hard work done for you!

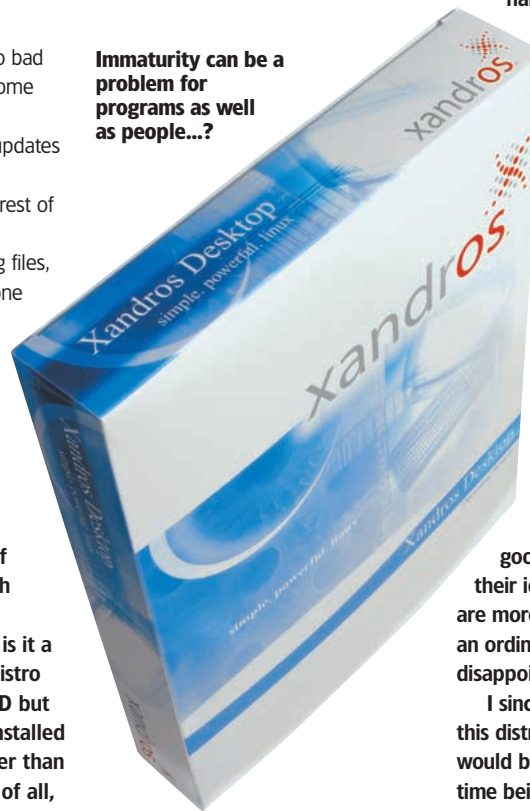
All you have to do is to run the *knx-hdinstall* script (on the CD) but it is helpful to first read the Knoppix Hard Disk Installation HOWTO by John Jubal. Useful websites to find this and other advice are [www.knoppix.net](http://www.knoppix.net) and [www.nzoss.org.nz](http://www.nzoss.org.nz).

The Knoppix CD is also being used a starting point to produce a remastered CD to demonstrate special interest software. This is clearly a development still in progress which interest your readers and could be the subject of an article in *Linux Format*. One must give special credit to Klaus Knopper, creator of Knoppix.

Des Sale, via email

Knoppix is indeed a great piece of software – which is why we put it on the coverdiscs! Thanks for pointing out that it can be used for more than just a run-from-CD Linux distribution.

Immaturity can be a problem for programs as well as people...?



## No Xandros ISDN!

The review of the Xandros distro inspired me to order it immediately.

Everything looked fine until... The distro doesn't support ISDN, but apart from this, there were more problems. I think the Xandros installation and hardware recognition is fabulous and I adore their FileManager; but...

There is a bug as far as network card recognition is concerned. I found the solution at the Xandros forums site. Nevertheless, this is really a major bug – all other distros support ISDN and my ISDN card; I was online in minutes..

I didn't manage to install (under the KDE2 desktop) the US-international keyboard layout with dead keys. This is for a European customer a mortal sin, as most of our languages use some accented characters. The distro still uses KDE2, which is a bit outdated.

As for the question “value-for-money” – the ‘cheap’ distro (\$39) consists of one CD. That would be fine enough – the ‘fully blown’ distro (\$99) adds an extra CD with a non supported version (TechView) with KDE3, and the *Konqueror* filemanager (which I hate!).

I found no way to configure my hardware. The only way was

via the *KDE Control Center* – by no means enough. All other distros provide an additional Control Center of their own to fill the gaps. For an ordinary user this additional CD doesn't make any sense, and I wonder if any professional would be interested in such a immature preview. They add a neat manual, but on balance it seems that I have to pay \$60 for that manual. I am convinced that their intentions are

good, and think that some of their ideas about building a distro are more than worthwhile. But as an ordinary customer I am highly disappointed.

I sincerely hope that one day this distro is mature enough and would be happy to use it. For the time being I'll return to SuSE.

Will JB Hus, Netherlands

There are many useful things about Xandros, and it's really interesting to see how it has progressed from the original Corel core. We always welcome readers thoughts on any distributions they have tried; does anyone have any more opinions on Xandros or any other distros? [LXF](mailto:lxf)

## SUBMISSION ADVICE

### WHAT WE WANT:

- Letters about the magazine or Linux in general
- Constructive criticism
- Your opinions
- Concise points about relevant subjects

### WHAT WE DON'T WANT:

- Technical questions – direct those to our Q&A pages!
- Random abuse
- Nonsense rants
- 200 pages of meandering diatribe

### WRITE TO US AT:

*Linux Format*, Future Publishing, 30 Monmouth Street, Bath BA1 2BW or email: [lxf.letters@futurenet.co.uk](mailto:lxf.letters@futurenet.co.uk)

# Reviews

All the latest software and hardware reviewed and rated by our experts

## LXF verdict explained

Each review is accompanied by a Linux Format Verdict to help you to assess the product at a glance (it's no substitute for actually reading the review, though). We award scores out of ten in the following categories:

**Features:** Does it provide the functions you need? Is it innovative?

**Performance:** How well does it do its job? Is it fast and reliable?

**Ease-of-use:** Is the interface well designed? Is the documentation well written, helpful?

**Documentation / Value for money:** Whichever score is most appropriate!

For those who like numbers, the Linux Format Rating is a score out of 10 summing up the overall excellence of a product. It will usually, but need not be, an average of the above categories. We award scores as follows:



**10** The close to perfect product.



**8-9** Good, but has a few niggles.



**6-7** Does the job, but needs work.



**5-4** Average.



**1-3** An utter disaster. Back to the drawing board.

## The Top Stuff Award

If we really, really like something – we really think that a particular piece of software, hardware or any other sort of ware is the best stuff around – then we'll give it our Top Stuff Award. Only the very best will be chosen. It's not guaranteed to all products that score highly.



## WHAT'S NEW...

### VariCAD 9.0

An Open Source mechanical CAD application for 3D and 2D design featuring interference checking and more **p28**

### SuSE Enterprise

The latest contender for the server distro crown with exemplary documentation and packaging **p34**

### IP\*Works!

An internet protocol library that can save proprietary coders time and trouble, though not for open source projects **p36**

### ZEND Studio 2.6

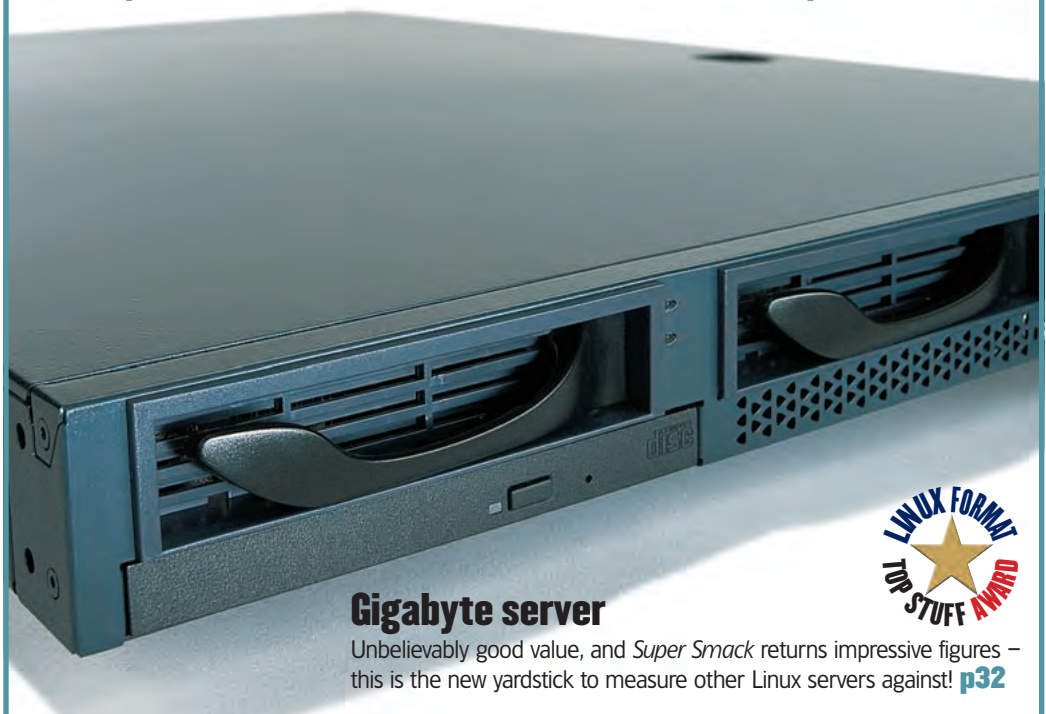
This PHP IDE is more streamlined than earlier iterations and includes a great debugger **p40**

### IEMS

'All-in-one' mail server package offering everything from SMTP to POP and address books in one single distro **p42**

### Books

*Implementing Intrusion Detection Systems*, *Linux For Windows Administrators*, and *DNS For Dummies* **p44**



### Gigabyte server

Unbelievably good value, and *Super Smack* returns impressive figures – this is the new yardstick to measure other Linux servers against! **p32**

## COMING UP SOON...

### Mandrake 9.1

The latest effort from Mandrake, still going strong on the development front in spite of the well-publicised financial worries. The release candidates were popular downloads, will the final release match up to the early promise?

### SuSE 8.2

Not content with several new server offerings, SuSE have upped the ante on their standard distro.

### Mozilla 1.3

The final release version is on the Coverdisc – we should be able to give a proper verdict next issue.

### Red Hat 9

Blimey, the newest Red Hat is due for release any day now, so it looks like the Distro wars are back on, big style!

### Opera 7

Still waiting, but it should be out in a matter of days...



## MECHANICAL CAD

# VariCAD 9.0

**Nick Veitch** has designs on the latest version of this CAD software.

**Mechanical CAD software for 2D and 3D design. Other CAD software worth considering.**

- **DEVELOPER** VariCAD
- **WEB** [www.variacad.com](http://www.variacad.com)
- **PRICE** US\$399

**G**ood quality CAD packages for Linux are still fairly thin on the ground. While there are some specific applications for many niches, and the excellent QCAD for 2D drafting, there are still few professional tools for mechanical CAD which include 3D design and modelling. *VariCAD* is almost in a field of its own, and since the 8.0 release we covered in *LXF26*, now has almost 11 years of continuous development work behind it.

*VariCAD* makes no pretence that it is really dedicated to the arena of

mechanical CAD design – though of course this does still give it some scope in application areas such as architecture or civil engineering, and of course, the 2D drafting mode can be used to draft anything you like – lines are just lines after all. However, all the internal object libraries and tools are very much geared towards engineering rather than any other discipline.

## Installation

*VariCAD* is supplied as precompiled binaries in RPM format. Version 9.0-04 is available for Mandrake 9, SuSE 8.1, Debian 3 and all versions of Red Hat since 7.2. It may well be possible to use one of these RPMs on other distributions of Linux, but these aren't necessarily going to be supported. It's worth noting that upgrades to your system can make a difference to how *VariCAD* operates – for example, we initially tried to run *VariCAD* on a Mandrake 9 installation that had been updated to KDE3.1, but ran into some problems. For the full rundown on the system requirements, see the boxout on the left.

The program installs itself by default to the /opt directory, where it stores libraries, executables and common files. Additional configuration and drawing files will be stored in the individual user directories.

## System requirements

**128MB free RAM**  
**30MB disk space**  
**KDE/Qt libraries**  
**Supported distributions: Red Hat 7.2, 7.3, 8.0; Mandrake 8.1, 8.2, 9.0; SuSE 7.3, 8.0, 8.1; Debian 3.0**  
**OpenGL-capable card recommended**

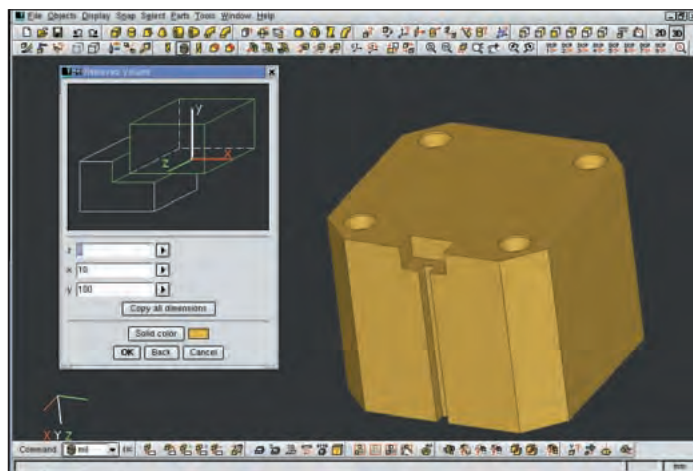


Interface-wise, *VariCAD* makes use of the KDE libraries and Qt. It uses the standard KDE file requestors and print setup, which makes things a little easier.

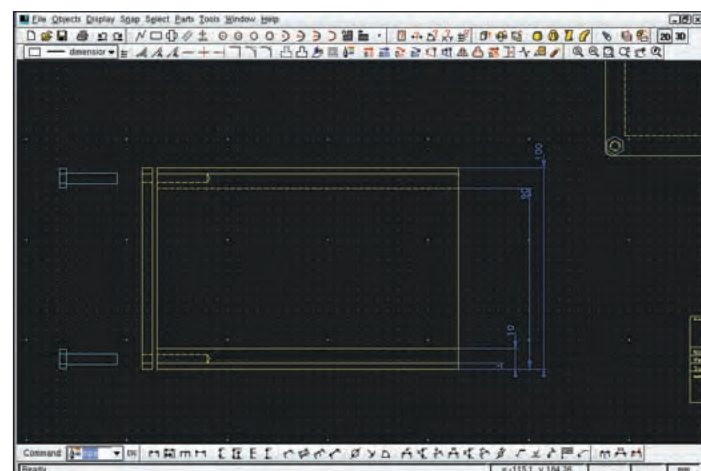
## 2D drafting

*VariCAD*'s tools for two-dimensional drafting are pretty much what you would expect. A layered drawing environment enables you to create different objects on up to 250 layers to make complicated drawings easier to manage. An automatic layering feature will put created objects in an appropriate layer – for example, all hatching is placed in a layer called 'hatches'; dimension lines in a layer called 'dimensions' and so on. This

To start *VariCAD*, you may wish to set up a shortcut or menu entry. A menu entry is created by the install process, but only for the root user. If you start from a shell, you need to make sure you are in your home directory. *VariCAD* seems to use a relative path from where it was called to locate your drawing files, so to avoid error messages it is probably best to create a consistent shortcut on the desktop or in a menu.



Designing even complicated 3D shapes is relatively straightforward with the tools provided by *VariCAD*.



Plenty of shortcuts, automatic layers and intelligent dimensioning make 2D design a breeze.

## Features at a glance

- 2D/3D design
- Assembly drawing
- Interference checking
- Part/symbol libraries
- DWG/DXF/IGES support
- Bill of materials
- Plot/print support
- Calculation tools
- OpenGL rendering

makes it very easy to show or obscure only the detail you need to at a particular time. Notes and other useful text could be added that aren't intended for final printout for example.

A range of drawing tools covers most of the shapes you are likely to need, and often several different ways of constructing them too. Circles can be generated from two points, a centre and diameter, or the intersect of three points. Tools for chamfering or fillet radiusing make short work of pointy corners.

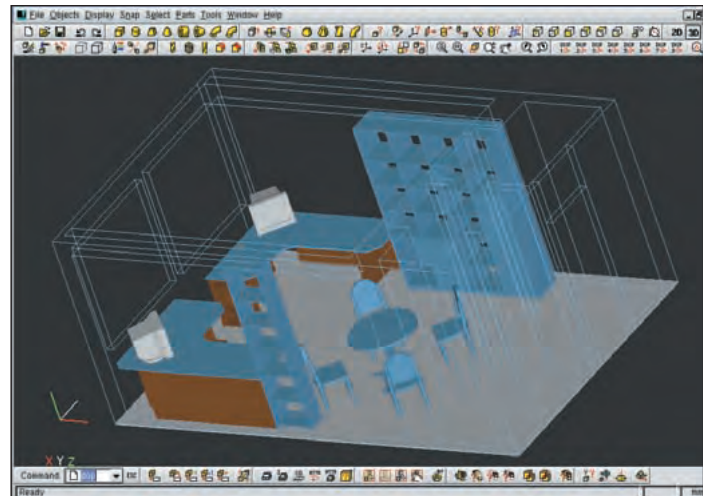
Dimensioning is pretty easy too, with a selection of tools to cater for types of dimension (offsets, radius, diameter, angles, horizontals) and catering for different orientations. *VariCAD* uses a selection of built-in line based fonts for dimensioning, including a cyrillic font and a few italic versions. The range of styles here isn't great,

and although there are mechanisms for changing the width/height ratio, these can be a bit fiddly to use – still much easier than lettering it all by hand though!

## Working in 3D

Obviously, traditional design has grown up from a 2D orthographic environment – it's difficult to model in 3D with just a sharp pencil and a piece of paper. While a 3D mode isn't therefore essential, it is certainly very useful. For simple objects, designing in 3D can be a lot faster – instead of having to draw construction lines and add eight or twelve lines to define a cube of material, you can do it simply by entering the three dimensions. Complex parts may seem trickier at first, though the design system goes some way to ease this. As in the 2D mode, you can snap to positions and align objects relative to other key points in the design.

3D objects can also be generated from your 2D drawings, by extruding



Although best at mechanical design, *VariCAD* can also be used for other design work too, as this example shows.

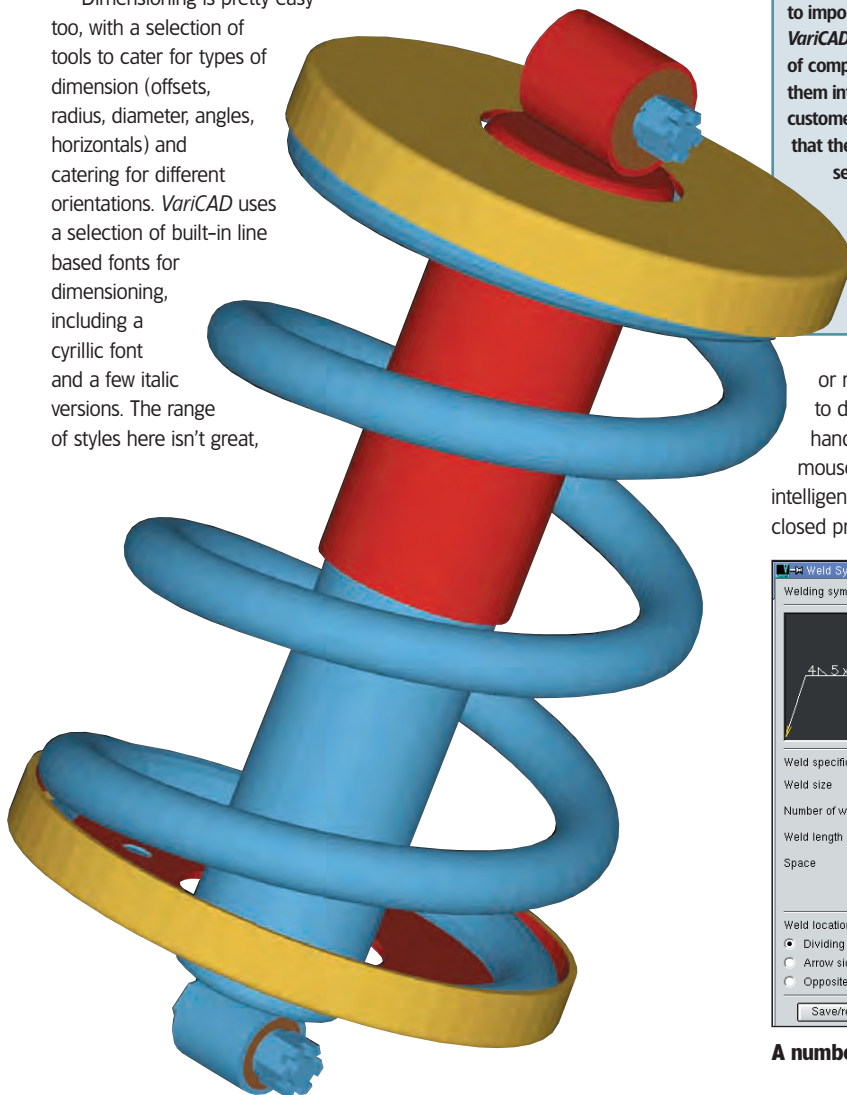
## Supported file formats

Upgrade from other systems is possible

If you are upgrading from a different CAD system, or wish to take advantage of existing drawings from elsewhere, you are going to be interested in the ability to import and export drawings from *VariCAD*. Suppliers often have drawings of components to help you integrate them into designs, or clients or customers may have required formats that they will accept. *AutoCAD* still seems to be the *de facto* standard for CAD systems, at least as far as file formats, so *VariCAD* has a robust import and export module for dealing with .dwg and .dxf files.

The newer, but popular IGES format is also supported, but alas, only for 2D drawings – trying to import a 3D IGES file will result in an error or an unholy mess of lines. There has been a lot of hard work done on the ability to export IGES files in 3D though, and these now seem to work in the variety of software we tested them with.

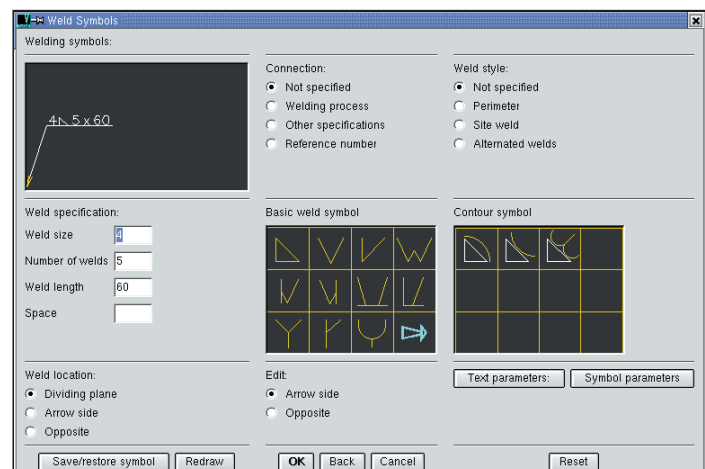
In use though, there still seem to be some glitches in importing 2D IGES files, so if this is an important feature for you, you may like to check out the trial version of the software before purchase to check that it can cope with your requirements.



or revolving these cross-sections to define a volume. This is all handled with a few clicks of the mouse – the software being pretty intelligent when it comes to identifying closed profiles. Complicated shapes

which have some wholly internal lines will cause some problems though.

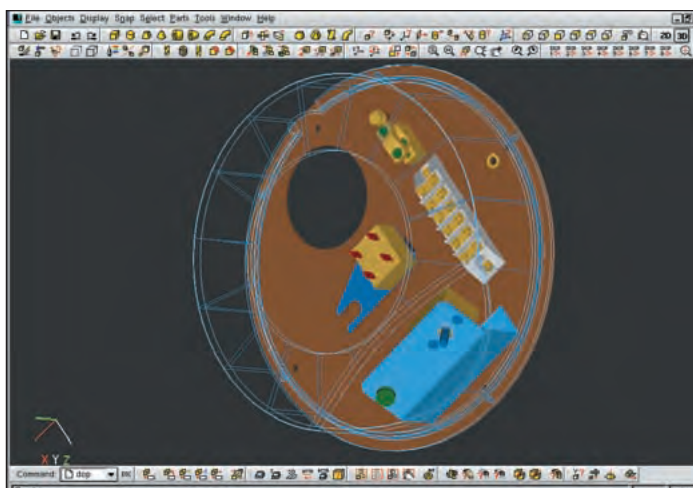
The really good news is that this system works both ways. As with leading packages for Windows, you can export the 3D view as a 2D



A number of predefined symbols are provided for easy markup.



# ReviewsVariCAD



Elements of the 3D view can be hidden, or switched to wireframe mode when viewing, to make obscured parts more obvious.

« drawing. This can make for incredible time savings and is certainly very useful for prototyping. As the 2D outline is linked to the 3D object, it can be updated as changes are made to the solid version.

One of the useful features of the 3D design mode is the interference check. This will detail any components in your design which co-exist in the same 3D space – obviously a physical impossibility in the real world. This is of vital importance when your are dealing with complicated assemblies – the parts themselves may seem right, but if they don't actually fit together, manufacturing will be a bit of a problem.

This comes into its own when dealing with assembly drawings.

Components in separate drawing files can be linked together into an assembly drawing. Changes to the original drawings will be updated in the assembly, so different people can easily be working on different components of the same overall system. Using other features such as solid groups, this makes for a highly productive environment for modelling complex parts.

There is also limited support for surface development – ie 3D objects created from bending sheet materials. Simple 3D objects can quickly and easily be turned into 2D surface plans simply by selecting the surfaces that need to be developed, along with how they behave in relation to neighbouring surfaces.

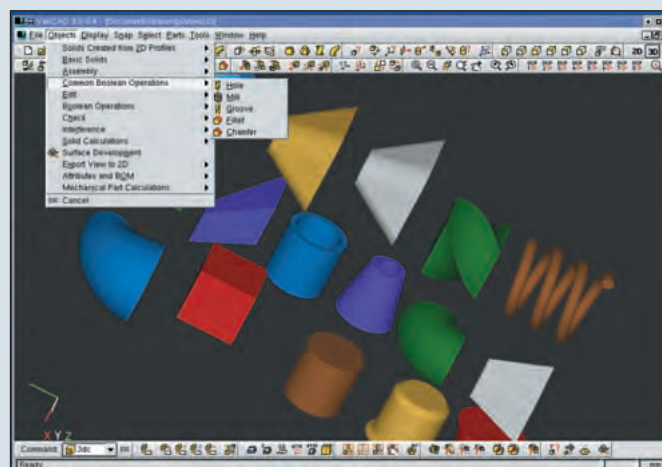
## OpenGL

Not strictly needed – but recommended

For 3D work, the *VariCAD* software makes extensive use of OpenGL if available. You will need to have the correct software already installed for this, but these days this is pretty straightforward for supported graphics cards than it used to be.

You can still work in 3D without an OpenGL card, but software rendering isn't really up to the task. Restricting yourself to wireframe views also makes complicated design next to impossible. For this reason we wouldn't recommend you designing in 3D without an OpenGL capable graphics system.

Holding down both SHIFT and CTRL allows you to rotate the 3D view smoothly, which is a great boon for seeing in intricate detail, or for seeking the exact angle to zoom in on. Various OpenGL functions can be set to refine the display, including whether to highlight the edges of faces or smooth them, which helps customise your view for different purposes. Specific viewing angles can also be saved to one of eight presets, making it much easier to zoom around your object, or to remember those useful angles for when you want to export 2D profiles.



OpenGL rendering makes the 3D view realistic and – more importantly for impatient engineers – very fast.

While overall *VariCAD* is obviously a good solid performer for CAD work, the real selling points of the software

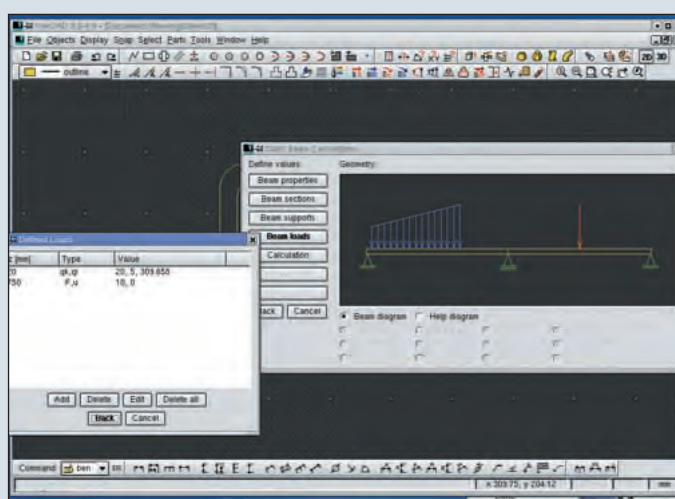
are all the additional features added to make a mechanical engineer's life that much easier. The scope of some of the tools may be limited, but the time saved by features like the component libraries, the convenience of the calculation tools and the flexible way of working with linked files all adds up to a very efficient environment. If you are interested merely in 2D drafting, you might want to look at the GPLed QCAD, but for mechanical CAD, there isn't anything to touch this on Linux. [LXF](http://www.linuxformat.co.uk)

## Useful tools

User-definable nuts and bolts!

As well as providing tools to draw objects, *VariCAD* can help you actually design them too. A number of very useful tools are available from the Objects>Calculations menu. These deal with a variety of design elements, from choosing the right springs and bearings to modelling loads on shafts and beams. Enter a few salient values and the tricky sums are done by the software, and a list of recommended parts displayed. Many of these can be drawn by the software itself, and you can save out the calculations to a text file.

As well as these design helpers, standard measurement tools exist for measuring distances, areas and volumes, up to calculating the moment of inertia of a solid – though you will often have to supply some necessary extra information (like the material used).



Designing shafts and beams becomes a lot easier when the software does the monkey-work for you.

## VERDICT

Features	8/10
Performance	9/10
Ease of use	8/10
Value for money	9/10

Certainly the best mechanical CAD solution for Linux, and at a great price.

**LINUX FORMAT RATING**  
**9/10**

BARGAIN SERVER

# Gigabyte server SR113



Interested in getting a great server and still get change from a grand? **Paul Hudson** investigates the new kid on the block...

Powerful 1U server with more than meets the chequebook. Also consider the less expensive GS-SR101(T).

## GIGABYTE SR113 1U RACKMOUNT SERVER

■ **SUPPLIER** Upgrade Distribution

■ **WEB** [www.upgrade-distribution.com](http://www.upgrade-distribution.com)

■ **PRICE** £492/£872 (exc. VAT)

■ **PHONE** 01252 332800

**T**he Gigabyte SR113 is first and foremost an entry-level server option targeted at businesses looking to get the most bang for their buck. Our review model came configured with a 2.4GHz P4, 1GB of RAM, three 30GB Maxtor hard drives that were connected through a Promise ATA-133 RAID card, and still managed to come in at £872 exc. VAT.

Given that you'd be hard-pressed to find a *desktop* of that calibre for such a low price, I imagine you're expecting to read that the SR113 is lacking in the kind of luxuries that administrators are used to: dual ethernet interfaces, hot-swappable hard drives, remote management facilities, and, most importantly, solid Linux support.

However, you couldn't be further from the truth. The SR113 comes with two Intel 82550 Ethernet interfaces running at 10/100Mbps, which are pretty much as good as you'll get in the class, and have a solid history of reliability and performance. As for the hard drives, it comes with three bays: the first will take an ATA-100 drive for your OS, and the other two are hot-swappable, taking ATA-133 drives for maximum performance.

As for remote management, much of this work is done by the standard tools we take for granted nowadays, like *ssh*. However, the SR113 does have a Winbond W83910F controller to handle its implementation of the Intelligent Platform Management Interface (IPMI). This specification, in use by Dell, HP, NEC, Intel, and others, allows the monitoring of the "health"

of supported hardware (temperature, voltages, etc) in order that managers are able to track the exact status of any computer remotely. Note, though, that the SR113 supports v1 of the IPMI specification, and not v1.5, which includes more sophisticated alerting functionality.

With regards to the level of Linux support, the SR113 came pre-installed with Red Hat 7.3 (Valhalla). Once it had finished booting – which didn't take long at all thanks to the high-end CPU powering this machine – the system had connected to our network and was ready for action. Each piece of hardware inside the box worked first time and just as one would expect it to, which reflects very well on the build quality of the supplier. You get the impression that Upgrade Distribution, who supplied the hard drives, the

memory, and the CPU, have a very stringent quality control system.

The Award BIOS that is installed is basic, but adequate for a server – don't expect many opportunities to tweak performance. Although, given that a popular maxim in the hi-fi world is "The fewer buttons an amplifier has, the better", perhaps it's best that 'have-a-go BIOS tweekers' are given less leeway.

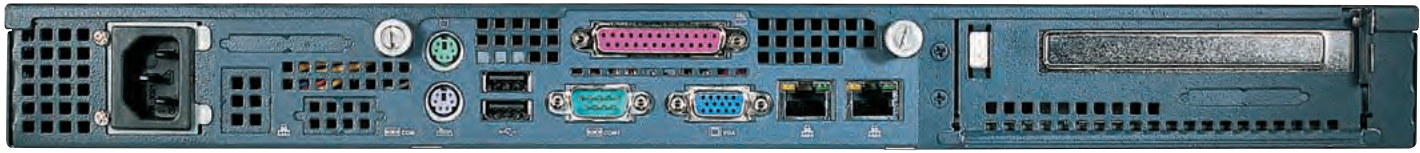
## Benchmarking the beast

Wanting to *really* know whether this thing was fully Linux compatible, I replaced the default RedHat install with Debian, half-expecting at least a few installation hassles. No such luck! With branded, well-supported hardware being used throughout this box, installing Debian was as smooth as anyone could ask for. If hardware is supported by software as old as that which comes with Debian, it gets a thumbs up from me!

**The front of the SR113 features space for three easily removed drives, two USB ports and also a serial port. There's more around the back!**







After installing various popular services (*Apache*, *MySQL*, *sshd*, *ftpd*), the server was ready for testing. Debian 3.0 was used with the default *idepci* kernel, an ext2 file system with go-

faster stripes painted on, and the tests were run as a normal user.

The first test was *hdparm*, the popular hard drive tweaking tool. If you want to reproduce the test at home, use the command

**hdparm -tT <your hard drive here>**. The SR113 read 128MBs from hard drive cache in under a third of a second, which is the equivalent of 412.90MB/sec, and read 64MBs from the primary drive itself in 1.28 seconds, or the equivalent of 50.39MB/sec. If I were reviewing a £5000 server, I'd describe numbers like these as excellent, but for a server in the SR113's price range, performance like this is unheard of and *most* welcome. NOTE that using **hdparm** to change settings on your own PC is potentially dangerous and you should consult the manual first.

Moving on to less theoretical tests, I installed *ApacheBench*, the *Apache* benchmarking suite. This is an excellent tool for measuring sheer web server performance and capacity, and it works simply by requesting creating a number of web connections, each requesting a given page from a server you specify until a set number of requests have been completed.

I ran about 50 *ab* tests: half with 50 clients connecting for a total of 1000 requests, and half with 1 keep-alive client connecting for a total of 10000 requests. The latter generally indicates network throughput for the server, and the former indicates network connectivity and CPU performance. Keep-alive works by sending multiple pages through one HTTP connection, so the latter test will naturally be a great deal faster than the former because it does not need to negotiate a connect and disconnect

each time. The content to transfer was a 4kb HTML file with no pictures or other attached content, and required no server side processing.

Again, it was a pleasant surprise to find the SR113 continuing its trend towards superior performance: 1,000 requests over 50 clients had *Apache* handling an incredible 3,789 requests a second. Furthermore, 10,000 clients on one client returned 7194 requests a second, which is the highest score I've ever seen *ab* return. If you're unable to run this test at home, things will become relative to you when I mention that the Mandrake box I'm writing this article on, an 800MHz PIII, scores just 1,089 requests a second for the second test – over six times slower.

Realising I really had to push this thing to the limit in order to make it struggle, I compiled and installed the *MySQL Super Smack* benchmark, the definitive way to make smoke come from your computer. *Super Smack* works in much the same way as *ab*: it connects to your server a specified number of times, then performs a large number of queries on each connection.

For this test, I set up *Super Smack* to connect fifty clients, and perform 100,000 iterations of the query set on each client. The query set was simply "update database with new random information, then select that information back out from the database". Testing a large number of **select** statements will result in the dataset merely being read from memory; writing first forces a data commit before the next **select** statement, thus ensuring that each query set performs a write and a read operation. Furthermore, due to the concurrent nature of having 50 clients working at the same time, *MySQL* has to lock the database as it writes new data, then unlock it once the write is complete, which forces read operations to queue up while the write is taking place.

Now, imagine that being done 100,000 times by 50 clients, totalling 5,000,000 query set iterations, and 10,000,000 total queries – I think you will agree that this pushes I/O and CPU load to the extreme!

**A very well-designed bit of kit, although perhaps not if you're colourblind! From left to right: Kettle lead port, mouse and keyboard, two more USB ports, a menagerie of serial connections and two network ports. Lovely!**

The end result? The SR113 handled 2,437 queries a second and took about an hour in total to run through the entire test. If you've never tried it yourself, it's probably hard to understand how much this particular test beats up a computer. There really is very little out there that approaches the processing requirements of *Super Smack*, which means that under the heaviest of loads, the SR113 will still be able to handle almost 2,500 SQL queries every second. Incredible.

## Beyond benchmarks

Beyond the benchmarks, the machine continued to perform admirably: "snappy" is the most fitting word I can think of as it juggles a variety of complex processes without the slightest slowdown. I must admit, it must be a little disheartening for users of more expensive servers to find that, even with a large collection of services I picked out from the APT database running and performing tasks, the SR113 doesn't seem to be affected in the least.

The machine itself is a 1U system, and as such is designed to be in a server rack, presumably in a server room. That's a shame really, because the unit has been designed with a great deal of common sense. The hot-swap hard drives slide out smoothly with a small tug, there are two USB ports on the front for easy access (with two more on the back), and even an added serial port for console redirection.

With regards to the graphics chip that comes bundled, the SR113 is provided with an 8MB ATI Rage XL card, which is more than enough for console use of all types. It also comes with a 24x CD-ROM and floppy drive as standard.

You will be forgiven for wondering at this point whether there are any down-sides to the SR113, but there is one (albeit minor) problem with the kit. The supplied documentation is a little flimsy, coming in at just 25 pages. Now, I realise that the SR113 is hardly a piece of kit that inexperienced users

are expected to deal with, however it would be good for the documentation to at least include a short troubleshooting guide, and perhaps even some support information if things go wrong.

## Conclusion

Apart from that one minor docs niggle, and it really is minor when you consider that system administrators are generally clued up about server management, the SR113 is a shining example of how servers should be done, low-cost or otherwise.

As mentioned earlier, our model was a modified version of the standard SR113 and costs £872 excluding VAT. The basic model is a bare-bones system and doesn't come with much of the hardware, which explains its even lower price tag of £492 before VAT.

£872 must be the lowest price around for the level of performance shown here, and I am surprised that such a low price point is being used for a machine of such capability. On the other hand, as a consumer, I'm more than happy to take equipment like this off their hands for this money.

In the end, I am much impressed by the SR113, and would even go so far as to say I now consider it the yardstick for Linux servers in the future. Indeed, when the time comes that I need to replace my own home server, I doubt I shall have look any further than this capable and very cost-effective little box! **LXF**

## VERDICT

Features	9/10
Performance	10/10
Ease of use	9/10
Value for money	10/10

With a better set of printed docs, this would have full marks. It still gets our coveted *Top Stuff* award though!

**LINUX FORMAT RATING**  
**9/10**

## ENTERPRISE LINUX DISTRIBUTION

# SuSE Linux Enterprise Server 8

Paul Hudson takes a look at the newest contender for the server distro crown.

**UnitedLinux-based distro. Also consider Red Hat, Debian, and other popular distros.**

- **VERSION** Enterprise Server 8 for x86
- **DEVELOPER** SuSE/UnitedLinux
- **SUPPLIER** SuSE
- **WEB** [www.suse.de/en](http://www.suse.de/en)
- **PRICE** £899+VAT
- **PHONE** 020 8846 3918

**W**hen the Editor dropped the review copy of SuSE Linux Enterprise Server 8 (SLES8) box on my desk late on a Wednesday evening, I was immediately impressed: whoever designs SuSE's packaging deserves a gold star, because the box itself is an enticing-looking translucent wrap that also makes up the binder for the manual and CDs inside.

As the first UnitedLinux distribution from SuSE, if not the first from anyone, I had particularly high expectations: SuSE are known in the Linux world for creating well-rounded distributions, and, with the standard UnitedLinux 2.4 kernel having a large collection of back-ports from the 2.5 kernel (including such delights as a new scheduler, improved memory management, and access control lists), I was expecting a considerable performance increase overall.

The review copy we were sent was SLES8 for x86 and AMD Hammer-based systems (x86-64), but SLES 8 is also available for Itanium, IBM pSeries/iSeries/zSeries and also the IBM S/390.

## First thoughts

Cracking the box open, I discovered that it housed almost 350 pages of well-designed documentation split into logical sections with thumb tabs for easy access, as well as four CDs. Interestingly enough, the first CD was sealed with a sticker, "By breaking the seal you accept the exclusion of

warranty". This is a smart move by SuSE; naturally it cannot warrant all of the software enclosed with a Linux distribution, and, as the distribution is clearly aimed at businesses, this sticker acts as a good reminder. Certainly it's better than a sticker which reads like "By breaking this seal you are accepting the terms of your End User Licence Agreement!"

## Documentation

Being a fan of good documentation, that was the first area I looked at. Section three of the manual covers installation in exceptional depth, with over a hundred pages being devoted to the topic. Section four forms the core of the documentation, with over 320 pages devoted to the task of system administration. This is another smart move by SuSE; if companies are to switch from Linux to Windows, it mustn't be assumed that their administrators will somehow become Linux gurus immediately.

So, to ensure administrators are able to find their answers without fuss, the maintenance section of the manual covers fairly basic topics such as configuring routing, disk partitioning, and setting up OpenGL, but it also goes on to discuss internationalisation, configuring BIND, and transparent proxy configuration. This is server documentation at its finest, and, even though I had yet to put the SuSE CD in my drive, I was

already very impressed with SLES8.

## Installation

Putting the documentation to one side, I broke the seal on CD 1 and popped it into the CD ROM of the Gigabyte SR113 server reviewed on page 28 of this issue of *Linux Format*. The reasoning behind this is that if I was going to test an enterprise-level piece of software, I figured it would be best to test it on an enterprise-level piece of hardware!

The CD booted up and, after a short loading process, launched *YaST* (Yet Another Setup Tool, SuSE's setup and configuration program). *YaST* is certainly the most attractive installation manager out there, and has lots of support for non-English languages.

Having detected the SR113's existing Debian install, *YaST* also chose what it considered to be the best partition layout, which used *ReiserFS* by default, which is good to see. The default selection for packages was somewhat odd: GNOME, KDE, LDAP, DHCP, DNS, NFS, SAMBA, CUPS, and others. I shouldn't imagine most



Administering your SLES server is done from one central location.



server administrators feel the need to have multiple window managers installed! Particularly unusual choices were the inclusion of the *Glade* GUI design environment (see page 72), and the *exclusion* of *Lynx*.

Even though I was going to install with the default packages, I took a quick sneaky peeky at how *SLES8* manages its package selection. Four options are given: Minimum System, Minimum Graphical System, Default System for UnitedLinux, and Default System. Alternatively, you can select 'Detailed Selection' to choose each piece of software you wish installed. However, YaST actually crashed when I clicked 'Cancel' on the Advanced Package Selection dialog, trying return to the main installer. Not encouraged, I rebooted, and kept the default packages.

Sadly, the default install failed in under five seconds because, it seems, SuSE had opted to install onto the RAID array, which apparently is not supported too well during installation. The Gigabyte SR113 comes with a hard drive bay specifically for the operating system (see review), so I reconfigured the partition selection in order to make use of this drive, and tried again.

Although I wonder how well new Linux administrators would cope with problems similar to the above, it's important to note that the SuSE manual does cover disk partitioning during install, and that may well be enough for most needs.

Continuing on, YaST copied files from each of the CDs to the hard drive as appropriate. There was a little

bit of confusion with regards to the disc number – they are numbered Installation, UnitedLinux 1/3, 2/3, and 3/3, but each disc has a large number printed on them from 1 to 4, which results in UnitedLinux disc 2/3 being numbered 3, etc, which might bamboozle some users.

After adding initial users, YaST gave me the option to choose between text mode only or a graphical desktop environment (XFree86). It had autodetected the ATI card in the SR113 and had configured it with the correct settings for me, which was promising.

The final installation stage for YaST was to detect hardware like network adapters, printers, etc. It found the network interfaces no problem, and autoconfigured the first Intel adapter with DHCP without any interference from me. However, to *change* the YaST-recommended network configuration was problematic, and I ended up giving up, because YaST appeared to be ignoring what I was telling it I wanted.

## Up and running

YaST then finished the installation, and launched me into X-Windows.

KDE 3.0 was pre-installed, with the Keramik theme from KDE 3.1 and also an attractive SuSE window decoration. Also, there's a clever menu item on the K menu, 'Start New Session', that allows you to have multiple X servers open on F7, F8, F9, etc. This feature, not available in other distros I've seen, is very cool, and I should imagine it will catch on quickly because it's useful to be able to cycle through X servers

freely. Potentially, I could be logged in on F7 in KDE, and my wife could be logged in on F8 with GNOME. Great.

Also installed as part of the default packages was GNOME 2.2, Mozilla 1.01, Apache 1.3.26, MySQL 3.23.52, PostgreSQL 7.2, *sshd*, Perl 5.8, and, peculiarly, *Glade*. Also, GNOME was not really configured as well as KDE. For example, under the SuSE Menu was 'BJ=Control Center' as a link the KDE Control Centre. Including such an old version of Mozilla is a shame, but indicative of *SLES8*'s long release cycle – indeed, it's unlikely the next version of *SLES* will be released in the next 15 months or so.

In hindsight, I'm not too sure why the command-line web browser *Lynx* didn't make it into the default install, and it's a shame; old favourites such as this should, in my opinion, always be kept around for users that like them.

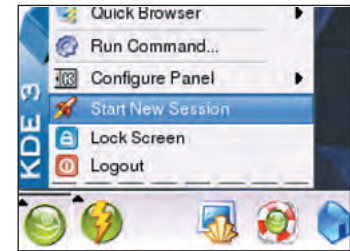
Owing to the 12-18 month release cycle of *SLES*, SuSE supplies service pack updates in order that users are kept updated with the latest patches and fixes. Also, SuSE offer an all-inclusive maintenance contract to help fix problems and supply upgrades as required – the price of the first year's maintenance is included in the sales price, and is £750+VAT a year afterwards.

The maintenance contract is almost certainly what will entice business users, and that (in combination with the excellent documentation) makes it pretty much the unique selling point for the product). Furthermore, SuSE guarantee support for all *SLES8* platforms for a minimum of five years, again in order to cater for enterprise-level customers.

## Problems in paradise?

Even though this is *SuSE Linux Enterprise Server* and not *Enterprise Desktop*, I was still surprised to find certain packages apparently missing from the CDs – particularly a choice of office suite. While it may be the case that there was a problem with YaST, I could not find any office product available to install.

On the server side of things, however, the default package list is very comprehensive, perhaps even too comprehensive. For example, from the 'web server' category of packages, the default list installs *Apache*, *Jakarta*, *Java*, *mod\_php*, *mod\_perl*, *mod\_python*, *MySQL*, and



**Multiple X sessions? We love them!**

PostgreSQL. I recommend that administrators installing *SLES8* on their own machines customise the packages destined for installation, because, if nothing else, installing a default of two database servers merely doubles the potential for security problems.

As I was testing, I noticed at least two further problems with the installed packages. Firstly, running **apachectl status** calls *Lynx*, which is of course not installed, and so spits out an error. Secondly connecting to the server via *ssh* then running *yast* results in *yast* segfaulting, which makes remote administration difficult if not impossible for some. However, I should at least be grateful that the default installation included and configured *sshd*.

## In conclusion

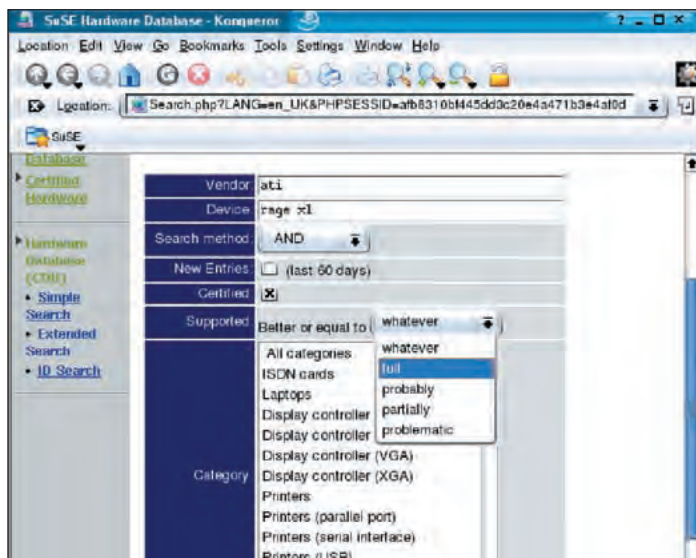
*SLES8* is a solid step forward in server software. The documentation particularly stands out amongst the crowd as how things should be done, but this is dragged down somewhat by a flaky installer and a less-than-perfect choice of packages. However, on the whole *SLES8* is an enjoyable product to use – it autodetected all my hardware and got all the important services working first time. Combine that with the super-cutting-edge kernel being used, as well as the fact that quite frankly it's excellent value for money, and SuSE may well be onto a winner – if they could just correct the minor niggles listed here. **LXF**

## VERDICT

Features	8/10
Performance	8/10
Ease of use	8/10
Documentation	10/10

Good all-round product for a great price; some small but significant niggles pull it back from greatness, though.

**LINUX FORMAT RATING**  
**8/10**



The SuSE online support is remarkably good, and well worth a look.

LIBRARY COLLECTION

# IP\*Works! C++ Edition for Linux v5.0

**Maurice R Kelly** finds that libraries save coders large amounts of time and hassle, but can lead to licensing confusion for the Open Source world...

**Off-the-shelf cross-platform collection of lightweight library programs for use in networking tasks.**

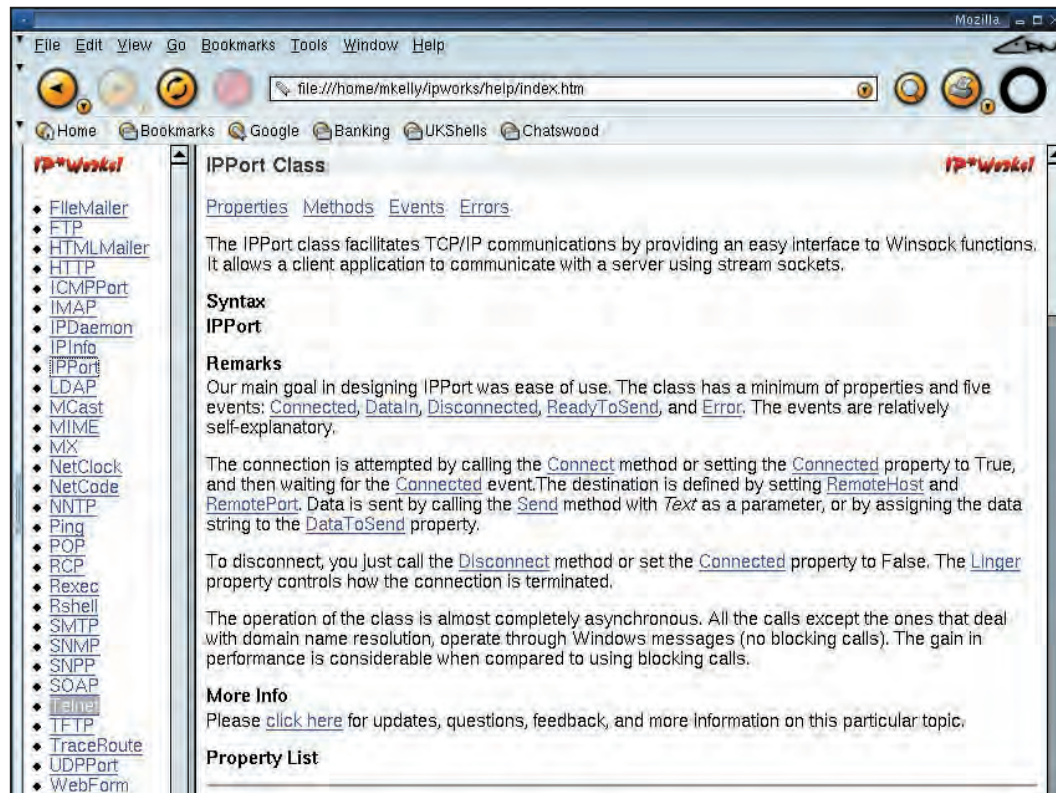
- **DEVELOPER** /n Software
- **PRICE** \$545 (approx. £340)
- **WEB** [www.nsoftware.com](http://www.nsoftware.com)

**T**he program *IP\*Works!* is a cross-platform library for writing Internet client applications. The version being reviewed here is the C++ Linux edition which is just one of approximately 20 platforms supported by the suite. /n Software have been producing *IP\*Works!* since 1995 and boasts an impressive list of companies already using its software.

So why bother with such a library? The answer is one of the key words in software development: re-usability. Every Linux system is more than just a collection of binary programs – it is also a massive collection of libraries as well. Many of these libraries are shared so that application writers can carry on with the specific task of their application rather than having to worry about re-inventing the wheel. The developer effort that can be saved in using an off-the-shelf library can often more than save the cost of buying the package in the first place.

## Let's Get It On

The version of the program tested for this review was downloaded from the /n Software website and came as a zip file, which, when expanded contained a read-me and a JAR file. For the uninitiated, a JAR file is a Java ARchive, which presented something of an installation hurdle for yours truly. The required commands for running the installer were clearly specified in the read-me, but it was quite obvious that a Java Runtime Environment (JRE) was needed.



**The *IP\*Works* documentation isn't particularly fancy, but it contains much in the way of useful information, presented in a logical and straightforward way.**

While not a very tricky procedure, I felt it was a bit much to be expecting a C++ developer to have to install Java to get this product installed. Closer inspection of the JAR file revealed a zip file contained therein which actually contained all the installable files. Once I had Java on my system the *IP\*Works!* installation itself was quick and painless, although the use of Java was probably overkill in this situation, considering it required a 20+MB download from [www.javasoft.com](http://www.javasoft.com).

The installation location yielded a further 4 subdirectories – the actual shared libraries for the application (both in dynamic and static forms), a directory of include files, a collection

of help files, and a series of basic example applications. Also, the installer presents the option to place copies of the libraries into your system library directory (which makes it easier to compile the example files.) One thing you don't get though is source, but more on that later.

## Documentation

While it is tempting to get stuck straight into development with *IP\*Works!*, it is worthwhile having a read at the class documentation supplied. It is presented in an HTML format and, with the exception of the use of frames, is basic enough to be readable in the likes of Lynx through a terminal window. This may seem like a

bit of a moot point, but makes it easier when *IP\*Works!* is installed on a development server rather than a desktop machine.

The individual classes comprising *IP\*Works!* are documented in their own sections, which are roughly broken down into member properties, methods, events and trappable error conditions. The documentation is of a reasonably high standard with detailed information on all properties and methods, though it would be advisable to take a quick look at the `ipworks.h` file in the include directory so that you know what the typedefs are when referred to in the documentation. There are plenty of hyperlinked cross-references, and



interestingly there is a link on most pages back to further information on the /n software site. The links contain a UNIX timestamp which I presume will be used to present updates to the user depending on the age of their build of *IP\*Works!*. Unfortunately I never got to see it in action, but was a nice forwarding-thinking touch.

The documentation isn't spectacular but is certainly handy to have around (and a printed manual is optionally available with the boxed editions.) Unfortunately the same cannot really be said of the example files. They are extremely basic demonstrations of some aspects of the classes, and disappointingly they do not come particularly well commented. Maybe this is asking too much, but as helpful as good class documentation is, I feel that it's just as important to have well commented practical examples in support. A beginners tutorial would also have been a nice addition.

## Getting Stuck In

Enough of pretending that we're sensible programmers who thoroughly read all the documentation before embarking on any project – it's time to dig into the code and see exactly how easy this toolkit makes programming. Thankfully getting started is not a particularly time-consuming task – it literally took no time at all to write a very basic HTTP client to fetch a URL and store it on the local filesystem (which is also the basis of one of the example files.)

That's one of the good things about *IP\*Works!* – the programming API isn't overly complicated, which makes it very easy to just go ahead and get started. Each component is a self-contained class although they share many property and method names (which again, makes it easier to remember what you're doing when you switch between components). The asynchronous events are handled by a series of virtual methods, over-rideable in your derived classes, which can be used simply to keep abreast of what is happening or to retrieve data on the event. A full set of trappable errors is also provided with each component, making for an extremely clean and well thought-out interface – considering that *IP\*Works!* has been around for such a long time this really isn't surprising.

The range of components – a full list is available from the website, but

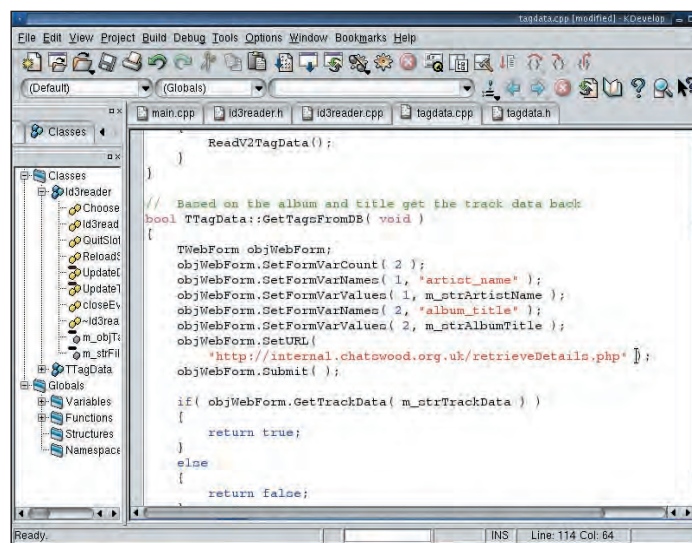
they manage to cover just about all the TCP/IP protocols you are likely to want to deal with. As well as these higher levels functions, there are the IPPort and IPDaemon classes, which can be used to create your own IP clients and servers (the documentation indicates that the IPPort component is actually the basis for many of the other client components). Additionally there is a SAX2-compliant XML parser that is also utilised within the SOAP client. Many of the classes have additional properties for negotiating firewalls as part of the standard set of transactions.

Once written, I had no problems with running any code based on *IP\*Works!*. The library is respectably compact considering what it does and applications were a decent size in memory. After running some of the applications both over time and for tens of thousands of transactions, I observed nothing in the way of memory leaks and was generally impressed with performance.

## Are You Licensed?

Licensing is always a contentious issue when it comes to Linux, and could really be a sticking point for the Linux edition of *IP\*Works!*. The license agreement supplied with the installer basically permits you to redistribute the shared library with your own software distributions. This isn't a problem for commercial software authors, who can simply throw the libipworks.so file in with their own compiled binaries. There is no additional cost based on the number of runtime copies you distribute of the *IP\*Works!* library, so your single development license is enough to support the distribution of every copy of your own application.

However, if you're an Open Source developer, things aren't going to be quite so easy. You don't get the source to *IP\*Works!*, and even if you did, the chances of /n Software allowing you to redistribute their source code is slim. So, the long and the short of it is this – if you're an Open Source developer, who wants to base an application on *IP\*Works!*, you may have to go closed source, or else give up on *IP\*Works!* for the present. I did contact /n Software about this and was told that they were considering a more open edition of *IP\*Works!* for Linux in the future. Unfortunately, one



Using the Webform class to retrieve data from an online database as part of an ID3 tagging program.

of the down-sides to *IP\*Works!* is the pricing system used. It does not come cheaply, and at \$545 for a single developer license it isn't likely to set the independent developer world on fire. It's much more likely that

*IP\*Works!* is going to be utilised by larger corporations attempting to develop in-house applications, or other software companies utilising *IP\*Works!* as a third party library for their own commercial software.

## What do you get?

Your \$545 gets you standard email-based support for licensing and installation issues, minor software updates from the /n Software Website, and access to the online knowledge base. The knowledge base is searchable and there are a few tutorials available within it. While I had no major problems getting on with *IP\*Works!*, I did contact /n Software a few times with questions about the suite, and was most impressed with the speed of their response. For more money you can get a better level of support – premium support (via phone and email) is available for an annual fee of \$495 (renewals for \$395) or on a per-incident basis at \$145 a shot.

## Conclusion

So is this software worth the price? It really all depends on who you are and what you want to achieve from it. If you're a software development manager for a company that writes proprietary applications, then *IP\*Works!* makes a lot of sense. It is

lean, stable, easy-to-use, and comes with the option of support, and could seriously make life easier for your project when it comes to developing your application. The fact that it is available across multiple platforms is a major advantage to organisations that interconnect more than one computing platform.

For the average developer on the street, it is more likely that *IP\*Works!* is a less accessible option. Shareware developers are more likely to find this option to be attractive, as the initial outlay could save them a lot of development time, and the cost could be recouped through revenues from their own applications. Independent Linux developers are more likely to be working on Open Source projects with no real revenue, and so the cost and license restrictions of *IP\*Works!* really aren't going to suit at all. A more open edition of *IP\*Works!* would be more than welcome; we eagerly await this development, as this program would certainly be very handy for Open Sourcers everywhere. **LXF**

## VERDICT

Features	10/10
Performance	9/10
Ease of use	9/10
Value for money	8/10

A great suite of developer components, suitable for commercial, proprietary apps. We await a more open version.

**LINUX FORMAT RATING**  
**9/10**

PHP IDE

# Zend Studio 2.6

**Paul Hudson** takes a dip in the pool of luxury PHP development, as he reviews the latest IDE release from Zend...

**Great PHP IDE with a fantastic debugger improves on previous incarnation. Rivals: *PHPed*, *phpmole*, *Quanta***

- **PRICE** Starting at \$195
- **SUPPLIER** Zend
- **WEB** [www.zend.com](http://www.zend.com)

**B**ack in LXF31 when we reviewed *Zend Studio 2.5* almost a year ago, I can remember being most pleased with it, thinking that PHP development had reached its peak. So it is not without a fair amount of surprise that I found myself sitting in front of the new release of *Zend Studio*, version 2.6, and yet again being impressed by the new functionality that Zend has managed to cram in.

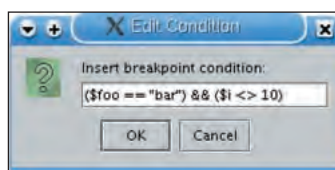
If you didn't read *Linux Format* 31 (backissues available on page 95), here's a quick reminder of *Studio 2.5*.

## Studio 2.5

The following quotes from issue 31 should pretty much sum it up: "Much to offer PHP developers at all levels"; "All in all, the *Zend Studio* outclasses everything else on the market"; "The Zend Development Environment shines as a PHP IDE"; and "Everything you could need from a PHP debugger". See what I mean about being hard to top?

However, *Zend Studio* has a reputation as being the Rolls Royce of PHP development environments, and Zend's regular updates to their software shows their commitment to keep improving.

All the usual IDE goodness is in there, including syntax highlighting, code completion, built-in FTP client,



**Conditional breakpoint control means the end of having to step through thousands of lines of code.**

etc. *Zend Studio* also has variable completion, which pops up a list of in-scope variables when you type **\$**.

One of the core advantages to *Zend Studio* is that it's written entirely in Java, allowing developers to use the same IDE wherever they find themselves coding.

## New in 2.6

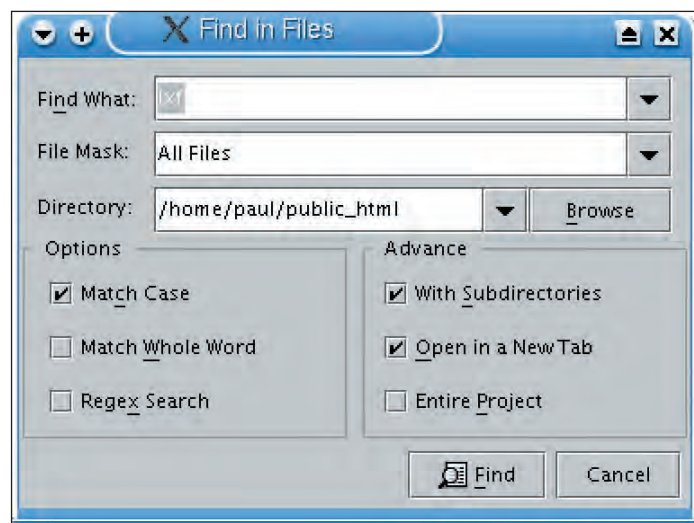
So, what does the new version of *Zend Studio* add to warrant a full-page review? Well, scoring a whacking ten out of ten on the LXF cool-o-meter is the new CVS integration. Previously there was a sidebar that could be used to load files and also manage FTP connections. The CVS integration has been executed as a popup menu to this same sidebar, which allows you to select any file from your project and check-out the latest copy from the server, or even compare changes between your local version and prior versions.

Another excellent feature addition is the ability to run cross-file searches, which allows you to extend your search criteria to all files in a given directory. You can also set bookmarks in your scripts so that you can jump to and from set locations easily.

From a debugging perspective, the key addition in *Studio 2.6* is conditional breakpoints, which means you can set a breakpoint to only stop execution when a given conditional is met inside the code. For example, conditional breakpoints can be used to pause script execution when variable **\$MyVar** is equal to 10, or, indeed, any other boolean expression that the PHP script is aware of.

## Faulty docs: begone!

With *Studio 2.6*, Zend took the opportunity of formatting the documentation in a much simpler manner than seen in *Studio 2.5*, and that means it loads much faster, and doesn't even crash Netscape any more. While I can't take credit for bringing about this change, clearly I wasn't the only one who disliked the graphics-heavy interface in the previous version. It's good to see Zend



**Finding text in many files is a cinch now, thanks to *Zend Studio 2.6*.**

listening to customers and making big changes where required.

The documentation itself is as well written and helpful as ever, although it's lacking descriptions of some of the new features in *Studio 2.6*. For example, if you're interested in finding out how to set conditional breakpoints, you're out of luck – you'll need to refer to the (currently being updated) online docs. Of course, how you *get* to the online reference is another fight...

Annoyingly, the documentation as a whole is still very Windows-centric, however I'm happy to forgive the Windows references on the grounds that they dropped the bulky graphics.

## It's the little tweaks...

Sure, there are new features galore, but Zend hasn't left it at that – there's a whole raft of tweaks and adjustments that make the program that little bit easier to use. My particular favourites are the improved regular expression searching, but also you now have the ability to customise which keys perform what action in the program. Those of you who are more used to Windows development can simply select *Visual Studio* as the key layout.

Finally, mouse wheel support now works where its applicable in the program, which is always nice.

*Zend Studio 2.6* is a much more refined, sleeker breed of PHP IDE than its predecessor – Zend seems to have pulled out all the stops to make as many improvements as it can without breaking existing functionality. As a result, *Studio 2.6* makes an attractive blend of new features, balanced by well-polished features introduced in 2.5. Furthermore, it's good to see that the system requirements haven't changed from *Studio 2.5* – in fact, if anything the program seems a little more responsive on the same hardware.

If PHP is your business, *Zend Studio 2.6* is what you want, what you need, and, at the same great price as before, what you should buy as soon as you can. **LXF**

## VERDICT

Features	10/10
Performance	8/10
Ease of use	8/10
Value for money	9/10

A well thought-out upgrade that will streamline your everyday development – shame about the continuing documentation flaws, though. If you use CVS, upgrade now.

**LINUX FORMAT RATING**  
**9/10**



## EMAIL SERVER

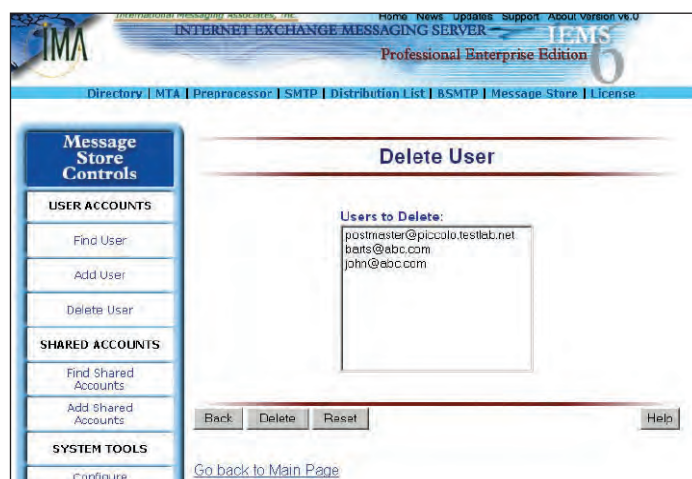
# Internet Exchange Messaging Server

With an eye on corporate mail servers, **David Coulson** looks at what's in the latest release for Linux users in particular.

Mail server software for networks. See also *Sendmail*, *Qmail*, *Exim* for similar functionality.

- **PUBLISHER** International Messaging Associates
- **PRICE** From \$99
- **WEB** [www.ima.com](http://www.ima.com)

Email is an integral part of most modern businesses. There are a wide variety of Open Source mail servers available for Linux, from the traditional *Sendmail* package, security issues notwithstanding, to more recently developed services, including *Qmail* and *Exim*. While many of these are particularly powerful and are capable of handling millions of messages per day, from a configuration point-of-view, they are certainly not for the novice or non-technical user. These servers only provide SMTP delivery, so to actually get mail onto the clients requires an extra set of network services to handle POP3 or IMAP4, meaning an extra set of configuration files and a whole selection of new things to go wrong.



Adding and deleting users is easily done via the web interface

Individual components of a mail service can be useful, but some don't have the time or the budget to deal with highly complex configuration files when really only two or three lines need to be modified in order to get it all working happily. Fortunately, a number of companies are offering 'all-in-one' mail server packages, offering everything from SMTP to POP and address books in one single distribution. One of these is *Internet*

*Exchange Messaging Server* from International Messaging Associates, commonly known as *IEMS*.

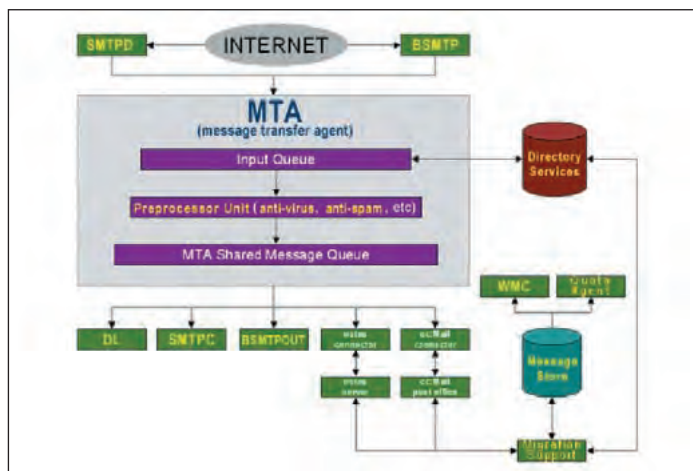
## Installation

We only looked at the Linux version of IEMS, although IMA ship a version which will run on Windows systems, making it a viable alternative to *MS Exchange* and *Lotus Domino* mail servers. Of course, from a TCO point of view, running it on Linux is far more cost effective as there are not the recurring license costs which are found with Windows' platforms.

As *IEMS* relies on a wide variety of third-party services, such as *Apache*, it will only install cleanly on a number of distributions. If one is building a new mail server and has a choice of distributions then this is not a major issue, but if there is an existing server running other services, then getting *IEMS* to work properly will require some manual configuration file modification, at the very least with *Apache*. Most distributions which use RPMs can have *IEMS* installed on them, although one may be able to install it on other distributions, such as Debian, by using a utility like *alien* to convert the RPMs to another package format.

*IEMS* installs with a very simple `./install` command which asks a number of basic questions about the location of the `httpd.conf` file then gives the URL to administrate the *IEMS* installation. One can either perform a single system installation, where everything is located on a single box, or in a distributed environment where certain services are installed on more than one system. For some strange reason the supplied manual suggests distributing the mail queue between the boxes using *Samba*. Why someone would use *Samba* over NFS in a purely UNIX environment is anyone's guess, but having the mail queue distributed from a central source makes for a single point of failure. Should the main queue server go down, then the rest of the mail servers are useless. In a large scale server environment, this sort of distribution quickly becomes very limiting, so while *IEMS* does offer distributed services, if anyone is looking for a system which is capable of surviving a system failure, then a home grown alternative will be in order. Interestingly, one can install some services on Windows and some on Linux, which would require the use of *Samba* for either mail queue distribution or mounting from a Windows share.

Once one has the *IEMS* system up and running, it is required to supply an existing license key to get it all working, or to register. Up to three users is free and there are varying levels of licensing available depending how many users will be using the service. Upon installation one can select to have a number of services running on the *IEMS* system and unless it is being used in a distributed environment then all should be selected. The main part of *IEMS* is the pre-processor which handles all of the message handling, as well as



How all the components of *IEMS* interact with one another is somewhat complex, but enables easy distribution over multiple physical systems

interaction between the separate components of the mail service. *IEMS* supports all of the standard capabilities one would expect a mail server to perform. It handles SMTP mail using a simple *smtpd* daemon, and supports both single message SMTP delivery as well as BSMTP for large amounts of mail distribution between servers. The use of BSMTP is significantly more efficient than individual messages, as it only requires a single connection to another BSMTP server, although it does mean that on a slower connection a single large message may hold the entire queue up. *IEMS* also contains a SMTP client, known as *smtpc*, which can handle sending mail from external programs via SMTP, similar to the *sendmail* tool found with most Linux mail servers.

POP3 and IMAP4 are also supported, although as *imapd* and *pop3d* are used, it will authenticate against PAM, rather than a separate database, resulting in the requirement of a user account for each mail account. For significantly large systems, where each mail user does not need a shell account, the use of a database such as MySQL or PostgreSQL, or possibly even LDAP, greatly increases the ease of user handling, plus it increases security

somewhat as it no longer requires a large number of shell accounts on the mail server. Both *imapd* and *pop3d* are basic mail collection services commonly found on most distributions, although on many mail servers those are replaced with *uw-imapd* or a courier mail service. Of course, one could switch to one of these when using *IEMS*, but this may not have a pleasant effect upon the functionality of the server in some cases.

## Not supported

Being fairly basic, the SMTP daemon does not support SSL or SMTP AUTH capabilities, which may become an issue for any companies which want to provide access to roaming employees without VPN access, or anyone wanting to setup mail accounts for others which include outgoing SMTP mail from an external client. One could install a separate mail server to perform this function, although this obviously makes the entire *IEMS* installation somewhat redundant. Using SSL would also be preferable in cases where passwords are sent in plain text across a link, such as during POP or IMAP authentication. One could use *sslwrap* to proxy the connection, but it would be nice to see this as a default, rather than something the user has to figure out on their own.



The web interface allows every aspect of the *IEMS* install to be configured.

*IEMS* also supports the use of LDAP for directory services, through the use of *Idapserv*. We have never heard of *Idapserv*, so why this is used rather than the very popular *OpenLDAP* package is questionable. Nearly all mail clients support LDAP directory access, so in an office environment this is a very valuable resource to have available. As with the SMTP end of the system, this is all configurable via the web interface, so one does not need to understand LDAP to add entries to the directory. Of course, only the main administrator has access to this, so an outside user is unable to modify the contents of the directory server.

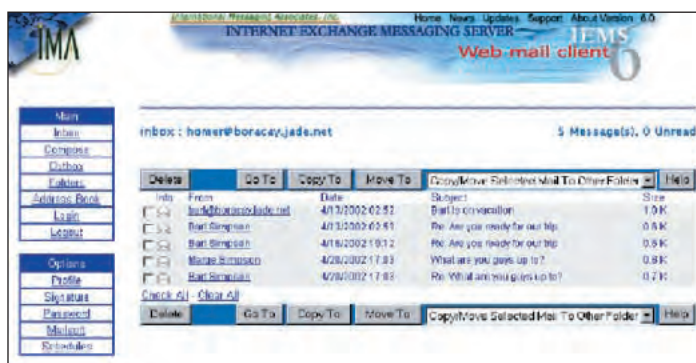
One also has mailing list capabilities available, making it very easy to create corporate information lists or to notify customers of new products or services. The list handling is fairly basic and does not support archives, although very useful features including message digests is available.

In addition, *IEMS* can also perform filtering based on virus and spam checking routines, using third-party filtering systems. As the actual decision is not based upon *IEMS*, the success of filtering by *IEMS* should be reasonably similar to that of any other mail server running the same or similar software. As Bayesian mail filtering seems to be one of the most successful spam limiting systems, then filtering client side may be more productive, assuming that bandwidth is freely available to download the spam messages.

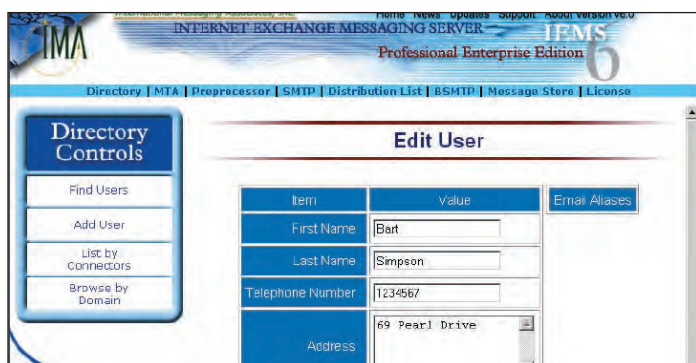
*IEMS* is a really nice 'all-in-one' package for Linux and almost anyone

could quite easily get it installed and up and running without a great deal of effort, even with limited Linux knowledge. With the support for *cc:Mail* and *Lotus Notes*, those migrating from *Exchange* or *Domino* will find the transition much simpler and the users will probably not notice a great difference in the actual usage. However, *IEMS* does have some limitations for large distributed mail services, and a more structured load balanced cluster will usually perform more effectively, plus there is no longer a single point of failure on the network. Certainly any Linux user with significant knowledge can setup something almost as well specified, the web interface will make life easier for administrators without much command line experience.

There is a free version of *IEMS* available from [www.ima.com](http://www.ima.com), allowing anyone to test out *IEMS* and run it on a small scale server. Most will find *IEMS* to be a very powerful and capable service, and those that don't can likely setup their own mail server without much heartache. **LXF**



Webmail is a very useful component of the *IEMS* package, allowing those without mail clients to handle email.



The directory services run on an LDAP server, but there is no need to learn how LDAP works to manage it.

## VERDICT

Features	8/10
Performance	7/10
Ease of use	8/10
Documentation	9/10

*IEMS* is a great replacement to *Exchange* and *Domino*, although for larger installations it may prove expensive.

**LINUX FORMAT RATING**  
**8/10**



# Implementing Intrusion Detection Systems

**Paul Hudson** dons his white hat and investigates a new hands-on guide to securing your network.

■ **PUBLISHER** Wiley  
 ■ **AUTHOR** Tim Crothers  
 ■ **ISBN** 0-7645-4949-9  
 ■ **PRICE** £29.95

**F**luffed out to just 316 pages, *IIDS* is not the biggest book in its area, and don't be surprised to find several listings of 50+ lines of log output that could easily have been explained in just two or three lines.

Indeed, it's infuriating in several places; eg while explaining how FIN stealth scans work, the book has 20 lines of log dumped along the lines of FIN/RST/FIN/RST, which doesn't actually demonstrate what a *successful* FIN scan looks like, so the reader is left with 20 lines of useless text.

You would think that perhaps the author might have bothered to a) cut the log down to five lines or under to make the point clear, and b) actually used the log for something, other than saying "Here is a log. This is what a successful FIN scan *doesn't* look like"

It's also a little embarrassing that on page 257, the author still needs to put in brackets what TCP, HTTP, FTP, and others stand for. What's a glossary for?

Is the book all bad? Well, no. In fact, because when the author isn't finding cunning ways to bump the page count up, he actually writes with a great deal of clarity and skill. Particularly interesting for newbies to network security is the entire chapter on network-based detection. There's good detail as to what makes up packets at each layer of the TCP/IP protocol suite.

Furthermore, the sections describing how to spot the various attacks will prove to be gold dust for junior network admins, if somewhat watered down with log files. Of interest to a wider audience is an insightful chapter discussing legal issues of intrusion detection. Finally, the author closes with a sample intrusion detection deployment, which is fairly thorough and happily Linux-specific.

All in all, *IIDS* is clearly authored by someone who has a lot of experience in the field and is passionate about sharing



it with others. However, I would also say that a lot of his good work has been edited into oblivion as all too often important gems of information are buried beneath line after line of extraneous and often irrelevant info.

Perhaps with a second edition this book would have a chance to shine as it should. Otherwise, as it stands, *IIDS* is not the best book out there by quite a large margin. Unless you're completely new to network security, I doubt *IIDS* will tell you anything you didn't know already. If you're looking for basic descriptions of network attacks and exploits, this book is for you. If you're

looking for more advanced topics such as nmap usage, setting up specific detection systems, or even personal tips and tricks to make your life easier, you could do much better.

## VERDICT

A great book for beginners, but at the price you could do better. Wait for a second edition.

**LINUX FORMAT RATING**  
 6/10

# Linux For Windows Administrators

**Amias Channer** examines the escape routes for Windows administrators going under the wire...

■ **PUBLISHER** Sybex  
 ■ **AUTHOR** Mark Minasi and Dan York  
 ■ **ISBN** 0-7821-4119-6  
 ■ **PRICE** £37.99

**W**ritten by an experienced Windows admin with advice from a UNIX guru and now in its second edition, this book contains enough info to install and configure a Linux system to run most common NT server tasks and is a good primer for further reading.

Given the amount of options for almost any aspect of Linux configuration, it can be hard to offer a simple guide. To escape this situation Red Hat 7.3 is used

as the example distro. Fortunately some of the distro dependency is alleviated by focusing on the command line instead of GUIs as a primary means of control.

A brief history of Linux is given, as well as attempting to explain the bizarre appeal of recursive acronyms. The install section is straightforward and whizzes you through without going into too much detail about what the commands do; this is great if your hardware is well supported, but not so useful for those with more obscure hardware.

All of the major services (*Apache*, *Sendmail*, *BIND*, *Samba*, DHCP, FTP and IP networking) are discussed in just enough detail to get going and with plenty of examples and references to the windows equivalents but not the



protocols or the finer details. These sections are clear, factually accurate and concise enough to stand up on their own as beginners guides.

Sadly, some very useful hints are left out, such as using tab completion on the command line (only needs a paragraph) and the insistence on symlinking manually (as opposed to using `/chkconfig`) for adjusting a daemon's runlevel or the complete lack of information on Linux user groups. Omissions aside, this book's well judged accessibility should make it ideal as a

first Linux book for Windows admins by concentrating on specific tasks instead of the often overwhelming possibilities.

## VERDICT

A good combination of gentle humour and balanced comparisons between NT/2K and Linux systems.

**LINUX FORMAT RATING**  
 7/10

# DNS for Dummies

Paul Hudson looks at the DNS “reference for the rest of us” and finds himself most impressed.



- **PUBLISHER** Wiley
- **AUTHOR** Blair Rampling and David Dalan
- **ISBN** 0-7645-1683-3
- **PRICE** £18.95

*Dynamic DNS, round robin, TTL, BIND* – all are covered in a solid amount of detail, and, even more impressively, there's a fairly comprehensive section on DNS security.

If you're thinking “this doesn't sound like it's pitched for dummies”, relax – there's a lot covered in the book, that's true, but it's written with style and wit that makes it *enjoyable* to read. Too many books concentrate on their authors using long, technical terms to show off their knowledge, and *DNS for Dummies* goes entirely the opposite route.

## Bad points?

As this is *Linux Format*, I guess I'm obliged to point out that the book covers Microsoft implementations of DNS as well as UNIX implementations, so there are perhaps 20 or so pages which may not be of interest to all readers. However, I think covering DNS



from a cross-platform perspective is crucial in a heterogeneous environment such as the Internet.

The other down-side is that more experienced DNS users are less likely to get as much useful information out of the book, simply because the book is aimed at the absolute beginner through to experienced DNS user level. For example, the ubiquitous Dummies “Cheat Sheet” inside the front cover includes space for “My DNS Client Settings” to allow readers to write in their DNS servers' IP addresses – perhaps a little too basic for some users.

However, the content of the book is sound, and a great read. In my opinion

the fact that it covers more than just UNIX is actually an advantage, and also I'm quite happy to ignore the few parts of the books that are aimed at absolute beginners – after all, it's better to cover things that some people might ignore than not cover things and have people fail to understand more advanced concepts.

## VERDICT

Interested in DNS? This is for you. Not interested in DNS? Buy it anyway. A great addition to anyone's bookshelf.

**LINUX FORMAT RATING**  
**10/10**

**T**he Dummies series has a long history of quality, though pitched at a very low level. As such, some tend to avoid them, preferring much more condensed books that don't water info down.

The book, to say the very least, is utterly engrossing, and I wish I could be somewhat more descriptive than just saying “wow”.

Rampling and Dalan waste no time in the book – they cover the absolute minimum introduction to TCP/IP to get you started, then jump into a no-holds-barred explanation of how the DNS system works on both the client and server sides.

EASY TO BUY • EASY TO SET UP • EASY TO SEE

DOMAIN NAMES  
 WEB HOSTING  
 E-COMMERCE  
 VIRTUAL DEDICATED SERVERS

  
**log on**  
 and be seen

Pick up a new domain name from £3.50 or a web hosting package from £30 per year. Backed with robust support and technology, Easyspace is one of the largest ICANN accredited registrars in the world, and the UK's favourite domain name registrar.

log on today at: [www.easyspace.com](http://www.easyspace.com)

Established in 1997 with over 1/2 million customers. Accredited ICANN registrar & nominet member. Prices exclude VAT.

**Easyspace**   
 your perfect partner for the web



# Hot Picks

The best new Open Source software on the planet!



**Mike Saunders**

A coder since Amiga times, Mike's a Linux and BSD guru.

This is the place where we get to profile some of the hottest software around.

Each month we trawl through the hundreds of open source projects which are released or updated, and select the newest, most inventive and best for your perusal. Most of the Hot Picks are available on our coverdiscs, but we've provided web links if you want to make sure you have the very latest version.

If you have any suggestions for things that we should cover, email us at [linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

## Hot Picks at a glance

Rhythmbox	45
Qixite	46
YAALA	46
Junkie	47
Avidemux	47
Darkstat	48
Zoinks	48
QtVision	49
GAV	49

### HOT PICKS AWARD

Everything covered in our Hot Picks section is unmissable, but every month we'll be singling out one project for outstanding brilliance. Only the very best will be chosen!

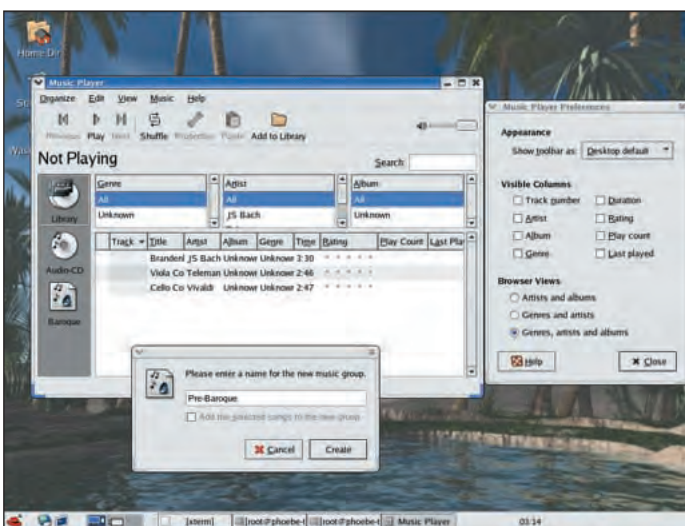


## MEDIA PLAYER

# Rhythmbox



■ **VERSION** 0.4.1 (Net 0.4.5) ■ **WEB** [www.rhythmbox.org/](http://www.rhythmbox.org/)



Rhythmbox running under Red Hat's Bluecurve GNOME 2.2.

Being able to play, manage and modify all manner of media files is now an essential part of any desktop OS. The outrageous popularity of MP3s, DVDs, AVIs and other formats – and increasing sharing of media thanks to better Internet connections – has led to GNOME and KDE developers looking for more ambitious solutions than simple file players.

In issue 36, we looked at *GStreamer* in *Hot Picks*; in a nutshell, this provides a plugin-based pipelined media handling framework for other apps to build on. The goal is to avoid unnecessary code duplication, with programmers being able to include media handling support in their apps easily and quickly (instead of hand-coding it into each program).

*Rhythmbox* is one such util making use of the *GStreamer* back-end, and it's shaping up to be the main media playing tool in future GNOME releases. Taking its inspiration from Apple's slick *iTunes* program, *Rhythmbox* is currently seeing a lot of active development –

so much so, that a temporary fork has formed. *Net-Rhythmbox*, included on the coverdisc, is a continuation of the stable branch with an emphasis on completing the proposed features and “replacing other media players like *XMMS* as soon as possible”.

Hopefully the fork won't continue for too long, and the new developments will be merged in future releases. Still, whichever you choose to try, you'll need the full complement of GNOME 2 and *GStreamer* libraries on your system, so compiling from source can be tricky if you're hunting down dependencies. If you use *apt-rpm*, the *GStreamer* apt repository has all the goodies.

*Rhythmbox*'s main window is clearly geared towards organisation: the main file list can be sorted by each of its columns (name, length, etc), while the left-hand pane holds buttons for selecting Groups. These can be used for holding different music genres, and a standard toolbar along the top contains the volume control and operation buttons. Currently,

*Rhythmbox* supports MP3 and Ogg formats, although more are planned for future releases.

## Hit me with your rhythm stick

Pleasing features include the ability to rate songs (one to five stars) and monitor how often they're listened to (or the last time played), and Groups can be sorted by these attributes. Drag-and-drop between Groups is supported, along with a search facility and the standard shuffle and repeat functions, and it's all satisfyingly simple to use. The competently written manual is chock full of tips and screenshots too.

In terms of configuration, the Preferences window allows general changes to the toolbar and displayed columns in the file list, together with the fields show in the overall Browser component. Annoyingly, the column widths don't appear to be changeable in the file listing without resizing the whole window – our only major aesthetic gripe, but it does make it difficult to read longer track titles.

Features currently in development (planned for future releases at some point) for the main tree include Internet radio functions, visual effects (like the visualiser plugins in *XMMS*) and support for CD playing, ripping and burning to writable media. The *Net-Rhythmbox* tree already includes Internet radio code along with a few other nice additions, and is the version we'd recommend using until the main tree puts out a new stable release.

Even though it's not the most feature-packed media player in existence at the moment, *Rhythmbox* shows a huge amount of potential and could prove to be highly important in GNOME's development as a modern and flexible desktop platform. It's easy to work with and will be familiar to Windows and Mac users, while at the same time providing the necessary advanced features for those with large amounts of music to manage. Stable, fast and nice on the eyes, *Rhythmbox* is worthy of our Hottest Pick award this month and is well worth watching. Give the *Net* fork a spin today.

## AUTOMATIC WEBSITE MAKER

# Qixite

■ **VERSION** 0.0.8 ■ **WEB** <http://qixite.sourceforge.net/>

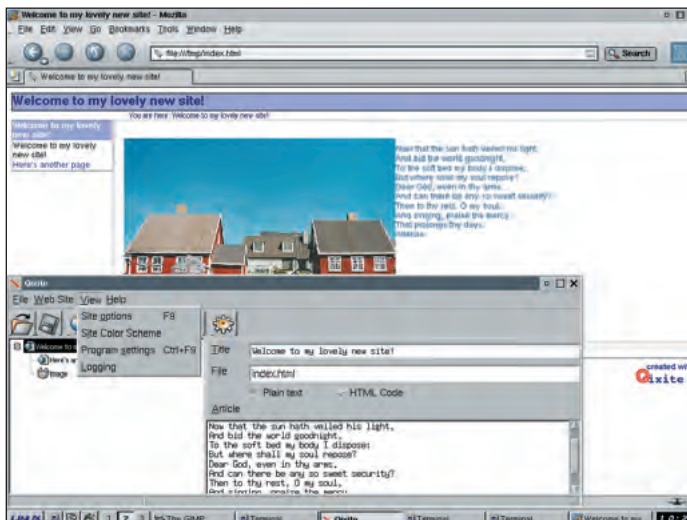
**A**lmost everyone who spends a lot of time on the Web these days has their own site – ranging from tiny home pages with minimal info of questionable interest to anyone outside the builder's immediate family, to fully-fledged weblogs with links, downloads and other goodies. Most ISPs offer a few megabytes of space, as do free providers like Geocities, so the hardest part is actually putting together the pages. WYSIWYG editors exist, and are fine for working with individual pages, but *Qixite* is a new utility designed to assist in building the overall structure of a site.

Available as Red Hat RPMs, Debian packages or vanilla source tarballs, *Qixite* has been created with the Kylix development tools and as a result requires the *kylixlibs3-borqt*

libraries to run. This gives the front-end a unique appearance, although the Qt widget set prevents it from being totally alien.

When first started, *Qixite* offers a two-paned display and toolbar. Hitting New from the File menu is the first step in generating a site; you're prompted for the initial page's title, its filename and the text (or 'Article') to be contained within. From here, you can add new text sections to the page being worked on, insert images, put in links to other pages, and so forth. The choices available are perhaps a bit limited from the point of view of the more accomplished web designer, but it all works pleasingly well.

Once the pages are in place, hitting Generate will write out the HTML files (in /tmp by default) which you can then view in your favourite



**Qixite** in the bottom-left, with its results being displayed by **Mozilla**.

browser. The result is a smooth table-oriented site, with links to the other pages down the left and main content on the right. A few ready-made themes are available, which alter the colours and background images, but individual elements can be coloured through the dialog boxes too.

Simple as it may be (and as we said before, clearly not targeted at

the demands of professional website designers), *Qixite* is an effective little utility and provides just enough options and commands to make minimal but usable sites. *Weblint* didn't find any serious troubles with the generated HTML either. Sweet, then, and we hope to see more depth of commands and variety of themes in upcoming versions.

# SERVER LOG ANALYSER

# YAALA

■ **VERSION** 0.6.2 ■ **WEB** <http://verplant.org/yaala/>

**R**ummaging through server log files to get a general overview of what's going on can be one of the most RSI-inducing chores known to man. Similarly, anyone trying to convince their non-IT savvy boss that a few system upgrades are needed can't just point them at a 10MB `access_log` without being treated to a great deal of head-scratching; sometimes it's more appropriate to have a cleaner visual representation of how stressed a machine is, and what people are doing with it. A number of log analysis tools exist, and a new up-comer is *Yet Another Advanced Log Analyser*, or YAALA for brevity.

YALA differentiates itself from other applications of its nature by allowing the user to generate the parser themselves. Supplied in the

tarball are parsers for the NCSA and *Apache* web servers, the *Squid* proxy cache, *Postfix* MTA and various FTP

servers which share the established *xferlog* format. This is enough for many users (*Apache* logs being the most commonly targeted by this type of tool)

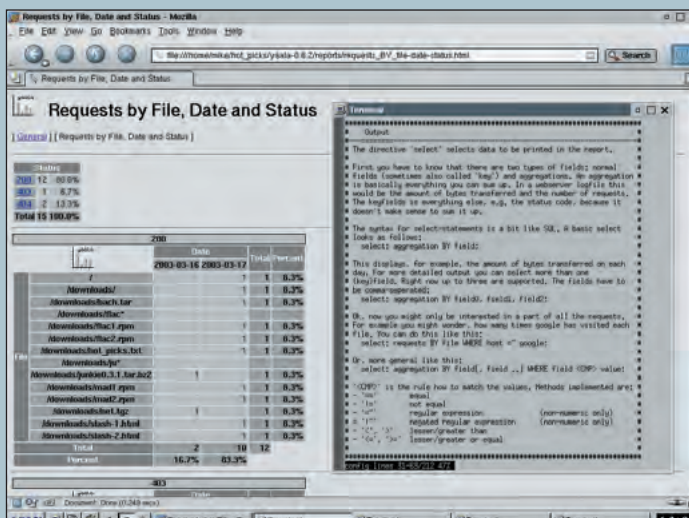
As it's written in Perl, **YAALA** needs no compilation and the package size is tiny. Before starting, though, you'll need to edit the 'config' file, which sets up the default output directory options, formatting and

other settings. The most crucial part of this file is the 'select:' line, which mimics SQL in picking out certain fields depending on their value. Here's an example, for a web server log:

This will produce a table listing the names of files accessed and their hit counts for each date that the log file covers. These select commands can be expanded to include a multitude of fields, and a WHERE addition can force it to match certain values. Powerful, then, but not too convoluted in producing decent reports.

YAAFLA's HTML output consists of tables and greyscales, and worked well in all of the browsers we tried. Development is speeding along (two new releases appeared while preparing this *Hot Picks!*) and hopefully other developers will add more parsers to be included in the package. Great stuff.

Remember, old boxes equals new toys for geeks like us, and an across-the-board upgrade is often a great way of gathering 'defunct' hardware for personal projects!



**Output from small access log, with the parser explanation on the right.**



## GRAPHICAL FTP CLIENT

## Junkie

■ **VERSION** 0.3.1 ■ **WEB** [www.doomed.org/~pin/junkie/](http://www.doomed.org/~pin/junkie/)

**D**espite the popularity of the Web, email and P2P networks for sharing files, FTP still remains heavily in use

through its simplicity and ubiquity. Almost all major open source projects put their works up for grabbing via FTP, and while most modern

networking OSES include the basic command-line tool, there's a scarcity of decent graphical alternatives. *Junkie*, the "juvenile jocund pukka GUI FTP-client, the job of joyous jocular juniors" (as the developers so succinctly put it) is built around GTK2.

Providing you have the necessary *Glib* and GTK2 development libraries installed, building from source should be hitch-free. Unfortunately, it'll segfault on startup if a sound card isn't present (it relies on a working /dev/mixer), but compiling with --disable-sound and related options should cure that.

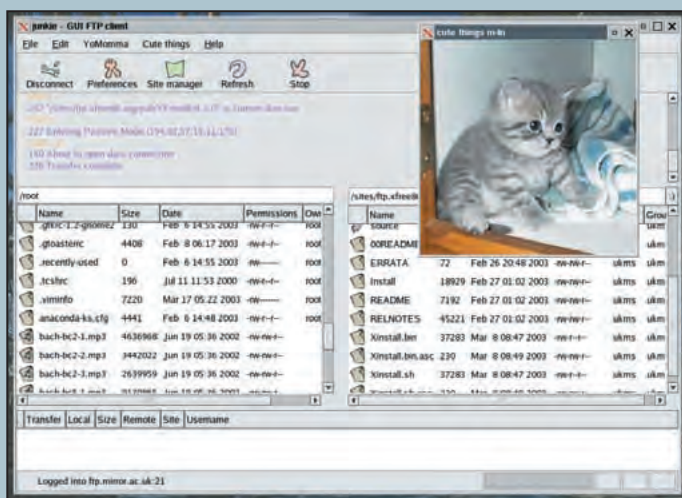
*Junkie*'s main window resembles gFTP (an older and popular GTK-based client) very closely, with the workable four-pane setup of messages, local files, remote files and transfer operation listings. Hitting 'Connect' throws up a stock window for choosing the host and port to connect to, together with active or passive mode and username/password if necessary.

Like bookmarks on steroids, the Site Manager assists in organising regularly visited hosts; these can be

grouped together and shifted around, and the list can also be password-protected for the terminally paranoid. Default transfer and login options can be altered through the prefs box, in addition to cache settings, toolbar themes and external applications.

So far, so good – it's a fast and pleasant FTP client with a good supply of general-purpose features. But! *Junkie* is clearly aimed at those who spend a good deal of time watching downloads, and consequently comes supplied with some sound files, a sickly-sweet album of kitten and puppy photos, and more "Yo Momma" jokes than even a potty-mouthed rapper like Eminem could use in a lifetime.

*Junkie*'s developers clearly have a sense of humour, and anyone suffering from intense boredom while waiting for large ISO images to download would be wise to take a look. Lots more pictures and jokes would be welcome in those endless hours, but then the client itself might take 15 hours to download... Why not read a book instead?



Too much downloading? Some kittens and jokes will make the megs fly by.

## VIDEO EDITING TOOL

## Avidemux

■ **VERSION** 0.9rc2 ■ **WEB** <http://fixounet.free.fr/avidemux/>

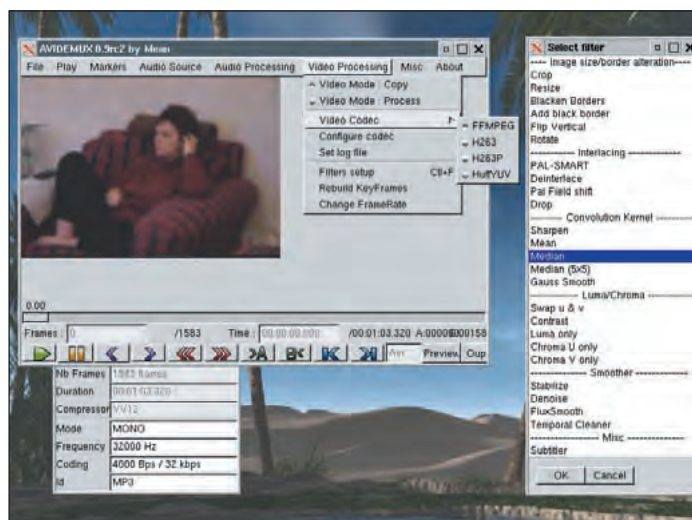
**A**mong the multitude of markets that Linux and open source software is working to get a foot-hold in, image and film processing is one of the more lucrative. We've seen Linux boxes involved in the production of major blockbuster movies (including *Lord Of The Rings* and *Titanic*), and with *Film-GIMP* (now *CinePaint*) proving to be hugely popular, others are fighting for a wedge of the video processing pie. Enter *Avidemux* – short for *AVI de/multiplexer*.

*Avidemux*'s primary goal is to import a video stream, perform operations on it with the supplied filters, mix in another audio stream along the way, and write out the result. Binary packages for the major distributions are available (along with supporting codec libraries), but when building from source, it'll pick up on

the codecs you have installed and configure appropriately for them.

The program itself is built on the GTK1 toolkit and presents a rather sparse interface on first inspection. Most video professionals will prefer it that way, of course, instead of being assaulted with bags of eye candy that detracts from the program's functionality, but newcomers may find it overly daunting. At least there aren't too many different ways of performing the same task to confuse newcomers! After opening a video file, the first frame image is joined by a tooltipped navigation toolbar, and status line with all the details.

Depending on the libraries *Avidemux* is compiled with, it can work with AVI, MPEG 1/2, groups of BMP images and others. Supported video codecs include DivX (3.11/4/5), WMV2, HuffUV and more, while the



*Avidemux*'s main window – the far-right box lists all supplied filters.

audio streams can be MP2/3, AC3 or even WMA (though the latter is only partially supported at the moment). This is an impressive list – particularly for the Microsoft formats – and will cover most bases.

In terms of supplied image filters, *Avidemux* is equally impressive with the generic range of resize, crop, rotate, smooth, sharpen, etc. joined by drop. This watches for suspicious

glitches in frames to avoid 'VHS drop out', subtitling (with font options) and many others.

Despite the interface being slightly inhospitable to new users, and possible concerns with stability (we experienced a few X lock ups in testing), there's no doubt that *Avidemux* is a flexible and powerful video tool and another string to Linux's bow.

## NETWORK MONITOR

## Darkstat

■ VERSION 2.5

■ WEB <http://members.optushome.com.au/emikulic/net/darkstat/>

**Q**uestion: what's the best way to monitor the CPU and memory usage of processes on a machine? Many users would agree that *top* does the job nicely, and there are a few GTK and KDE-based

equivalents around, but what happens if you're being Slashdotted and need a similar tool for keeping tabs on network usage? One such utility that could be used in such a situation is *ntop*, but programmer Emil Mikulic wasn't overjoyed with its memory-munching habits and stability issues, and created *Darkstat* in response.

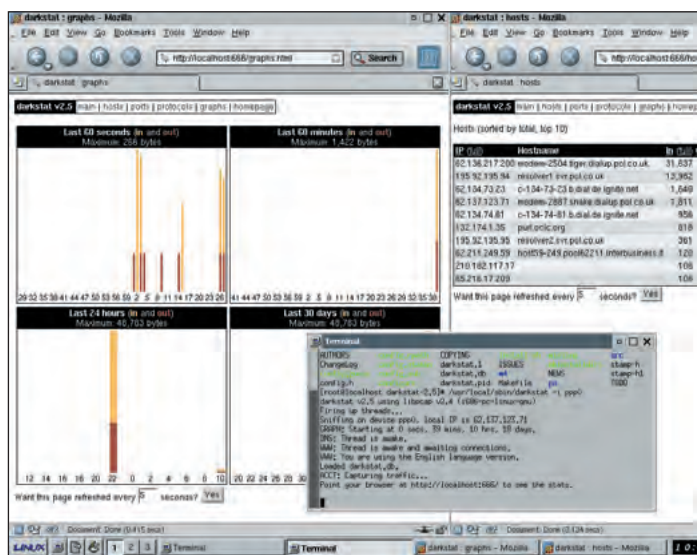
In essence, *Darkstat* is a network traffic monitor which runs in the background, counting up the incoming and outgoing traffic from a machine, among other things, and presenting the results through a built-in mini webserver (<http://127.0.0.1:666> is the Satanic default). A simple enough concept, and the reasonably small download only requires *libpcap*, which is supplied and installed with most distros, so it's nothing too obscure that you'd need to hunt around download sites for.

After installing, *Darkstat* can be started just by calling its binary; from there, it'll throw up a small server on the aforementioned port to display its findings. Options are available to change the interface that's being monitored, port for the web server, and to toggle constant DNS resolving, alongside the ability to pass it packet

filter expressions. There's little in the way of documentation, but the INSTALL file and **--help** flags provide most information.

*Darkstat*'s simple HTML output for the main page consists of a few tables on a white background, with a menu along the top, general info and graphs to visually represent traffic levels. Other pages available include lists of hosts, ports and protocols, and there's a larger graphs section which details the activity over longer time-frames. The graph pages naturally require an appropriate browser to view, but the other lists display correctly in Lynx and Links.

As a small and no-nonsense tool for watching levels of network traffic and related information, *Darkstat* does the job rather well. It's miniscule and unobtrusive, and the output is viewable in virtually every browser. As with most programs that we feature in *Hot Picks* though, as *Darkstat* is being continually developed, there's certainly room for improvement in several areas. For now, some better documentation and perhaps the ability to customise the output (colours, sizes etc.) would be great to see in further versions.



A couple of pages from a poor, over-stressed dialup connection.

## TEXT EDITOR

## Zoinks

■ VERSION 0.2.8

■ WEB <http://zoinks.mikelockwood.com/>

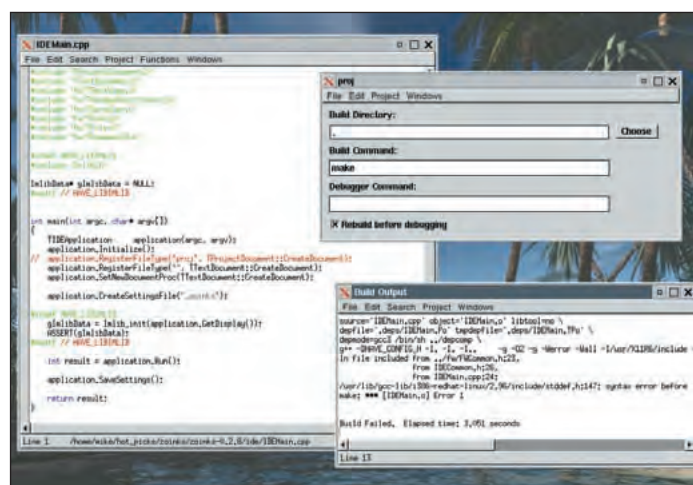
**T**ext editors – is there any better source for a flamewar 'discussing' the merits of your favourite? BSD vs GPL and KDE vs GNOME may stir up some good scraps, but the Emacs vs Vi debate is likely to rage on forever. Still, those looking for something different in a text editor are being presented with new options all the time, and the intriguingly named *Zoinks* is one such lightweight tool. Primarily intended as a programmer's editor, the interface shares many similarities to professional coding environments like *CodeWarrior* and *Code Crusader*.

Written in C++, *Zoinks* will make use of the *ImageMagick* libraries (if installed) for certain HTML editing features; in particular, this simplifies working with image tags, as *Zoinks* will

automagically use the correct WIDTH and HEIGHT attributes. Compiling is straightforward and the resulting binary is small at around 300K, making it a genuinely lightweight alternative to the more traditional Emacs or ViM.

*Zoinks* supports syntax-highlighting for C, C++, Java, HTML and TeX, and can display Mac and DOS text files correctly. Those wishing to run *Zoinks* under KDE or GNOME may find that the inbuilt widget set doesn't fit in well – in appearance it's akin to a mixture of GTK+ and Motif, but uses the standard grey palette so it's not particularly ugly, and copy-and-paste works well with other apps.

Although it performs well as a standalone text editor, *Zoinks* sports a few IDE (Integrated Development



A syntax-highlighted editor window, IDE-esque command box and output from a build. "Zoinks!" is what Shaggy from Scooby-Doo exclaims regularly...

Environment) features for those working on larger programming projects. Selecting 'New Project' from the menu brings up a box where the working directory, build command (usually just **make** or similar) and debugger-activation command can be specified; with this setup, compilation output is sent to a separate log window, and double-clicking on an

error message zooms the editor window to the offending line. Superb.

In all, *Zoinks* excels as a fast and friendly little editor with some nice touches. Naturally, it doesn't have the outrageously large featureset of Emacs or the flexibility of ViM, but for writing code or HTML with minimal distractions it's a clean, usable and interesting new offering.



## TV-TUNER VIEWER

## QtVision

■ **VERSION** CVS150303 ■ **WEB** [www.kwintv.org/rewrite.html](http://www.kwintv.org/rewrite.html)

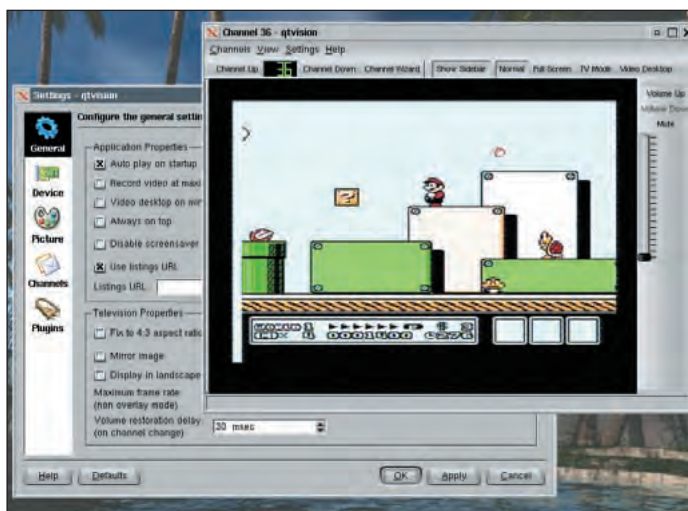
Linux supports a variety of TV-tuner and video-capture cards through its Video4Linux bunch of drivers – such devices are handy for keeping an eye on a television programme while working, or taking screenshots from games consoles, DVD or VHS movies. Traditionally the most popular X app in this field was *XawTV*, but the archaic interface behind it has led to a few flashier alternatives like *QtVision*.

KDE users may have come across *KWinTV*, the desktop suite's original TV-card tool, but problems with the code (and demand for new features) has led to a rewrite, with *QtVision* being the working title. A CVS pull is needed to get the very latest version – or an older Alpha1 tarball can be downloaded – and it needs the KDE 3.x libraries to build. Check out [www.kde.org/anoncv.html](http://www.kde.org/anoncv.html) for

more details on grabbing updated software via CVS.

*QtVision*'s interface is a clean and simple affair, with the toolbar, channel selection list and volume panel surrounding the main display section. Initially, *QtVision* prompts for a device to use – currently, the CVS version supports directly accessing Video4Linux cards, or taking the video stream from the XVideo component of the X server, both through its plugin architecture.

Additionally, a number of other plugins are supplied with the source: OSS mixer for sound (hopefully ALSA will appear at some point in the future), a handful of channel-setting file-format importers (the old *KWinTV* option being notably useful), and lastly some on-screen-display choices. These are all loaded and configured



**NES Mario 3 in action on *QtVision*, with the Prefs box in the background.**

through the Settings dialog, and we anticipate even more plugins appearing in later releases. The usual colour/brightness/contrast levels can be fine-tuned – along with the keybindings – while full-screen, TV-mode and 'video desktop' options are also available.

Most aspects of the application can be modified through the well

laid-out Settings box, and while it suffered from slow-downs on our test systems it proved to be stable. Even though it's still in heavy development, *QtVision* is already showing signs of turning into a solid successor to *KWinTV* and maintains KDE's well-deserved reputation for ease-of-use. Here's hoping *QtVision* will take centre stage in KDE 3.2.

## VOLLEYBALL GAME

## GPL Arcade Volleyball

■ **VERSION** 0.7.2 ■ **WEB** <http://gav.sourceforge.net>

Volleyball was invented in 1895 by William G Morgan. Today, it has established itself over sandcastle building and Charles Atlas-style muscle-flexing as the number 1 beach-showoff game, and televised women's events still attract large numbers of viewers. However, those of us who can't waste time in the sandy coasts of America have to make do with ice-cold water, sewage outlets and splinters from washed-up flotsam. Or stay in with a mug of hot Lift Tea and play *GAV*.

Games players from the DOS days may recall *Arcade Volleyball*. Considering the eyelid-tearingly hard task of making DOS games

run properly under WinXP, a few coders have produced an SDL (ie multi-platform) remake, mixing in some spiffy new features along the way. *GAV* is available in RPM form for a number of distros, but compiling the source should be hassle-free, and once installed, simply running **gav** brings up the retro-style menu screen.

*GAV*'s basis, as a game, is pleasingly simple yet sufficiently challenging against a tough opponent. With three keys, you control a character on one side of the net whose aim is to prevent the ball from hitting the floor. Run left, run right, and jump. When the ball approaches, a quick tap of the jump key (all are redefinable) and some in-air control

will knock it over the net, and it's your opponent's turn to reciprocate. Miss, or spend too long trying to get it over the net, and your enemy gets a point.

Despite the slightly questionable physics, the ball behaves reasonably well and the responsiveness of the player (mid-air changes of direction) is excellent. *GAV* offers more for the professional office-chair volleyballer, though: a few themes are available which discard the cartoony defaults

and replace the background and characters with more lifelike equivalents.

Equally, many aspects of the game can be tuned through the Extras menu, and there's even a network game for long-distance tomfoolery. Although it's a simple concept, *GAV* is fun, especially when playing against human opponents, and requires fast fingerwork to master. **LXF**



**"To me, to you, to me, to you" – watch as their eyes *never* move.**

# Linux for Mac users

Long-time Mac user **Paul Sellars** forsakes the familiar home comforts of his Apple desktop, and tentatively boots into a brave new world...



**T**raditionally, Apple Mac users have craved idiot-proof WYSIWYG functionality above all else. Command lines and shell scripts have played no part in most Mac users computing experiences, and most Mac users have been quite happy about that. Recently, however, things have begun to shift. Apple's Darwin-driven OS X has moved the conventional goalposts somewhat. While it retains an extremely intuitive and easy-to-learn mouse-driven interface, it has undeniably brought some significant changes in its wake.

Thanks to OS X, Mac users like myself have had a gentle introduction to things like user accounts and file permissions. The more inquisitive may have even opened the Terminal application, and been astonished to find a command prompt blinking back at us!

In short, whether we knew it or not, Mac users have been exposed to UNIX – and some have realised that it's nothing to be feared.

A few months ago I decided it was time to confront my fears, satisfy my curiosity, and finally try installing Linux

on my Mac. I don't pretend to be able to present a definitive guide to Linux on the PowerPC platform. What I will try to do, however, is demonstrate just how easy it has become for even the least UNIX-savvy, point-and-click Mac user (that's me) to get started with Linux.

## Why Linux?

Before we get to "how", let's briefly consider the "why" and the "which". Why would Mac users, a breed traditionally fiercely loyal to Apple's own operating systems, be interested running a third-party alternative? The first (most obvious) reason is the cost advantage. Even for a home user, equipping oneself with an up-to-date Apple OS and a suite of apps can be a costly exercise. If you need to equip multiple workstations, those costs can mount up alarmingly.

Some Mac users (for example, those working in pre-press side of publishing) may have no real practical choice but to work with the industry standard packages. However, others may find that they can get by very comfortably with Open Source replacements for their most commonly used applications.

Many Mac users were astonished by the extra stability their systems acquired, after moving from OS 9 to OS X. Suddenly application crashes became rarer, and those that did occur did not entail forced system restarts. Linux offers stability that rivals – and perhaps exceeds – that of OS X.

And given the ease with which it's possible to dual-boot Linux alongside conventional Mac operating systems, the question should perhaps not be "Why Linux?" but rather "Why not Linux?" Rather than asking myself "am I willing to learn about Linux?" I ultimately found myself asking "Am I willing to deny myself access to the wide range of

tools available from the Open Source community?" On balance, I decided the answer was a resounding "No".

## Which Linux?

Many of the best known Intel Linux distributions are also available in PowerPC versions, although PowerPC release schedules tend to lag behind their Intel cousins slightly.

Some of the more famous names currently supporting Apple hardware include Debian ([www.debian.org/ports/powerpc/](http://www.debian.org/ports/powerpc/)), Gentoo ([www.gentoo.org/doc/en/gentoo-ppc-install.xml](http://www.gentoo.org/doc/en/gentoo-ppc-install.xml)), Mandrake ([www.linux-mandrake.com/en/ppc.php3](http://www.linux-mandrake.com/en/ppc.php3)) and SuSE ([www.suse.com/us/private/products/suse\\_linux/ppc/index.html](http://www.suse.com/us/private/products/suse_linux/ppc/index.html)).

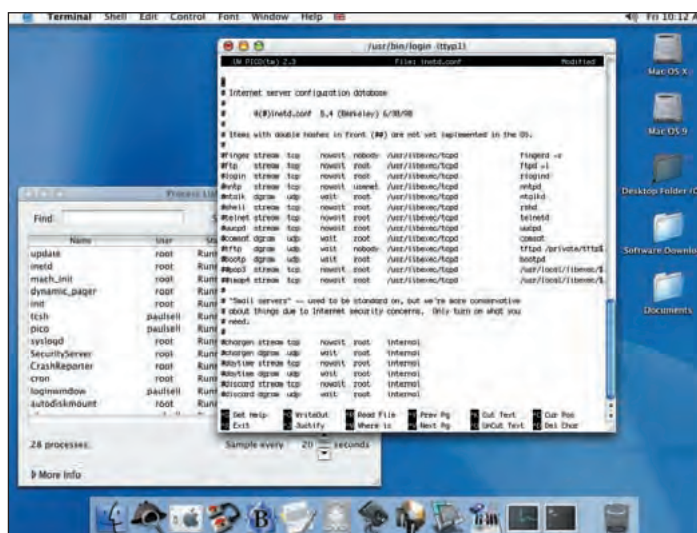
Additionally, the PPC community has its own dedicated distribution in the shape of Yellow Dog Linux ([www.yellowdoglinux.com](http://www.yellowdoglinux.com)).

I'd heard a lot of good things about Mandrake, particularly as an approachable distribution for new users. In the end I decided to opt for Yellow Dog – reasoning that, as PPC specialist, they were probably more likely to offer more complete support for older Mac hardware like mine.

Theoretically, any PowerPC-based Mac (or clone) should be capable of running some PPC distribution. Even machines from the old Mac Performa range could theoretically be made to work, although one wouldn't expect them to perform particularly well for more graphically-intensive desktop use. Any Mac built since the first wave of iMacs should be a prime candidate for a Linux desktop installation.

My own machine, a Blue and White G3 PowerMac, seemed to fall around the middle of most supported hardware lists. Far from the cutting edge, but hopefully not so ancient as

**Beneath its shiny Aqua surface, OS X is UNIX at heart.**





to cause serious problems. After double-checking the system requirements, I began downloading an ISO of Yellow Dog 2.3.

## How Linux?

Installation causes much anxiety among first time Linux users, and I was no exception. Fortunately, my experience with OS X proved to be an advantage. I was already in the habit of dual booting two operating systems on different partitions, and reading through Yellow Dog's installation documents I was pleased to discover that preparing a Mac for a Linux installation is no more troublesome than preparing it for an OS 9 and OS X dual-boot installation.

Having backed up my files and documents to a couple of CD-Rs, I inserted my Mac OS 9 system CD, shut down, and rebooted holding down the **C** key (the standard method for booting a Mac from the CD drive). After booting to the Desktop, I located Apple's Drive Setup application in the System CD's Utilities folder and ran it.

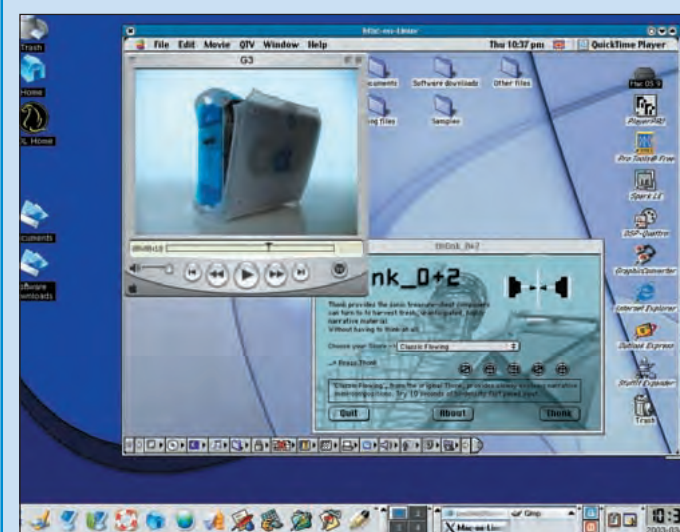
I opted to create three partitions; one for Mac OS 9 (still sometimes essential, despite its apparent obsolescence), one for Mac OS X, and one for Yellow Dog. I chose the **Custom Setup** option, and from the 'Partitioning Scheme' menu selected **3 partitions** to create three equally-sized partitions.

Choosing how different partitions are formatted with Drive Setup is simply a matter of clicking, then selecting the desired option from a pull-down menu. Following the instructions in the Yellow Dog installation guide, I chose to format the first partition as unallocated space (for the Linux install). The second partition would be formatted as HFS (Mac OS 9), while HFS+ would be used for the third (Mac OS X).

I could have chosen the HFS+ files system for both Mac OS partitions, although to do so would have made accessing my Mac OS partitions from Linux less straightforward (more on this later). After installing OS 9 and OS X in the usual way, I took my newly-burned Yellow Dog 2.3 CD-R, and once again rebooted holding down **C**. After a brief pause, a flash of white-on-black text, and a glimpse of a distinctly self-satisfied penguin, Yellow Dog's default graphical installer began.

My trepidation was dissipated. The graphical installation went without a

## MOL Run Mac OS in Linux



Mac On Linux, on the Internet at [www.maconlinux.org](http://www.maconlinux.org) is a free virtual machine environment for running Mac OS on top of Linux. It's capable of running 'Classic' Mac OS versions from 7.5.2 to 9.2.2, and even Mac OS X 10.1 and 10.2.

Mac On Linux is included with many PPC Linux distributions, is relatively simple to set up, and works remarkably well. It's surprisingly fast, and admirably lives up to its author's claims to operate at 'near-native' speeds.

While it doesn't offer all the same functionality as booting Mac OS natively (for example, most USB peripherals won't work), it still provides an invaluable tool for accessing a whole variety of Mac applications from the Linux desktop.

hitch. My keyboard and mouse worked without any intervention from me, and I was in business immediately. I chose Yellow Dog's default installation, in order to avoid being asked too many questions, and was relieved to find the first half of the process involved nothing more taxing than choosing my preferred language, keyboard layout and so on.

The only phase in the installation which might have caused me problems was the extra partitioning required for the Linux installation. Fortunately the Yellow Dog installation guide includes a simple no-nonsense explanation of what needs to be done and why. Having read this through a couple of times (and after getting my head round the idea that one Linux installation might require not three partitions) I felt ready to proceed.

Following the on-screen instructions I created a 10MB partition for a bootloader, a 128MB swap partition, and a root partition to use up the remaining space (about 13GB in this case). One more mouse click, and my partitions were created and formatted.

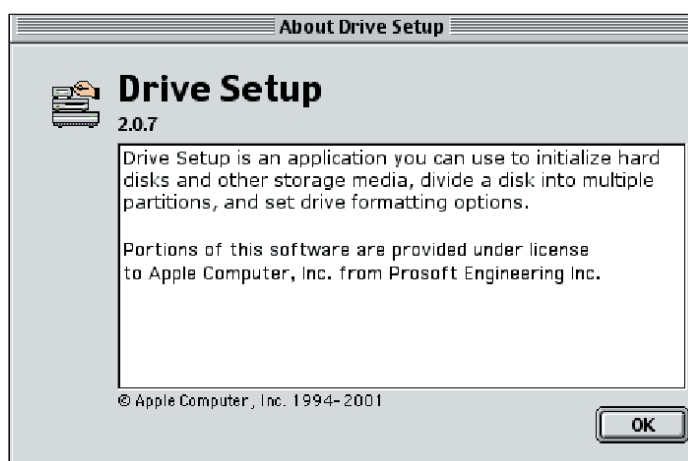
Next I was invited to choose from one of a number of predefined package installations. I chose **development workstation**, for no other reason than that it had the word 'recommended' in brackets beside it. Frankly, I wasn't feeling confident enough to ignore recommendations just yet. In the following screens I was asked if I wanted to configure LAN networking (I didn't), and prompted to choose my time zone.

Next came account creation. I was asked to choose a root password, while simultaneously being warned never to log in as root. Once again, this would have baffled me more if not for my exposure to OS X. I remembered that in OS X a root user does not exist by default, but must be created in order to edit certain configuration files. I also remembered being advised never to log on to OS X as root, as to do so risks breaking things. The same, I now realised, held true for Linux.

After supplying a root password, I was asked to create a normal user account, which I realised would be equivalent to my default user status under OS X. Having done so, the installer moved on to X11 Config. Thankfully this involved no more than choosing my monitor's name from a list, and picking my desired screen resolution and colour depth. I was also



**Apple's Drive Setup utility is a simple tool for partitioning Mac hard disks.**



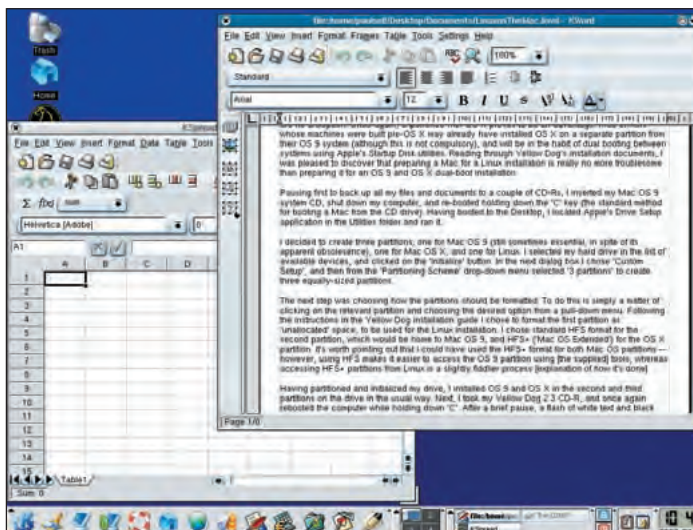


asked if I wanted to use a text-based or graphical login. I opted for a graphical login, trusting that my monitor would work as expected.

The final stage of the installation involved installing a bootloader. There are basically two options. Owners of 'OldWorld' Mac hardware (essentially anything older than the original iMac or Blue and White PowerMac) need to use a special third-party Mac OS control panel called BootX. 'NewWorld' Mac owners need to use a utility called *yaboot* ('yet another bootloader').

Yellow Dog's installer correctly identified my Mac as a NewWorld machine, and prompted me to install *yaboot*. I was asked to locate my OS 9 and OS X partitions, which I remembered from the formatting stage were named `/dev/hda7` and `/dev/hda8` respectively. Finally I was congratulated on having completed the Yellow Dog installation, and prompted to reboot my computer. The whole process had been painless, and had taken no longer than an OS X install.

**KDE's KOffice suite performs all the same tasks as AppleWorks.**



## More options

Other ways of shoe-horning Linux into your Mac

Mac Users who are not willing to go the whole hog, and install Linux for real can still dip their toes in the water, thanks to the UNIX ancestry of OS X.

### The Fink project

(<http://fink.sourceforge.net/>) aims to "...bring the full world of UNIX Open Source software to Darwin and Mac OS X", by porting UNIX applications to OS X and making easy-to-use distributions available for Mac users.

**XDarwin** ([www.xdarwin.org/](http://www.xdarwin.org/)) provides an X server to run on top of

Darwin or Mac OS X. XDarwin can act as an alternative Apple's Quartz technology, and makes it possible to switch between the conventional OS X user interface and XDarwin's X server with a specified key combination. Apple's own X11 ([www.apple.com/macosx/x11/](http://www.apple.com/macosx/x11/)) is similar to XDarwin, but promises closer integration with OS X, and an easier install. It potentially offers the most painless way for OS X users to explore out of the choices listed here.

Rebooting afforded me my first look at the *yaboot* bootloader, which seemed pretty self explanatory. A list of options appears as white text on a black screen. Pressing **m** allows you to boot Mac OS 9, pressing **x** allows you to boot Mac OS X, pressing **l** allows you to boot Linux, and pressing **c** allows you to start the machine from a bootable CD. If no choice is made, *yaboot* times out and boots whichever OS was chosen as default during installation.

## Whoa! Linux!

Having entered **l**, I waited eagerly for a first glimpse of my new Desktop. After a few screenfuls of frankly baffling white text (all overseen by the same smug penguin) a graphical log in dialog appeared as promised. After entering my username and password, I watched KDE 3.0's start up screen, and was presented with my new desktop.

At first glance, everything looked reassuringly familiar. The Trash was in the wrong corner of the screen, but I could adapt to that. The clock had moved too, but I didn't foresee it causing any real problems. The Home icon, I reasoned, must serve the same purpose as the Home folder in OS X. That panel along the bottom of the screen obviously contains program icons, like the Dock in OS X or the Apple menu in OS 9. Oh, and there's a Terminal application, just like in OS X!

I double-clicked on the Home icon, and waited to see what would happen. And waited, and waited, and waited. A spinning hourglass icon appeared in the panel at the bottom of the Desktop, alongside the word *Konqueror*. After a wait of ten seconds, a browser window opened, and very slowly redrew to show the contents of my home directory. Not an encouraging start. Was Linux meant to be this slow and unresponsive?

The answer, of course, is "No". A quick look at the "errata" downloads page at [www.yellowdoglinux.com](http://www.yellowdoglinux.com) revealed that a number of patches for the Dayton Yellow Dog 2.3 distribution were available, including some to address a Qt bug known to cause some KDE programs to start very slowly. Installing these patches proved easy, thanks to the clear instructions, and the problem was solved in no time.

With my new desktop behaving more responsively, I was free to explore the bundled applications. The *KOffice*

apps suggested themselves as an excellent alternative to *AppleWorks*. The infamous *GIMP* won me over within minutes. What a fantastic application. As a free tool for web graphics and general image manipulation, it's impossible to fault. I'd almost be prepared to argue that The *GIMP* alone is reason enough for Mac users to dual-boot Linux alongside Mac OS...

I could go on and on about the exciting Open Source apps I discovered following my first Yellow Dog installation, but I won't. Regular LXF readers will already be aware of them, and there's no sense in preaching to the choir.

Suffice to say, I had tremendous fun, with very few headaches. The whole process of finding, downloading and installing a Linux distro on my Mac was remarkably trouble-free. Without even shelling out for a book, I was able to acquire enough knowledge about the basics of Linux to be able to work productively with the system within just a couple of days. It's difficult to see how the process could have been easier.

Of course, I did not (and still have not) become an expert sysadmin, but that wasn't my goal. As it happens, I did manage a couple of minor tweaks...

## First tweaks

While my first impressions of my new OS were extremely positive, after a few days playing, a couple of minor things began to bother me. First of all, my Apple Pro Mouse has only one button, and I found right-clicking was necessary to do many things in KDE. By default, Yellow Dog has the **F12** key mapped to emulate a right mouse button - but reaching across the keyboard for it felt awkward and inconvenient. The traditional way for a Mac user to access context menus from a one-button mouse is to **Ctrl-click**, and this is what I constantly found myself trying to do.

A quick search of the Yellow Dog support pages turned up a solution. By editing the `etc/sysconfig/mouse` file it's possible to change the default emulated buttons. Moreover **showkey**, available from the command prompt, provides an easy way to find out the relevant code number for any key on your keyboard.

While my first instinct was to use **Control** as my right mouse button emulator, I realised that to do so would prevent me from using all those handy **Ctrl**-based shortcuts (**Ctrl-S** for Save,



Ctrl+O for open, etc.) On a Mac the **Command** or **Apple** key would normally be used for these, but under Linux the **Apple** key has no function – and is therefore an ideal candidate for reassignment. **Showkey** revealed that code number **125** refers to the Apple key (on my keyboard).

Armed with this knowledge, I edited my `/etc/sysconfig/mouse` file. Thanks to an excellent beginners' guide to the Linux Terminal (<http://linux.org.mt/> and [www.newtolinux.org.uk/](http://www.newtolinux.org.uk/)), I learned how to temporarily switch to root in order to be able to edit system configuration files. I'll admit that I found *PICO* slightly intimidating. Fortunately *KWrite* worked just as well.

As root, I entered:

```
kwrite /etc/sysconfig/mouse
and found myself presented with:
Mouse Setup
Run yimouse to re-configure this file
TYPE=usb
DESC="Universal System Bus (USB)"
GPM=imps2
X11=IMPS/2
DEV=/dev/input/mice
EMULATION=yes
EMU_BUTTON2=87
EMU_BUTTON3=88
TRACKPAD_OPT=notap
```

The crucial line is **EMU\_BUTTON3=88**, which defines **F12** as the emulator for the third (ie right) mouse button. I edited this to read **EMU\_BUTTON3=125**, saved the file, and restarted. Booting back to the desktop, I found that it worked. My previously inert Apple key now served as a more comfortable emulated right mouse button.

## Mounting unease

The other thing had been bugging me was how to access to my HFS (ie Mac OS) partitions and the files on them. I felt sure it must be possible to get at

them somehow – and again, a search of the web turned up all the answers.

Skimming another excellent tutorial, this time at <http://linux.org.mt/>, I learned a couple of useful terminal commands, and discovered the purpose of `/mnt` directory. I also dug up a couple of posts from the Yellow Dog mailing list archives which explained everything else I needed.

Yellow Dog 2.3, like most modern PPC distros, comes with HFS support already compiled into the kernel. To mount an HFS partition is therefore no more difficult than issuing:

```
mount -t hfs /dev/hda7 /mnt/macos
```

as root, where **hda7** is the HFS partition you want to access. I chose `/mnt/macos` as the mount point simply because the directory already existed (being the default used by Mac On Linux – see earlier **MOL** box).

With my HFS file system mounted I was then able to **cd** to `/mnt/macos` and **ls** to see my files, including several normally invisible Mac OS system files. Using **cp** I found I was able to copy files back and forth between my Linux and OS 9 file systems with no problems, but I couldn't help wishing for a more intuitive solution.

Fortunately, this was easy. As per instructions I found on the Yellow Dog mailing list (as root) I entered:

```
kwrite /etc/fstab
```

and was presented with the info shown at the bottom of this column.

This, I learned, provides the system with essentially the same information as the manual mount command, but does so at startup so the HFS volume is mounted automatically.

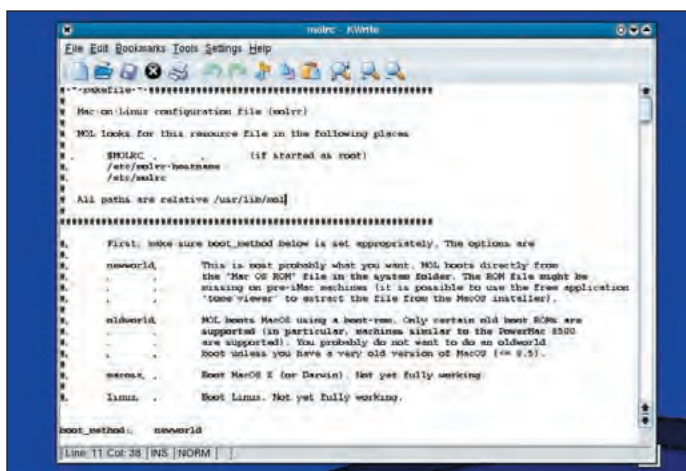
The **rw** in the fourth column specifies read/write access to the HFS file system. Some more experienced PPC Linux veterans warn against writing to your HFS partitions unless

/dev/hda10	/	ext2	defaults	1	1
/dev/hda9	swap	swap	defaults	0	0
none	/proc	proc	defaults	0	0
none	/proc/bus/usb	usbdevfs	defaults	0	0
none	/dev/shm	tmpfs	defaults	0	0
none	/dev/pts	devpts	gid=5,mode=620	0	0
/dev/cdrom	/mnt/cdrom	iso9660	noauto,owner,kudzu,ro	0	0
/dev/hdf4	/mnt/zip100.0	auto	noauto,owner,kudzu	0	0

I added the following line at the end of the file, using tabs to space things out appropriately:

```
/dev/hda7 /mnt/macos hfs rw 0 0
```

Entering `kwrite /etc/fstab` is similar to the manual `mount` command.



*KWrite* can serve as a less-intimidating alternative to *PICO* for UNIX virgins.



This is a paint app called *The GIMP* on the Mac. You may have heard of it.

absolutely necessary. While it's always a possibility that data may become corrupted when one operating system writes to another's file system, the general consensus seems to be that the risk with current PPC Linux distributions is tolerably low. However, if one prefers to err on the side of caution, and doesn't require write access to HFS partition, one can simply replace **rw** with **ro**.

To make things even easier, after a reboot, I navigated to `/mnt/macos` in *Konqueror*, and drag-and-dropped my OS 9 Desktop Folder to the Linux desktop, opting to create a link when prompted. This made it possible to copy files from my Mac OS partition by simply dragging them.

The only fly in the ointment is that, at present, it's not quite so easy to access the HFS+ partitions used by OS X. Some clever utilities available from <http://ftp.penguinppc.org/users/hasi/> make read-only access of HFS+ partitions possible however. **LXF**

## Useful links

Useful links for would-be Mac Linux users

<http://penguinppc.org/>  
[www.powermaclinux.net/](http://www.powermaclinux.net/)  
[www.newtolinux.org.uk/](http://www.newtolinux.org.uk/)  
[www.justlinux.com/](http://www.justlinux.com/)  
[www.tldp.org/](http://www.tldp.org/)

## Platform issues

Linux binaries compiled for Intel processor architectures cannot be run on PowerPC hardware. While some applications are available as PPC binaries, many are not.

However, if the source code for an application is available, it's very often possible for Mac users to compile it themselves. Not every application is guaranteed to compile, and those that do may not always behave predictably. Nevertheless, if there's a particular app you want to try, and the source is available, it's very often worth a go.

# AWARD WINNERS!



No red carpet, no revealing frocks and no boring speeches, but plenty of recognition for a job well done – it's the *Linux Format* awards.

**B**ack in LXF35 we printed the shortlist for the *Linux Format* Awards 2002. The list of nominations came from our website, where many hundreds had put forward

their favourites for selection in the named categories. A few months of frantic voting has followed, but now the virtual ballot boxes have been opened, the votes counted and it's time to announce the winners!

These are, to the best of our knowledge, the only Linux awards nominated and voted for entirely by the readership of a publication, so winning them is a great source of pride, and a small reward for those who have toiled

hard over the year for the good of the Linux community as a whole. And give yourself a pat on the back for taking the time to vote. All we need to round things off are some uncomfortable truths from a famous documentary maker...

## BEST GAME

**It's easy to think that the Linux faithful don't have time to play games, but nothing could be further from the truth – the people that do play games on Linux seem to be up there with the most itchy trigger-fingered fanatics on any platform. In recent years, the demise of Loki has been a blow to Linux gaming**

generally, but the game releases are still coming, both from the Open Source and proprietary sectors. This is borne out by the nominations in this category, which included a little of everything.

In the end there has to be a winner though, and in this category it seems that nothing could stop the

momentum of *Unreal Tournament 2003*. In spite of being a 'stealth' release (there was no mention of Linux compatibility on the box, and the installer is hidden on disk three), *UT2003* managed a massive 39% of the vote. Second place goes to the addictive *Frozen Bubble* game, which managed 29% of overall votes cast.



## BEST SUPPORT RESOURCE

**Finding good, accurate, and reliable support for your various Linux problems isn't as hard as it once was, but it's still worth recognising those who provide these services with an award. Many thanks to you all for the massive number of nominations for *Linux Format* and the LXF website in this category, but we felt uncomfortable putting these through to the voting stage, as the success of our site**

depends equally on the readers as it does the magazine and its staff.

The major distros were represented by their respected support sites and services, and the voting reflected their popularity from the distro category.

Last year's winner became this year's runner-up. The Linux Documentation Project continues to play host to the widest range of HOWTOs available, providing an

indispensable reference resource for all Linux users.

The surprise newcomer in this category romped home to victory. And in some ways, they owe their success to the other nominees here, and everyone in the Linux community who has ever posted an answer to a users question. The overwhelming winners here are Google! With such a wide array of material already available on the

internet, many people are finding that the answer to their question is already out there somewhere, and to get a quick answer without causing any trouble to anyone else, all they need to do is search for it. Google's newsgroup search engine, as well as the main search site make this a quick and surprisingly reliable way of getting the right answers, though of course, it really is just a front end to the resources built by others.

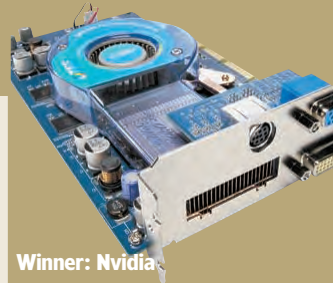


## BEST HARDWARE SUPPORT

**This was a difficult category,** because it meant different things to different people. The nominations seemed to follow the line that proper hardware support meant good drivers for products, and consequently Nvidia, Matrox and others made their presence felt. For others it meant expanding Linux onto new architectures, servers or processors, which seemed to have been behind the support for IBM, Intel, AMD and others. From the beginning, it could have been anyone's award.

In the end the award went to Nvidia, by a large margin (42%). The driver support for its graphics chipsets is pretty good, and using the unified architecture on Linux just makes things easier. Nvidia has come in for some criticism for keeping the drivers closed source, but in the end there are so many ways to install them, and they work so well that it can be (almost) excused by most.

IBM was a worthy runner up in this category, one of the many they received nominations for!



Winner: Nvidia



Runner up: IBM

## BEST SERVER SOFTWARE

**This was a fairly open category in** that many different products and projects could have been, and were, nominated. From web servers to databases and mail servers, nominations poured in, but in fact the main contenders for this category became clear early on.

MySQL has made a huge impact on the web, and are a company transformed since its relatively recent (2000) entry into the Open Source world. While many nominated the PostgreSQL database, it didn't get quite enough votes to make it to the shortlist. *Qmail* and *Postfix* both did though, along with the excellent *Samba*, which has grown in

usefulness and importance in recent times. So many people seemed to depend on *Samba* that it narrowly pipped MySQL to the second place spot with 22% of the vote.

The triumphant winner though, with just 10 votes short of a full 50% share of all the votes cast, was none other than the *Apache HTTP Server, version 2*. An update to the old 1.3.x series was perhaps overdue, but the developers used this as an opportunity to add loads of fundamentally important features. As well as IPv6 support, *Apache 2* works with POSIX thread support and offers a range of engine options to get the best performance from the processor configuration you



Winner: Apache

Runner up: Samba

are running on. A new API and module structure allows for more complex modules and less problems with loading orders. Filters allow multi-pass processing of content streams. As well as all this, it's actually easier to build and configure too.

## BEST LINUX ADVOCACY

**The title of 'Best Linux Advocate'** must surely be an ego boost if nothing else. This is an award which recognises the selfless efforts to promote the Linux OS and all things associated with it.

The nominees this year followed a familiar pattern. Leading light of the Free Software movement Richard M Stallman was a cert for the shortlist, as was kernel veteran and welsh-speaking diary maintainer Alan Cox,

both of whom are pretty active members of the Linux community. Even though he lost his job at HP, Bruce Perens must have taken some solace from his nomination, his showing in the polls no doubt boosted by his DVD-playing antics.

In the end though, this category was narrowly won by a complete newcomer to this category. In a further break from tradition, it isn't an individual either, but a corporation,

and indeed, one not involved even slightly in the creation or promotion of the Linux OS. Quite the reverse in fact, for their name is Microsoft.

Nominated for the compelling speeches by management figures (Balmer *et al*), the real reason for their popularity in this category seems to have been the new licensing model, which aims to deliver a friendly and flexible but regular and reliable way to transfer

## BEST DEVELOPMENT TOOL

**For many Linux users, the real** strength of the operating system is in its rich development environment. A plethora of tools, IDEs and languages compete for the coders attention, so this was always going to be a category where people made their opinions known. Excellent IDEs such as *Anjuta*, *Kylix* and *Kdevelop* had a very good showing, particularly the latter, which has shown tremendous improvement over the last year (and *Gideon* looks pretty good too).

The virtually indispensable *autoconf/automake* tools also had a fairly strong showing in the polls. In the end though, the award went to the particular development tool without which the others probably wouldn't exist – the *GCC* compiler suite – which won't be a surprise to many *Linux Format* readers.

It has been a year of transition for *GCC*, as it moved towards the 3.2 base. This wasn't without problems for some distribution vendors, but now we're at the other side, and the enhanced features and performance have been well worth the effort. *GCC* deservedly romped this category, with 40% of the vote.



**Thanks for all your hard work Bill and co. Same time next year?!**

customers wealth to the Redmond-based company. With 38% of the vote, Microsoft is Linux's best advocate for 2002.

# Awards2002

## BEST DESKTOP APPLICATION

**This is another category where** thousands of software projects could have been nominated, from window managers to the sort of cool apps that make it worthwhile running Linux.

*Fluxbox* made the shortlist, but all the others were, maybe unsurprisingly, graphics and multimedia app. *XMMS* may have been around for some time but it still manages to lead the pack for small, easy-to-use audio players. *Mplayer* and *Xine* both featured in the shortlist and also polled strongly, with the nod going to *Mplayer* at the moment in the battle of the media players. Both *Xine* and *Mplayer* have had a lot of development during 2002, and have really excelled in supporting, in a flexible way, the myriad of formats and codecs in use.

The winner in this category, with 37% of the vote though, is *The GIMP*. In many ways, this has been one of the 'flagship' free software applications. For as long as anyone can remember



Winner: *The GIMP*

(well, at least for as long as we can, your mileage may vary), *The GIMP* has been hailed as an example of how you don't need a proprietary OS and software to be able to 'do real stuff'. Although it isn't everyone's idea of a well-rounded piece of desktop software, there is no denying its features and robustness, which has pretty much dominated the graphics creation arena to the extent that there are few competitors. Development continues, and the latest developer release of the GIMP includes new dialogs using a tree view structure.

## BEST OFFICE SOFTWARE

**At first glance this may seem to** be a narrow category, but we encouraged and received nominations for any app used as part of a corporate desktop environment. The main office suites won out in the shortlist, but in the company of *Evolution*, which certainly lived up to some of its early hype.

This category also saw one of the still rare proprietary entries in the form of *Crossover Office*. Based on Codeweavers implementation of *Wine*, the *Crossover Office* has been a great boon for those who want to switch to Linux but have a desperate need for some *MS Office* functions and compatibility not yet provided on Linux (eg macro-intensive spreadsheets, or custom *Access* databases).

The loose confederation of GNOME office applications, which includes the excellent *Gnumeric*, and the venerable but still functionally incomplete *KOffice* struggled for

**OpenOffice.org 1.0**

Winner: *OpenOffice.org*

second place, though they both lost out eventually, after a few recounts, to *Evolution*, which achieved the worthy runner up spot in this category.

The outright winner probably won't come as a big surprise. We say that because the great majority of you voted for it, in one of the best represented categories in the whole awards (less than 10% of voters declined to pick an option in this category). In the end it achieved a massive 69% of all the votes cast, which only goes to show how much you all like *OpenOffice.org*. *OOo* as we like to call it, has rocketed from the newcomer spot to the most popular office software in a very short space of time. It has done this mainly by being very good indeed.

## BEST ISP OR HOST

**This is a new category for the** awards this time around so pretty much anything could have happened. There were a huge range of nominations here, and we accepted hosting services, resellers and ISPs for nomination in this category. There were so many different nominations with similar levels of support that we extended the shortlist in this category to cover the most popular eight.

Being a good ISP is one of those areas of life where often you'll find if you are doing a good job, people just don't notice you. On the other hand, when they run into difficulties, they expect rapid and positive results. For Linux users, they also expect their ISP to be aware of Linux issues, and to be able to provide at least some level support.

While there was ultimately a winner in this category, it wasn't a conclusive victory, with the vote split

almost equally between the top four. Polling slightly lower than the main group were Telewest, possibly due to the limited extent of their coverage; perhaps not surprisingly, given the recent capping controversy, NTL weren't even among those ISPs nominated. Speakeasy, Rackspace, and Freeserve all received fairly even shares, with Demon and UKLinux.net just inching ahead. In the end the nod went to UKLinux.net with 20% overall, but there really were only a handful of votes in it.



Winner: *UKLinux.net*

## BEST INTERNET SOFTWARE

**This category was slightly** adjusted from previous years to allow for more than just web browsers, but unsurprisingly they still dominated the nominations and the voting. The exception was the *GAIM* instant messaging client, which had a pretty good showing in the voting, given that it was up against much more widely used software.

Four browsers dominated the voting though. Towards the rear of the field were *Opera* and *Galeon*. *Opera* had a better showing than last year in this category, and certainly seems to be getting more popular with Linux users. We expect the main reason for choosing other browsers here is that *Opera* is not Open Source. *Galeon*, the lightweight reworking of *Mozilla* also had a strong showing.

*Konqueror* has improved by leaps and bounds over the last year, and is sure to increase in popularity again this year thanks to the great



Winner: *Mozilla*

improvements in *KDE 3.1*. As a file-manager and web browser it manages a degree of integration with the desktop environment (understandably) far in advance of anything you could expect from the others in this category. With 24% of the vote, it was the worthy runner up.

The runaway winner in this category was, as last year, *Mozilla* itself. Since the 1.0 release, the developers have proved that they still have the desire and ability to build on top of this base and provide more features and greater usability to a demanding audience. We're sure that with the release of 1.3, we'll probably be seeing *Mozilla* in next year's awards too.



## BEST TEA/COFFEE

**Apparently you can't be a proper** Linux hacker without consuming a lot of hot caffeine-based beverages. This category was originally Best Coffee, but we had to change it due to the sheer number of nominations for types of tea (that's Linux users for you). This was a pretty close race, with PG Tips putting in a good showing for the leaf-based drinks and ending up

with a very respectable 21% of the vote. Inching ahead of them were two competing coffee brands. Lavazza beaten into second place by Cafédirect, the Fairtrade coffee. We suspect that their ideology of paying more money to the (often impoverished) bean growers struck a chord with Linux idealists, who gave them their winning 25% of the vote.



Find out more about Fair Trade coffee, tea and cocoa at [www.cafedirect.co.uk](http://www.cafedirect.co.uk)

## BEST DISTRIBUTION

**Another category that attracts a** large response, and a lot of discussion throughout the rest of the year. As the fundamental differences between distributions are quite small, this usually comes down to personal preference. Is the installer easy to use? Are the configuration tools comprehensive? Does it have the best possible support for your hardware?

Although Slackware and Libranet did get a reasonable showing in the polls, this was pretty much a four horse race between the main Linux distributions – Red Hat, SuSE, Mandrake and Debian.

Debian has long been a favourite amongst the LXF team and seems to have picked up quite a lot of support in the last year. The fact that it is based on a charitable organisation

structure may explain why Debian has a very different way of approaching the market. Different package support (once you get the hang of apt, it is hard to live without) and a rather different approach to the speed of new releases are the main differentiators here. It is said to be harder to get to grips with than most, but it just managed to pip SuSE for third place and wasn't that far behind Red Hat.

The overall winner here was Mandrake. Very much at the leading/bleeding edge of the Linux scene, Mandrake has always tried to include the very latest available software and release regular new versions. But it isn't all about the package selection, but also the effort that goes into the installer and tools that make Mandrake stand out.



Winner:  
Mandrake

## EMBEDDED AWARD

**The embedded arena may often** be overlooked, but it's one where Linux has made stunning inroads in the last few years. This award should go to those who have done the most to promote and adapt Linux in this demanding marketplace. Last year this award was dominated by actual devices, whereas this year the nominations have leant more to the underlying technologies, with the likes of Montavista, uClinux and Bayonne being represented. The awards were pretty close amongst the runners up,

Winner:  
Trolltech Qtopia

but a clear winner emerged. Providing the power behind the Sharp Zaurus, and extending their reach further into the embedded market, Trolltech's Qtopia environment has proved a real winner, not the least because it allows Qt code to be easily repurposed on the smallest of devices. We'll have more of Qtopia in a future issue, but for the meantime, well done Trolltech!



## ENTERPRISE AWARD

**The enterprise award is a new** category this year, to recognise the achievements of those pushing Linux as an Enterprise platform. This can take many different forms, from development of software, contributions to the kernel or just sheer marketing muscle applied to this area. Nominees here included Intel, HP, Sun and even

United Linux, but the outright winner, with its fingers in many pies and possibly the biggest force pushing Linux forward in high-end enterprise was, according to your votes, IBM. From its facilities partnerships, development work, code contributions, and initiatives in promoting Linux on 'Big Iron', it is thoroughly deserved.

## FREE SOFTWARE PROJECT OF THE YEAR

**The final category is surely the** most prestigious. Entry requirements are simple – which free software project was the best? It's no surprise that the shortlist contained nominees and winners from other categories, and some nominees from last year.

GNOME and Wine both featured, but GNOME will probably fare better in the next awards if the great features of GNOME 2.2 are anything to go by. Apache 2 also put in a spirited performance. The winner of our Internet category, Mozilla, got third place overall with just over 18% of the total votes cast in this category (which as you might expect was the).

OpenOffice.org, ended up a fairly close second place. Consistent good performance in polls for this software just underlines how good it is, and how much work has gone into it.



Winner: KDE

The winner though, for the second year in a row, was KDE. The project involves a lot of developers working on a lot of different projects. From docs and artwork to solid C coding, from applets and utilities to complex apps, the KDE project encompasses a lot of code and a lot of effort. This probably explains why it just keeps getting better in all sorts of ways, and as you have proved with your votes, that is the opinion of the Linux community as a whole. Congratulations to KDE, and to all our winners! May you all be inspired to surpass yourselves once again over the next 12 months.

# What on Earth is... EFI?

For more than twenty years, PCs have relied on a BIOS to boot up and initialise hardware. Is it time for a change? asks **Paul Hudson**. Intel certainly thinks so...

» I rather like my BIOS as it is. Isn't this a case of "fix it till it's broke" that you told us not to do in your Opinion this month?

Well spotted, and that's true to some extent. However, let us run you through a 'true-to-life' example scenario:

**User:** "Hi there. My games are running kinda slow, and a friend told me it was because my AGP Aperture was too low. She said I need to use my B-I-O-S to make the changes? How can I do that?"

**Tech support person:** "OK. First, reboot your computer. You know when your screen is black and lots of funny text appears on the screen? Wait till it beeps, then press the F2 key on your keyboard. A blue screen will pop up, and you need to use the cursor keys with the + and - keys on your keyboard (although it might be page up and page down on your model) to navigate to AGP Aperture and change it to 128MB. Then hit F10 to save your settings and reboot again."

**User (despondent):** ".....uhm, actually, I guess the games aren't running that slow after all. Thanks anyway. 'Bye!"

In these days of high-res graphics, plug-and-play devices, and, ahem, 'artificial intelligence', many believe that BIOS has seen its prime and is now well and truly over the hill.

While breaking perfectly good systems is clearly wrong, the PC BIOS has been around for a very long time, and hasn't really advanced much since it was invented in 1981. For example, my AMI BIOS beeps once for a DRAM refresh failure, twice for a parity circuit failure, all the way up to beeping 11 times for an external cache error.

A proposed replacement for BIOS is EFI.

» I see. But how could the European Forest Institute replace my BIOS?

EFI, at least in the computing sense, has little to do with oaks, larches, and elms! Instead, it stands for Extensible Firmware Interface, and it defines a wholly new model for interfacing between operating systems and hardware.

You might think of EFI as being a tiny operating system in its own right. It allows users to boot into a high-resolution, high-colour environment designed to be much friendlier – and much more advanced – than today's VGA BIOS.

As it's such a technological leap forward, Intel have taken the opportunity to offload a lot of the

fluff that has accumulated in PC-compatible BIOSs for the past 20 years, which has allowed them to make far-reaching changes on many fronts. For example, computers back in the early 1980s didn't usually come with hard drives, so all the BIOS information had to be squeezed into the ROM. EFI, however, is stored on its own dedicated hard drive partition, so there's a great deal more room available for online documentation, graphics, drivers and anything else your startup requires.

» That sounds pretty straightforward – what's all the fuss about?

Well, the new look is only part of EFI. Perhaps the most drastic change in EFI is the removal of real-mode operation. Modern operating systems run in what is known as *protected mode*. Protected mode systems are able to use virtual memory, multi-task, use extended memory (>640K), and other such critical things.

However, protected mode was only introduced into CPUs with the 286 processor, which was released in 1982 – after the release of the BIOS – which meant that the gremlin we know as real mode was already in widespread use before protected mode started to be used. Furthermore, because DOS was stuck using real mode for most tasks, real mode continued to be in widespread use up until the early 1990s.

Even today, operating systems still may need to make real mode calls to the BIOS, which are incredibly slow because the CPU literally has to be taken out of protected mode to execute the instructions, then switched back into protected mode to continue.

» Crikey! Get back to the easier stuff!

Well, as we said, EFI provides the capability to have a much more flexible interface presented to users; the whole real mode / protected mode thing is transparent to users.

Graphically speaking, EFI BIOSs have a point-and-click, mouse-operated interface that allows users to configure options, select which OS to launch. One sample EFI BIOS even had an MP3 player built-in!

» Cool! So it's like a pretty front-end to a BIOS?



Yes, and no – it's much, much more than that, although certainly many users won't see anything beyond the more attractive user interface.

For example, EFI systems are designed to have basic networking capabilities built into the system itself so that remote diagnostics can be performed. In fact, one of the goals of EFI is to have systems that are 'plug and play' in the most advanced sense. For example, a new server with no operating system installed on it could just be plugged into a server room somewhere without a monitor, and have the base operating system installed over the network using EFI.

» **Ah, but involving networking must mean drivers, and having drivers at the BIOS level must surely be very complicated...?**

Not at all! In fact, EFI was designed to be very flexible with regards to drivers, and utilises a special language known as EFI Byte Code (EBC) so that drivers are cross-platform.

Put simply, the drivers are written in various languages, then compiled down to EBC so that they are cross-platform. These drivers are then interpreted when they are called during system start-up, and, potentially, an IA-64 driver will work just as smoothly on an IA-32 platform. At least, that's the *plan*...

» **Does that mean, then, that EFI is available on several architectures?**

Yes, but as Intel are leading the drive towards EFI, it's only currently available on Intel platforms. So far, that's the Itanium architecture (IA-64), x86 (IA-32), and XScale architecture. However, the specification for EFI is freely available online for anyone to read, and hopefully more hardware manufacturers should be making use of it as time goes by.

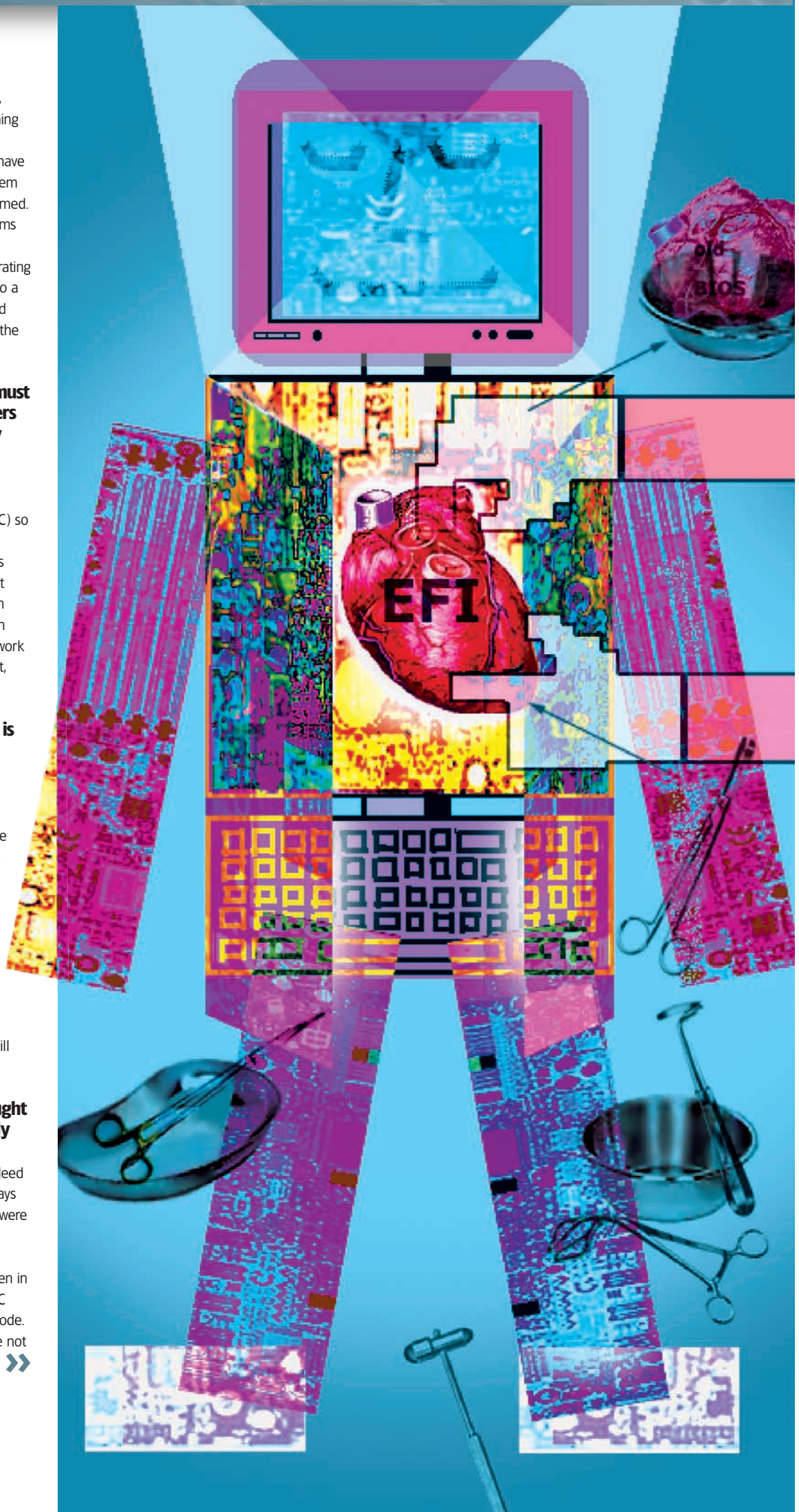
If you didn't know already, XScale is Intel's ARM v5 PDA CPU, and that alone demonstrates how flexible EFI can be.

The key thing, though, is that it's common source code throughout – the same C code will compile and run on every platform.

» **Wait a minute – C code? I thought BIOSs were written in assembly for maximum speed?**

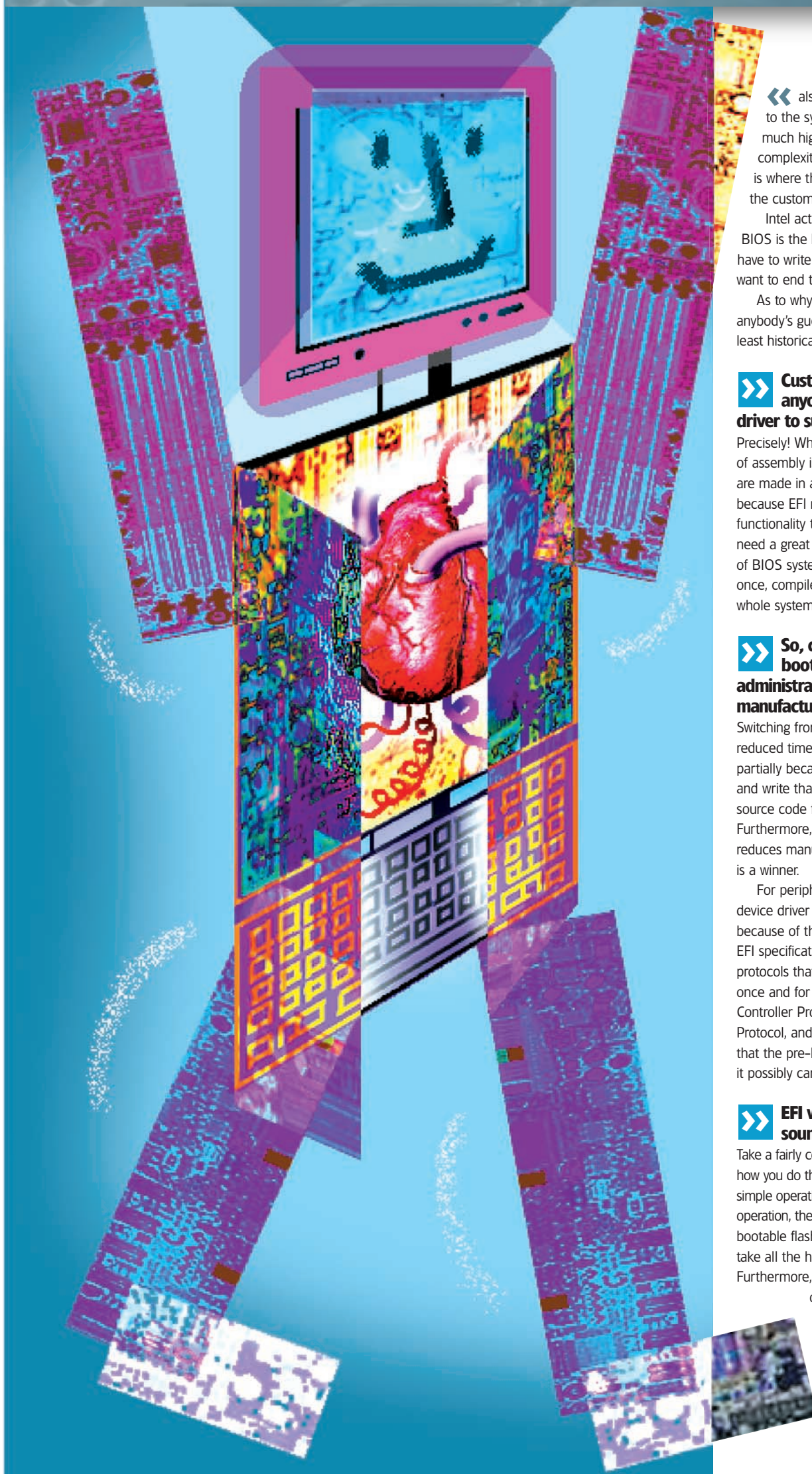
The current generation of boot loaders are indeed written in assembly because back in the old days of computing, processing power and memory were severely limited and every effort was made to minimise program complexity. Furthermore, C compilers have come on leaps and bounds even in the past 10 years, and very often well-written C code will outperform hand-written assembly code.

By dropping assembly source files, Intel are not only making EFI cross-platform, but they are »





# WhatOnEarthEFI



« also adding a large performance boost to the system. Furthermore, because C is a much higher-level language, the overall complexity of the system can be increased: this is where the high-res graphics come in, and also the custom device driver modules.

Intel actually said it best themselves, "The BIOS is the last place on the PC where people have to write in low-level assembler code, and we want to end that".

As to why C was chosen in place of C++, it's anybody's guess. However, C code generally (at least historically) has a speed advantage over C++.

## » Custom modules? You mean anyone can write an EFI device driver to suit their own needs?

Precisely! While a few people still believe that use of assembly is I337, most realise that fewer errors are made in a higher-level language. Furthermore, because EFI replaces nearly all of the BIOS functionality that currently exists, new developers need a great deal less training in the archaic ways of BIOS systems. Instead, they are taught write-once, compile-anywhere techniques that make the whole system more flexible and streamlined.

## » So, customers get a more friendly boot interface and also remote administration. What's in it for manufacturers?

Switching from assembly to C allows greatly reduced time to manufacturing for suppliers. This is partially because C code is much easier to read and write than assembly, but also because the source code for EFI is shared across platforms. Furthermore, having an easier-to-use system reduces manufacturers' support calls, so everyone is a winner.

For peripheral manufacturers, the chore of device driver writing will be greatly simplified because of the EFI Byte Code. Furthermore, the EFI specification defines a collection of device protocols that can be used to unify PC devices once and for all. For example, there's a USB Host Controller Protocol, a Universal Graphics Adaptor Protocol, and a Simple Pointer Protocol – all so that the pre-boot environment is as streamlined as it possibly can be.

## » EFI will be that easy to use? It sounds too good to be true!

Take a fairly common task: flashing your BIOS. Think how you do that. Usually it requires booting up into a simple operating system, coaxing your CD drive into operation, then running a risky ROM patch. With EFI, bootable flash update CDs can be provided that take all the hard work – and all the risk – away. Furthermore, because the EFI data itself is stored on a hard drive, backups can be made in case of problems.



## »» How come no one has thought of this before?

Well, that's not entirely true. The OpenFirmware initiative [www.openfirmware.org](http://www.openfirmware.org) contains a lot of the same functionality as EFI: pluggable, machine-independent drivers, network protocols, and self-test diagnostics.

The key difference is that EFI is backed by some very large companies indeed, which has triggered very quick adoption at the enterprise level. Indeed, EFI is mandatory for Itanium and Itanium 2 systems.

Furthermore, EFI is much more wide-ranging than OpenFirmware. You can think of OpenFirmware as being an evolution of the BIOS, whereas EFI is much more of a *revolution*.

## »» What else can it do?

Because EFI is much more advanced than existing BIOS solutions, and also because it can have third-party drivers tacked on with little fuss, it's possible to use EFI to solve operating system problems. For example, let's say a Windows box you're administering crashes (because we all know that Linux doesn't crash, [cough]) and becomes unable to boot. Using EFI and the correct tools, you could correct the problem on the Windows partition without the need to boot up.

Alternatively, thanks to the network connectivity available, you could temporarily give control of your system over to remote engineers to allow them to solve the problem. Intel calls this 'the afterlife'.

A clever feature of EFI, which really demonstrates how much more advanced EFI is when compared to BIOS, is that EFI allows you to run a simple command prompt shell window to perform command-line tasks on your computer *inside* the pre-boot environment. Naturally it's not designed for usage beyond running diagnostic tools, but that may change.

## »» Now for the most important question: Does it work on Linux?

Yes! EFI has had excellent support for Linux for over a year now, particularly on the IA-64 kernel port. Furthermore, adoption of EFI will actually help Linux a great deal, because, owing to the cross-platform drivers that EFI uses, Linux could potentially use precisely the same drivers as Windows. Whether that's a good thing or not is entirely subjective!

Intel have really been pushing development of EFI for some time, and have computers working smoothly with both IA-64 Linux and Windows XP. Although there are no plans to fully integrate EFI in the 2.6 kernel, at this rate it won't be long before other Linux ports join the Itanium port.

## »» I keep hearing 'Itanium' and 'Intel'. Is EFI Intel-only?

Well, as we said, that's how it is currently. EFI was designed to promote a uniform pre-boot

## Open Firmware explained

### Yesterday's machine-independent BIOS

Open Firmware is the name given to the IEEE 1275 Standard for Boot Firmware: Core Requirements and Practices, and is a specification for a machine-independent BIOS designed to find and configure compliant hardware. Since its launch, IEEE 1275 has been officially withdrawn by the IEEE, and has no real official support today.

However, there is – or perhaps *was* – a great deal of community support behind a group of developers who formed the Open Firmware Working Group with the aim of continuing development of the standard and creating practical implementations for mainstream use.

I say that there perhaps *only was* support for Open Firmware, because the support has slowly died out as time has gone by. There *are* still people out there who fervently believe it's the way forward, but, for example, the 'What's New?' section of the Open Firmware website is dated 1999 – hardly up-to-date! Open Firmware was a great idea, with backing from Sun and Apple, but it's quite possible that it has had its day now that EFI is approaching a critical mass popularity.

While it's possible that Open Firmware and EFI may duke it out in the near future, it's likely to be a short fight – especially with EFI's continued heavyweight backing from Intel.

environment for Intel platforms, but the wide-ranging scope of the system allows EFI to be ported wherever it is needed. Given that Itanium and XScale processors both support EFI and are pretty much polar opposites of each other, anything is possible!

Remember that Itanium was a joint initiative between Intel and Hewlett-Packard. Some of the lead engineers on EFI work for HP, so by no means is EFI limited to one company.

## »» Will it run as fast as existing BIOSs?

In many cases, EFI will perform much, much better than current BIOSs. However, once you're booted into your OS, this shouldn't be noticeable anyway as Linux will still do most of the hard work itself.

With regard to the speed that new features and functionality can be programmed for EFI, the worst case scenario predicted by Intel is that EFI will be three times faster.

## »» So, is that "it" for BIOS, then?

Not necessarily. In fact, to support backwards compatibility for those 'just in case' scenarios, EFI boards may include a legacy BIOS subset locked deep away from the OS. This option would provide the best mix of new functionality and backwards compatibility.

## »» Where can I learn more?

Intel have a large amount of documentation on their website, at [www.intel.com/technology/efi](http://www.intel.com/technology/efi), and you can also find the full EFI specification there as well. Also on the same site, Intel have released a full sample implementation of EFI that is compatible with IA-32 and Itanium.

If you're interested in developing for EFI, Intel produce an open-source EFI Application Toolkit that includes a complete TCP/IP implementation as an EFI protocol, a remote debugger, and more.

Before you charge off to read the EFI specification, one small note: EFI 1.10 (the current

specification) is a weighty 1084 pages at the time of writing this article. Concise it *isn't*, but we'd recommend prospective developers get a copy.

## »» I'm a treeware fan: what's available about EFI that I can get from my local bookstore?

*IA-64 Linux Kernel: Design and Implementation* by David Mosberger, Stéphane Eranian, and Bruce Perens (Prentice Hall PTR ISBN 0-13-061014-3) is an excellent book on IA-64 and also kernel internals. Not only does it make very easy reading, but it also is *the* authoritative reference on how the Itanium processor family interacts with Linux. If you own a copy already, turn to page 426 to start reading about EFI. If you don't already own it, what are you waiting for?

## »» So, can you sum up EFI in one sentence for me?

Put simply, EFI is the result of adding accepted best practices to the problem of booting and managing the low-level aspects of a computer. That's EFI in a nutshell. **LXF**

## BIOS and you

### Growing old gracefully...

BIOS has been around for an awfully long time, at least as far as other components of computers go, so why has it taken so long to change it?

Well, the problem is that there is an exceptionally large amount of legacy code out there that relies on BIOS being there, particularly code that relies on real mode CPU operation. As a result, it's taking a long time for such old code to die out and therefore no longer require supporting in a BIOS replacement. Furthermore, some parts of computers are just so ingrained into our collective consciousness, that to change them would require a big mental step as well as a big technological step – look how long it's taking for the floppy disk to fade into obscurity!

While BIOS will certainly be used for the next few years at the least, momentum is building up to replace it, and I sincerely doubt that it will be missed.

# Tutorials >>

Our experts offer help and opinions on a whole host of Linux applications

## Your guide to getting things done!

Whether you are just starting out in Linux, or an experienced veteran, there's always more to learn. Every issue of *Linux Format* is packed full of practical advice, and nowhere is it more concentrated than in our tutorials pages.

Here you'll find expert guides to all sorts of things, from basic Linux usage to understanding and deploying network solutions, from simple script coding to the complexities of Perl regular expressions, Java server apps and more. We aim to bring a good mix of tutorials to each issue, but if you have any suggestions for topics you'd like us to cover, why not contact us, by post, by email ([linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)) or log on to our website and post your suggestions in our forums? ([www.linuxformat.co.uk](http://www.linuxformat.co.uk)). Hope to hear from you soon!

**Nick Veitch** EDITOR

## THIS MONTH TEACH YOURSELF...

### Mozilla

Package up your bespoke browsers so that others can appreciate all your hard work – plus more handy tips **p64**

### GIMP

Layer Masks – how to make 'undoable' non-destructive changes to your images **p68**

### PHP

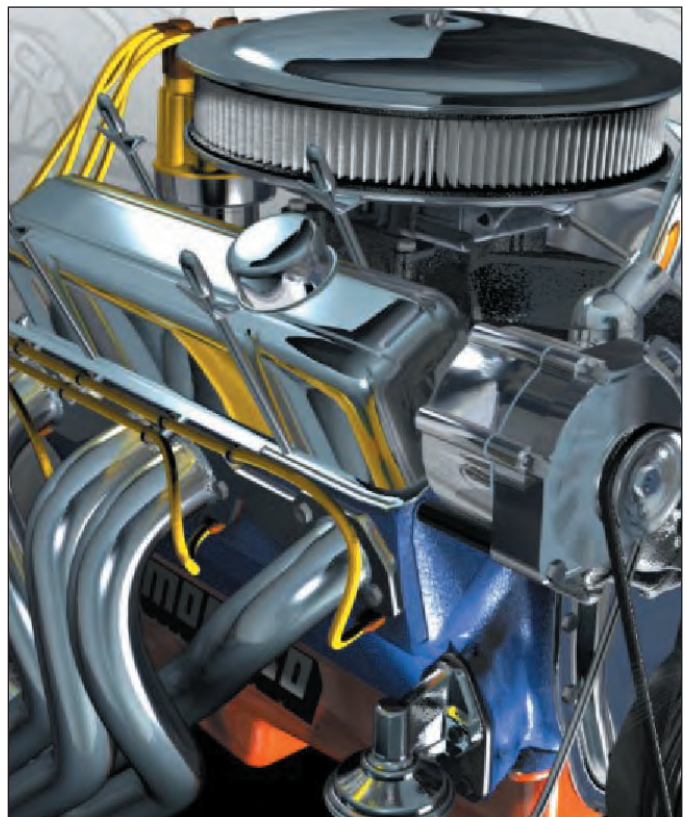
EIGHT PAGES to introduce you to the concept of using PHP to create your own GUI **p72**

### Python

Assigning individual variable names to each object is impractical – so why not use containers? **p80**

### Blender >>

NEW SERIES! Whether stills, game building, or animation are your bag, this newly open-sourced app is for you! **p84**



## How code is represented

Including code in magazines can be tricky, but we hope our notation will help it become clear. When lines are too long for our columns, the remaining text appears on the next line in a solid blue box:

```
procedure
TfrmTextEditor.mniWordWrapClick
(Sender: TObject);
otherwise, there is usually a gap
between lines:
begin
mniWordWrap.Checked := not
end;
Usually, you'll find the code on
our CD/DVD too.
```

## TIP OF THE MONTH!

Ever find that some background tasks make your computer run slower than you can handle? Don't worry: help is at hand with this month's tip.

With the command **nice**, you're able to launch a task with a higher or lower priority than it would otherwise have had. As an example, running **nice -n 10 top** would mean that **top** would be launched at quite a low priority, and hence be given fewer chances to take up CPU time. The **-n** parameter is the adjustment you want to make to the process' starting priority, and it can range from **-20** (launch the

## Playing nice-ly

program with the very highest priority) to **19** (launch with the lowest priority).

Through the use of **nice**'s close companion, **renice**, you're also able to modify the priority of processes that are already running. For example, running:

```
renice 5 -p 828 830
```

would cause the **nice** value for processes **828** and **830** to be adjusted so that they have a lower scheduling priority. You can also use **renice** with the **-u** parameter to specify a user for whom you wish to alter all processes, for example:

```
renice 30 -u hudzilla
```

would set all of user **hudzilla**'s process to priority **30** (very low; great for timeslice hogs!).

Note that only root may **renice** jobs it does not own, whereas non-root users are limited in the extent they may **nice** in order that they aren't allowed to chew up inordinate amounts of CPU time – this is particularly important on shared systems. Note that non-root users are only allowed to **renice** in a negative or downward direction – they can't force their processes to take up more CPU time, only less.





## BESPOKE BROWSERS

# Leader of the pack?

The final stage in the creative process is packaging everything up for others to install. **Andy Channelle** also looks at the mail filtering features in Mozilla 1.3 and an old trick that you may have missed...



Over the past four months, we have covered the basics of designing new skins for the *Mozilla* browser and creating very simple Internet-centric apps. While the results have not been spectacular, our endeavours at least have exposed some of the methods available to control the user interface and some basic input and output options. There is, of course, an awful lot you can do without needing recourse to 'proper' programming; a quick visit to [www.mozdev.org](http://www.mozdev.org), which hosts a vast range of disparate projects, should convince any sceptic of the quality of *Mozilla* as a development environment. This work, though, will count for nothing if we do not eventually provide a way for other users to install our creations. We need to make it distributable.

There are two stages to this process: firstly all the elements of the package need to be archived into the correct directory structure, then we have to write a short installation script. The latter will invoke *Mozilla's* Cross-Platform Installer (XPI – apparently pronounced “zippy”) from a link on a web page and integrate the application/skin into *Mozilla's* structure. As the name implies, XPI works across the platforms supported by *Mozilla*.

## Simple Application

Packaging applications and skins increases in complexity as the project grows, but at its heart, it simply involves ensuring that all the elements are archived, in the correct format, where the *Mozilla* browser on a potential user's machine expects them to be. Our directory structure will look like this:

```
/dictoolbar
/content
files
```

The first release of our Dictionary.com toolbar is incredibly simple and the resulting package – which is based on a single XUL file and one graphic – requires little work. As work continues on it though, the Javascript will be pulled into a separate file to be called from the XUL document. We may also have Cascading Style Sheet (CSS) information and further graphic files to include at a later date. In addition to the XUL file, we also need to create **contents.rdf** which will add the toolbar to the main package list and register the the XUL file as an overlay to be integrated into the main navigator.xul file. As expected with any XML implementation, we first need to define the namespace, in this case we need to define both RDF and *Mozilla's* Chrome namespace:

```
<RDF:RDF xmlns:RDF="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:chrome="http://www.mozilla.org/rdf/chrome#">
```

The next section begins with a standard **RDF:Seq** element which adds the toolbar package to *Mozilla's* root list. The second element provides a little more information for the RDF:Description

```
<RDF:Seq about="urn:mozilla:package:root">
  <RDF:li resource="urn:mozilla:package:dictoolbar"/>
</RDF:Seq>

<RDF:Description about="urn:mozilla:package:dictoolbar"
  chrome:displayName="Dictionary Toolbar"
  chrome:author="andy channelle"
  chrome:name="dictoolbar">
</RDF:Description>
```

Finally we add code to join our toolbar to the overlays sequence and explicitly tell *Mozilla* that it needs to be overlaid onto navigator.xul:

```
<RDF:Seq about="urn:mozilla:overlays">
<RDF:li resource="chrome://navigator/content/navigator.xul"/>
</RDF:Seq>

<RDF:Seq about="chrome://navigator/content/navigator.xul">
<RDF:li>chrome://dictoolbar/content/dictoolbarOverlay.xul</RDF:li>
</RDF:Seq>

</RDF:RDF>
```

We save this plain text file as **contents.rdf** in the same directory – /dictoolbar/content/ – as our main files.

The next stage is to archive the 'content' subdirectory into dictoolbar.jar. The /dictoolbar directory now contains a single file, which in turn contains our distributable toolbar.

## Installer

The next task is creating the installation script. This is simply a short Javascript that can be executed from a link on a web page. This will be called **install.js** and will be saved inside the /dictoolbar directory.

For the sake of reusability it makes sense to add a couple of user variables at the start of the script which can then be called later in the script. Remember that the variables must match the names in **contents.rdf**.

```
const myProductName = "dictoolbar";
const myProductRegKey = "/Mozilla/dictoolbar";
const myProductRegVersion = "0.1.0";
const myJarFileName = "dictoolbar.jar";
```

Now we have the script which actually does the work. The script itself uses the **initInstall()** command to initialise the installation and, if needed, the first section can be appended with the line **logComment("initInstall: " + err);** to create a log of the installation in *Mozilla's* top level directory:

```
var err = initInstall(myProductName, myProductRegKey,
myProductRegVersion);
```

The script now finds the default Chrome folder using **getFolder()**, sets it as the default and then copies over our jar file using **addFile()**:

```
fChrome = getFolder("Chrome");
setPackageFolder(fChrome);

err = addFile(myJarFileName)
logComment("addFile() returned: " + err);
```

Finally, the **registerChrome()** element adds our XUL file to *Mozilla's* navigator.xul and then, if there are no errors the installation is performed. If an error is encountered the installation gets cancelled.

```
regErr = registerChrome(PACKAGE | DELAYED_CHROME,
getFolder(fChrome,myJarFileName), "content/");
logComment("regChrome returned: " + regErr);

if (0 == getLastError())
performInstall();
else
cancelInstall(err);
```

The last piece of the jigsaw puzzle is archive procedure. The contents of /dictoolbar need to be zipped up and then renamed to replace the .zip extension with .xpi. So we now have a single .xpi file, containing both install.js and dictoolbar.jar, which can be uploaded onto a web server for installation.

Now all we need to do is create a link to let the masses install our toolbar. The simplest way to do this is to upload the XPI file to your web space and add a standard hyperlink to the file from a web page. A user clicks the link to invoke the installer, a dialog will pop up to confirm the installation and the files will be copied across.

You can of course launch the installer with a smattering of Javascript. Add the following to the head of your document:

```
<script>
function XPIInstall(file, title) {
var xpi = new Object();
xpi[title] = file;
InstallTrigger.install(xpi);
}
</script>
```

and then call the script with:

```
<a href="javascript:XPIInstall('dictoolbar/dictoolbar.xpi'
'Dictionary Toolbar');">Link</a>
```

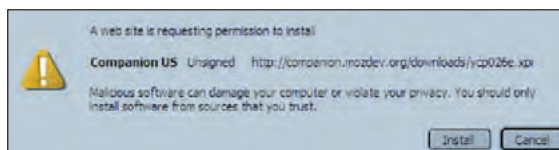
XPI is a powerful system capable of much more than just copying and registering files; it can also upgrade or patch existing installs and manage, change or monitor the *Mozilla* file system. Most of XPI is, as expected, cross-platform but there are a few platform specific elements which, for instance, allow you to create objects which interact with the Windows Registry or create Mac Aliases. On a non-supported system these simply return **null** and allow the script to continue.

In addition to digesting the contents of the XPI manual (<http://devedge.netscape.com/library/manuals/2001/xpinstall/1.0/xpinstall.html>) we've found examining the installation scripts of other projects available at [www.mozdev.org](http://www.mozdev.org) very instructive, especially that of the Companion Toolbar (<http://companion.mozdev.org/>) which has similar aims, though with a Yahoo! bias, to what we've been attempting.

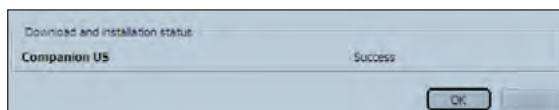
Larger applications such as *XUL Maker*, a visual XUL application designer, or *LiveLizard* have far more complex directory structures to take into account the separation of display and data. Briefly, these larger applications divide files between the content, locale and skin directories which are, in turn, contained in an application specific sub-directory below /Chrome. The content directory is for Javascript, XUL, CSS and contents.rdf, it is also where you would put any HTML files, such as the 'about' page and the license, associated with the project. The Skin directory is used for storing graphical elements such as icons, throbbers, backgrounds, etc, while the locale directory takes care of localisation issues and, occasionally, a properties manifest.

## Automation tools

The whole computer industry has been characterised over the last 20 years by a process of automation, so it would be insane if there was no way to automate the creation of XPI files. The state of the art when it comes to XUL development is a collection of

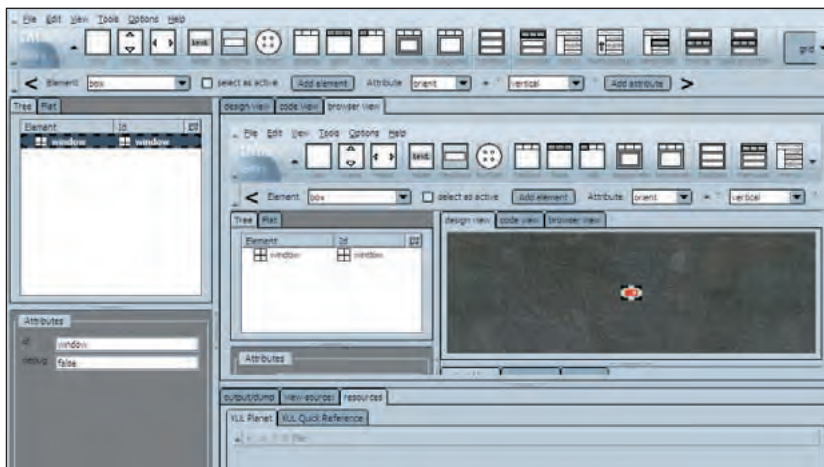


**A simple installation script invokes the XPI installer.**





# TutorialMozilla



**XULMaker has a go at rendering its own XUL file.**

◀ Perl scripts called *XULKit*, available from [www.hacksrus.com/~ginda/xulkit/doc/](http://www.hacksrus.com/~ginda/xulkit/doc/). The core of *XULKit* covers the beginning and end of *Mozilla* application development.

*New-from-template.pl* is a script which creates a source tree for a new application. Users provide a name and the script does the rest, providing a menu overlay, XUL window with a selection of menus, and makefiles – both Windows and Linux – that can be checked in to *Mozilla*'s extensions subdirectory. The second script is called *makexpi.pl* and is a totally self-contained programme that creates jar and XPI files from your source tree, automating much of the process of making your application distributable. It uses an installfile variable, in tandem with the included **install.js** template to complete the process.

Finally, in *XULKit*, is a useful script called *build-example-app.pl*. This takes a selection of user provided answers, covering areas such as application name and version number, and then creates a sample application which can be edited manually in the normal fashion and packaged using *makexpi.pl*. This is a very good way of experimenting. While working with *XULKit* isn't intuitive, the results are useable across all versions of *Mozilla*.

A more visible approach to XUL development is the big idea behind *XULMaker*, a project which has lain in development hell for the past year but has recently been picked up by Franklin de Graaf. The current release is definitely Alpha software and it works better as a tool for creating rather than editing existing applications so far. The

next release is slated for the first quarter of 2003 and should feature a full implementation of XUL meaning all elements and attributes will be available. It would also be desirable to integrate the functionality of *XULKit*'s XPI building script, but this doesn't seem to feature on the project's roadmap at present.

*Chameleon* is/was an Open Source theme development tool created from Netscape's early *Themebuilder* tool. Like *XULMaker*, development has been stalled for some time and the 'current' version only works with *Mozilla* 0.8. This is a shame as the app had an intuitive user interface, and a comprehensive themeing tool is vital to bring *Mozilla* customisation within the grasp of users without the time or inclination to develop programming skills.

## Mozilla for users

As we've been at pains to point out, *Mozilla* is not just a development system. In fact, unlikely as it may seem, some users have no intention of fiddling around with the insides of their browser! Fortunately it also has some really cool enhancements for general browsing including keywords, bookmarklets and, as of the 1.3 series, basic filtering in the email client.

Bookmarklets are very small Javascript programs that can be dragged onto the browser's Personal Bar and, once clicked on, act locally on the page you are viewing. The origins of the toolbar we've created started as a bookmarklet which enabled us to search for highlighted words at Dictionary.com, but there is plenty more you can do. Bookmarklets were actually part of the original Javascript implementation in *Netscape 4* and have since been adopted by both *Opera* and *Internet Explorer*. As a result, many of the scripts are usable across all three browsers, but due to the vagaries of the various Javascript implementations, there are a few platform-specific elements.

To create a bookmarklet, select Bookmarks>Manage Bookmarks, highlight the Personal Toolbar Folder and do File>New>Bookmark. Give your creation a name and input your Javascript. Try some of these:

**SEARCHING** Ever wondered what a website is using to serve up pages? Add this bookmarklet to check via *Netscape*.

```
javascript:Qr=document.getSelection();if(!Qr){void(Qr=prompt('
Site...(e.g. - www.domain.com)'))};if(Qr)location.href=
'http://www.netcraft.com/cgi-bin/Survey/whats?host='+escape
(Qr)+'&port=80
```

## Moz-Tech 101: RDF

### Data model and syntax

The Resource Document Framework (RDF) is a W3C web standard *Mozilla* uses to integrate data and metadata such as bookmarks, profiles, sidebar tabs and other ostensibly user controlled (or controllable) data into the browser suite. The standard itself divides into two elements, the Data Model and Syntax. The former is a 'labelled-directed' data graph, meaning every node and arc in the graph is identifiable and all the arcs go one way. The Data Model itself is subdivided again into three object types:

All things described by **RDF expressions** – which could be anything from a single text element on a web page to the page itself or even an entire site – are called resources. Resources are named using a Universal Resource Identifier (URI). According to the W3C standard, URIs are not restricted to elements that are directly accessible on the web: a book, person or "any entity imaginable" can be assigned a URI.

**Properties** are the attributes that are used to define a resource. Each property has a precise meaning and sets its own permitted values, the kind of resources it can describe and its connections with other properties.

A resource combined with a property and its value makes up the third object type, an **RDF statement**. In the statement, the resource becomes the subject, the property becomes the predicate and the value equals the object. In a further twist, the object can be another resource or a literal, which is a resource (specified by a URI), simple string or other primitive datatype defined by XML.

The sentence "Nick Veitch is the editor of Linux Format" can be broken down as follows:

**Subject** = Linux Format

**Predicate** = Editor

**Object** = Nick Veitch

A directed-labelled graph for that sentence would look like the diagram on the right, where nodes (ovals)

represent resources and arcs represent named properties. Rectangles represent the nodes for string literals (our object). The arrow will always travel from the subject to the object of the statement.

While the Data Model covers the definition and use of metadata, its presentation is defined using **RDF Syntax**. Unlike standard XML languages, **RDF** does not conform to the tree-like formation you may expect and the same data can be represented in more ways than one. For instance, it is possible to structure child and parent elements in a number of ways in the same document, but retain the same meaning. You could think of it as being a little like learning Latin where word order is often subordinate to the relationships between verbs, nouns, prepositions, etc.



## Web links

RDF Specification: <http://www.w3.org/RDF/>  
 Information on RDF Properties:  
<http://www.w3.org/TR/1998/WD-rdf-schema/#core>  
 Mozilla's use of RDF: <http://www.mozilla.org/rdf/doc/>

If you have followed the earlier tutorials you will have noticed that this script uses **document.getSelection()** which we used previously to search for highlighted words with Dictionary.com and Google. It is also possible to create a link which uses Google to reveal pages that contain links to the currently displayed page.

```
javascript:void(window.location.href='http://www.google.com/search?q=link:'+window.location.href)
```

One of the most useful bookmarklets we have found for Mozilla fans is called *Buglinkify* at [www.squarefree.com/bookmarklets/mozilla.html](http://www.squarefree.com/bookmarklets/mozilla.html). It was created by Jesse Rudderman and when invoked transforms numbers on a page, such as 29346, into links to the relevant *Mozilla* bug.

**VISUAL** If you're having trouble reading the yellow text on an orange background, this bit of code will transform the page colour to white and the text to black as well as normalising link colours. Of course, you can set the colours exactly as you'd like to see them.

```
javascript:(function(){var newSS, styles=" { background: white !important; color: black !important } :link, :link * { color: #0000EE !important } :visited, :visited * { color: #551A8B !important }"; if(document.createStyleSheet) { document.createStyleSheet("javascript:" + styles + ""); } else { newSS=document.createElement('link'); newSS.rel='stylesheet'; newSS.href='data:text/css;' + escape(styles); document.documentElement.childNodes[0].appendChild(newSS); } })();
```

As before you may find some of these elements, such as the **!important** declaration, familiar from previous tutorials.

**USEABILITY** The scope of Javascript for improving UI usability in web pages is limited, but you can, for instance, have a bookmarklet which will automatically scroll a page for you:

```
javascript:var wN2scRl;Sa5gNA9k=new Function('clearTimeout(wN2scRl)');document.onkeydown=Sa5gNA9k;Sa5gNA9k();void(wN2scRl=setInterval('if(pageYOffset<document.height-innerHeight){window.scrollBy(0,1)}else{Sa5gNA9k();50}'))
```

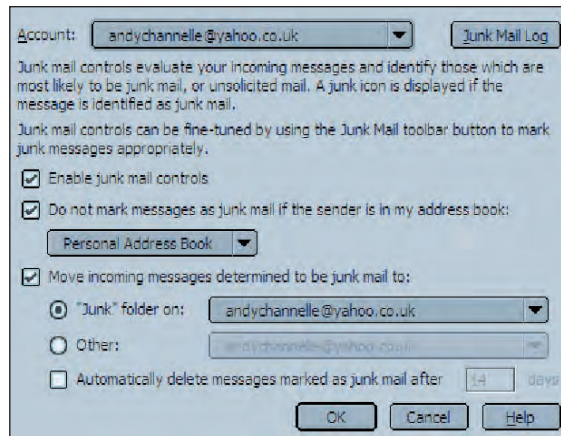
You can change the speed of the scroll by adjusting the **scrollBy()** variable or, if you want to manually select the scroll speed you can:

```
javascript:var wN2scRl;S3g=new Function('clearTimeout(wN2scRl)');document.onkeydown=S3g;a7T=prompt('Rate...');if(a7T!=null){S3g();void(wN2scRl=setInterval('if(pageYOffset<document.height-innerHeight){scrollBy(0,a7T)}else{S3g();50}'))}else{void(null)}
```

Clicking on this bookmarklet will launch a dialog. Input a number between 1-8 (1 is slowest) and the page begins to scroll. On both of these scripts, hitting any key will stop the movement.

And finally, this bookmarklet offers a little extra space and a run button for the creation of new bookmarklets.

```
javascript:W7=open('A';width=320,height=240,resizable');W7.focus();with(W7.document){write('<center><form><textarea name=X rows=10 cols=34 wrap>javascript:</textarea><p><input type=button value=Run onclick=opener.location=X.value>');void(close())}
```



The filtering controls work across every mail account you set up in *Mozilla*.

It is surprising that more fuss isn't made of bookmarklets as they can be very useful in day-to-day browsing. Visit [www.bookmarklets.com](http://www.bookmarklets.com) for more information and lots of examples that you can use.

One of the things that *Mozilla* has been really good at in the past is the suppression of Internet annoyances, most notably in the form of popup advertisements, which can be shut off unless a user explicitly requests them to be displayed. The latest stable release (1.3 – get it from this month's *Linux Format* coverdiscs) boasts an addition which may help us all in the war against spam. Bayesian filtering is a statistical approach to the problem which examines your mail and discards those that fit a defined pattern based on a "Bayesian combination of the spam probabilities of individual words."

Like popup choking, spam filtering is not enabled by default, but it is not hard to get it going. Bayesian filtering relies on you first 'training' it to recognise spam. You can either select a range of emails, using <Ctrl> to make multiple selections and select Tools>Mark Selected Messages as Junk or hit the Junk button on the toolbar.

Once that's done, you need to enable mail filtering with Tools>Junk Mail Controls... and select enable. Incoming mail will now be classified according to your efforts. Your mileage may be improved by having suspected spam moved to a junk folder, you can then review it and correct any errors, which obviously impacts on future filtering. If you don't want to move the spam around, but still don't want to see it, select the 'view' filter just above the message pane and select 'Not Junk'.

For more on the process and potential of Bayesian filtering, see <http://www.paulgraham.com/spam.html>. [LXF](#)

Bayesian filtering relies on you telling the processor what is and isn't junk.





## TutorialGIMP



## NON-DESTRUCTIVE EDITING

# The GIMP Layer Masks

**PART 3** Layer masks allow you to undo changes made to previously saved work because they don't remove any content. **Michael J Hammel** shows us how to use them to overcome common compositing problems.

Two of the most frustrating issues that digital artists face in their day-to-day work are the need to undo changes made to previously saved work; and the desire to modify colour content with simple brush strokes. For the former, artists turn to the *GIMP*'s layer masks to generate non-destructive changes to layers. For the latter, they look to *GIMP*'s blend modes, which we'll peek at today and cover with a more in-depth tutorial in next month's *Linux Format*.

Layer masks provide a simple method of removing pixels in a layer from the composited image displayed in the Canvas window – without actually removing the pixel content! The mask is a grayscale image that shows as a thumbnail to the right of the layer thumbnail in the Layers, Channels and Paths window. Black pixels in the mask block the relative pixels in the layer from being used in the composite image. White pixels in the mask allow layer pixels to be used. Grey pixels in the mask provide transparency to

their relative cousins in the layer – the darker grey the pixel in the mask, the more transparent the pixel in the layer. By using layer masks you can modify a layer without removing content, thereby allowing you to return to your layered artwork at a later time and, with a modification to the mask, retrieve portions of a layer currently unused. The use of layer masks is known as non-destructive editing because you don't actually remove pixel data from the layer. The pixel data is still there, which can be returned to later if necessary; the mask just specifies if it will be used and, if used, how transparent it will be.

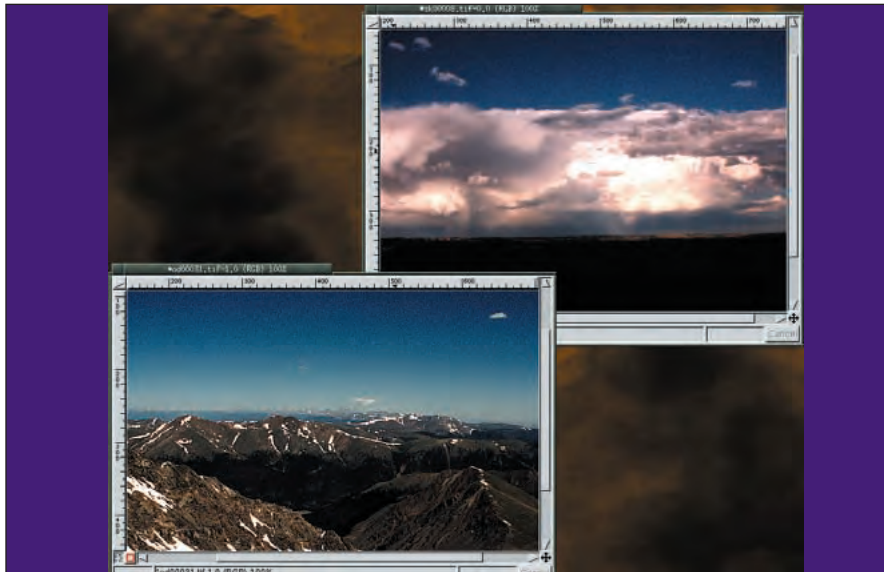
Blend modes are available in many tools, and in the case of the layer blend modes, can also be used for non-destructive editing. A blend mode is a method describing how a pixel will change when composited with another pixel. In the case of layers, blend modes define how a higher level layer will be combined with the next layer down.

One use of layer blend modes is to produce a colour negative from an image by placing a white layer on top of the image and setting the white layer's blend mode to Difference. The colour negative can then be desaturated and used to create a complex selection or Layer mask. We'll start this month's tutorial by showing how to use Layer masks to perform simple merges of two layers and more complex shape masking.

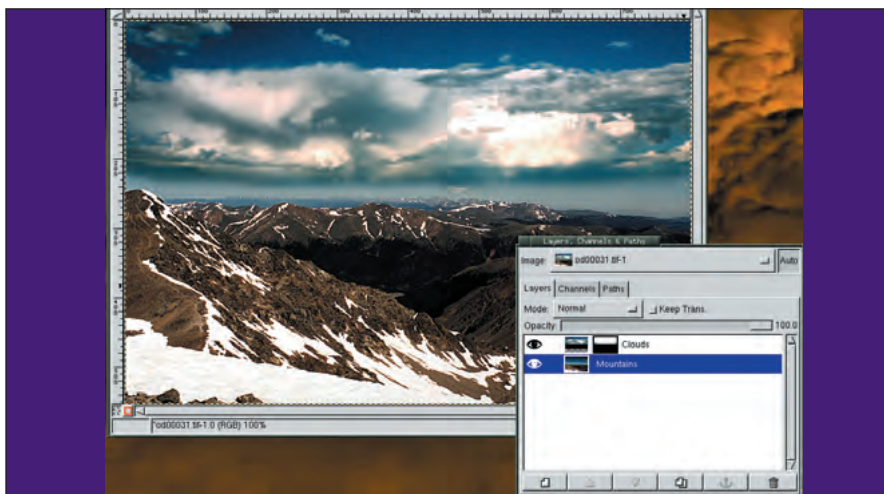


## MERGING TWO LAYERS

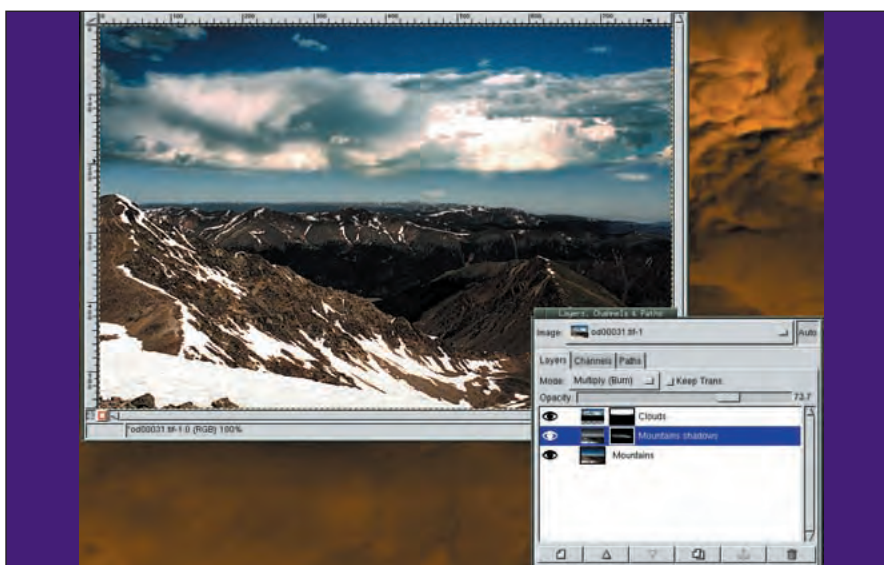
A common use for layer masks is to seamlessly merge two layers. The effect allows you to apply part of the top layer as a replacement for a portion of the bottom. In this example, we'll be adding a cloud filled sky to a skyless mountain range.



**1** Our two images for this example: The first is a photo of a beautiful Colorado mountain range with a clear blue sky; but that sky is a little boring. Our other image shows an interesting cloud pattern over a dull, nearly dark earth. We'll take the clouds from the latter and place them over the empty sky in the former. Use **CTRL-A** to select the entire cloud image, then **CTRL-C** to make a copy of it. Move to the mountain image and use **CTRL-V** to paste the cloud image into it.



**2** Move the opacity slider to the left so we can see the tops of the mountain range better. The clouds are a little low compared to the top of the mountain ranges so we've moved them up a little using the **Move** tool. We've made a freehand selection around the cloud, selecting everything above the clouds and just a little below them. Note that you can click and drag within the image and, with the mouse button still held down, drag outside of the window, around the top and back into the window on the other side. When you release the mouse button, a straight line is drawn to connect the start and end points of the selection. Add a layer mask to the cloud layer that is completely white. Make sure the mask is active by clicking on it (its thumbnail in the Layers dialog should be enclosed in a white border). The selection is then inverted, heavily feathered, and filled with black. Finally, the opacity slider is returned to its rightmost position.



**3** The layer mask has allowed us to easily merge the two images without modifying either layer directly. The mask isn't perfect, however. We still need to touch up the mask on the bottom side of the clouds to remove the dark edges. We can do this by zooming in and using a soft edged brush and the **Paintbrush** or **Airbrush** tools. Also, the cloud layer may need to be colour adjusted so it looks like it came from the same lighting as the mountain range. In our example, we've removed much of the red, magenta and yellow from the image using the **Image>Colors>Colour Balance** tool. The two small lower clouds in the mountain image were selected using the **Elliptical Selection**, feathered lightly and blurred with a **Gaussian Blur** so the appeared more distant and at the same focal depth as the new clouds. Finally, a desaturated version of the mountains, with a mask of the background mountains only, was used to add some shadows where the clouds were added.



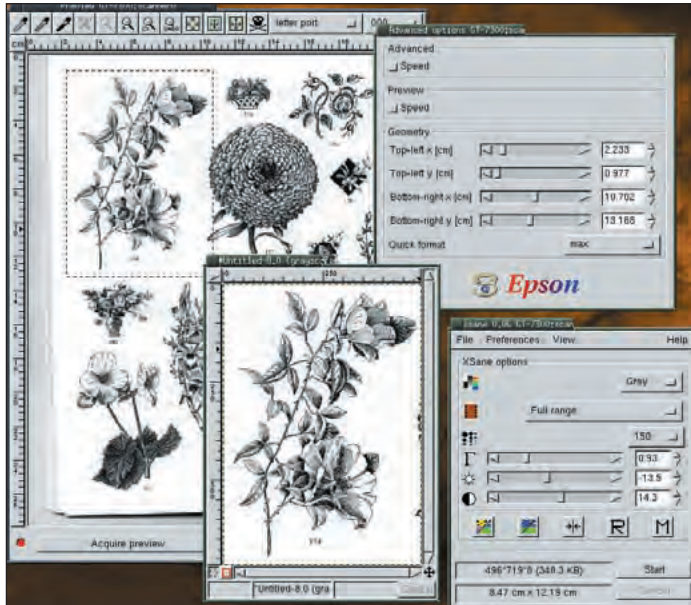


## « ISOLATING COMPLEX SHAPES

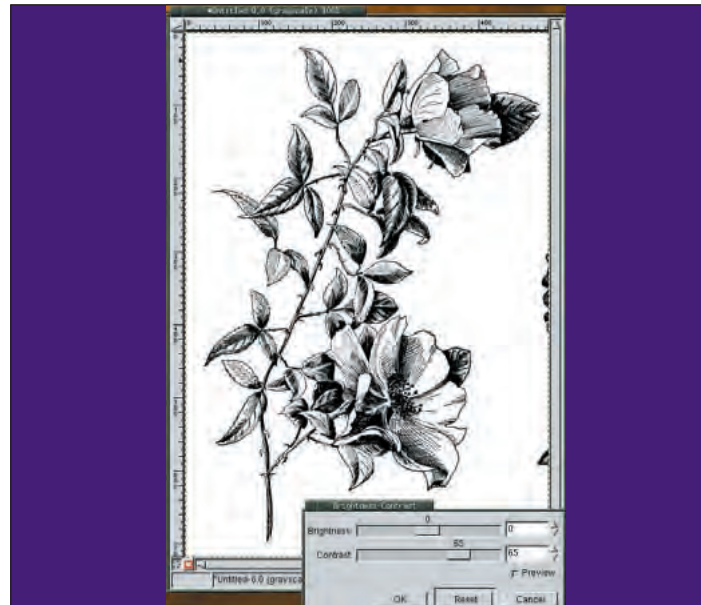
A common question from *GIMP* users is how to make selections of complex shapes in order to paste them into another image. The trick isn't to try and use the generalized selection tools like the Freehand or Bezier tools, but to use a mask to grab the shape's intricate details. In this example we'll take an image of a

flower and stem from a book of clipart and scan it, place it over a solid background and colorize it.

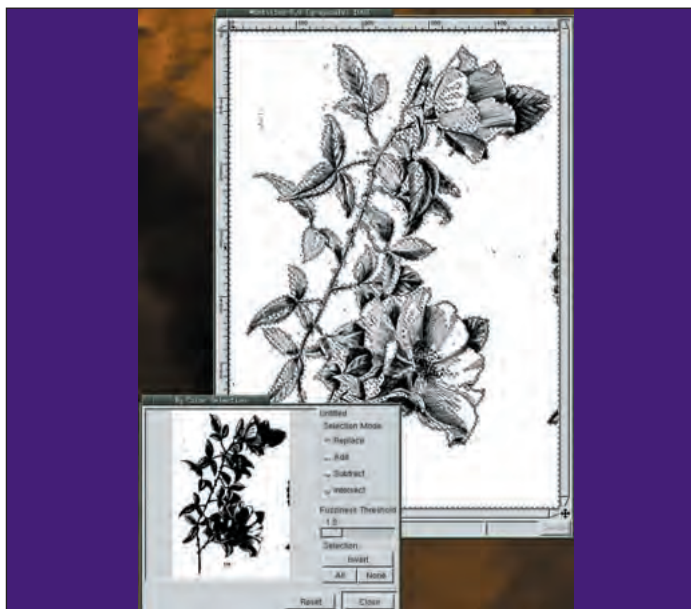
We create an image large enough to hold our flower and fill it with a colourful background. Put this image aside for now. We won't be using it until our last step.



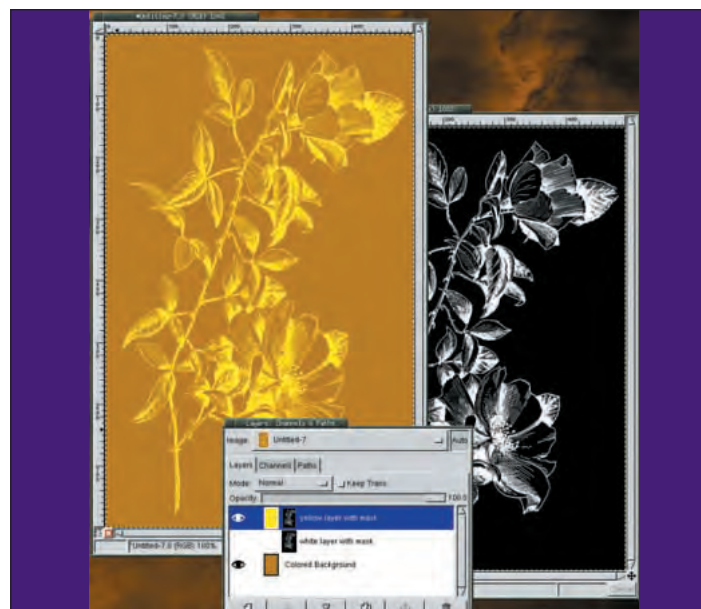
**1** We start with a scanned version of some clipart. We're using the KOWA Epson backend driver to XSane with an Epson Perfection 1260 here. The scanning software allows us to select a section of the page, but we can't completely isolate the flower we want. We'll have to remove the extra bits using the *GIMP* – this is a technique used every day by professional digital artists. We've scanned the image in grayscale. We could scan in black and white only, but this would remove much of the detail in the flower.



**2** The scanned image of this complex shape is not completely black and white. We first run *Image>Colors>Levels* and select *Auto* to make sure we get a full range from absolute white to absolute black. But we still have some non-white pixels (though we probably can't visually see them) in the area surrounding the flower. We'll use *Image>Colors>Brightness Contrast* to get rid of most of these by increasing the contrast a bit. Too much contrast change will cause us to start losing details in the flower, however.



**3** To rid any non-white artifacts from the area around the flower, we'll use *Select by Colour (Select>By Colour...)* with a 1 pixel Fuzziness threshold and clicking on the white area of the image. This shows us small regions of non-white pixels, including the area on the right that came from another flower in our scan. We invert the selection and paint over those areas with white. Deselect the image and repeat the *Select By Colour*/paint process until we're satisfied.



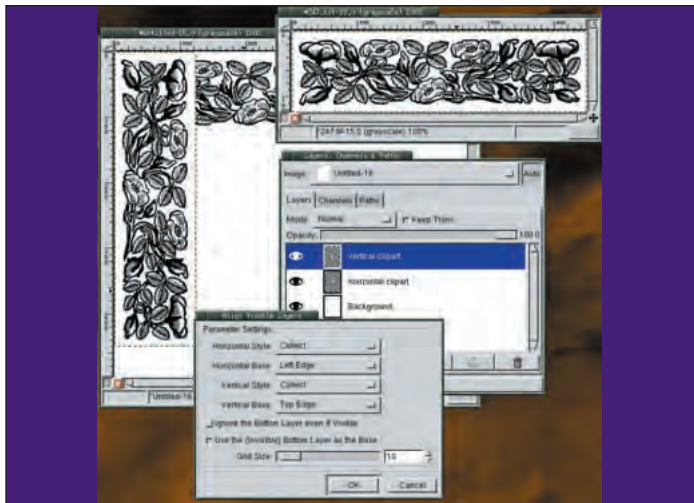
**4** Invert the image (*Image>Colors>Invert*). Add a white layer to the coloured image. Create a white mask for the white layer. Copy the flower image and paste it into the mask of the white layer. If the flower doesn't fill that layer, all that you need to do is scale up the floating selection until it does. The complex shape is now a white image over your coloured background. We can fill the white layer with any colour to change the colour of the flower.



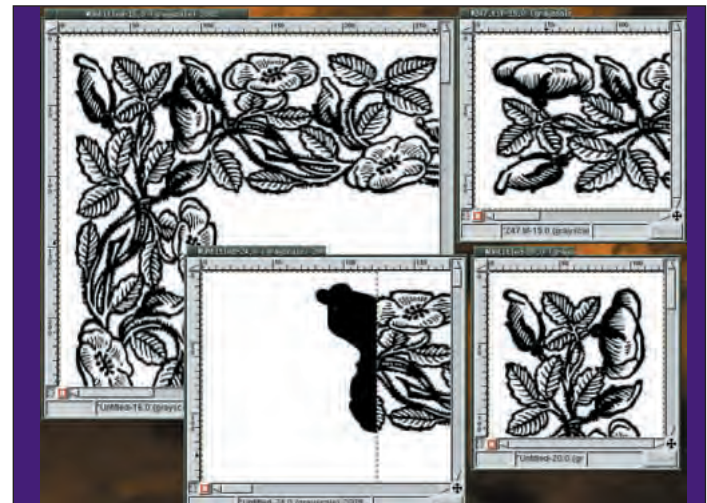
## CREATING A BORDER FROM CLIPART

In this third tutorial we'll be painting inside our masks to meld two complex patterns together to create a border around an 8.5 x 11 inch page. The process is simple but requires some close up work using soft and hard edged brushes to make the merging appear seamless. We start with a white image sized to fit a standard 8.5 x 11 page (the actual size doesn't matter –

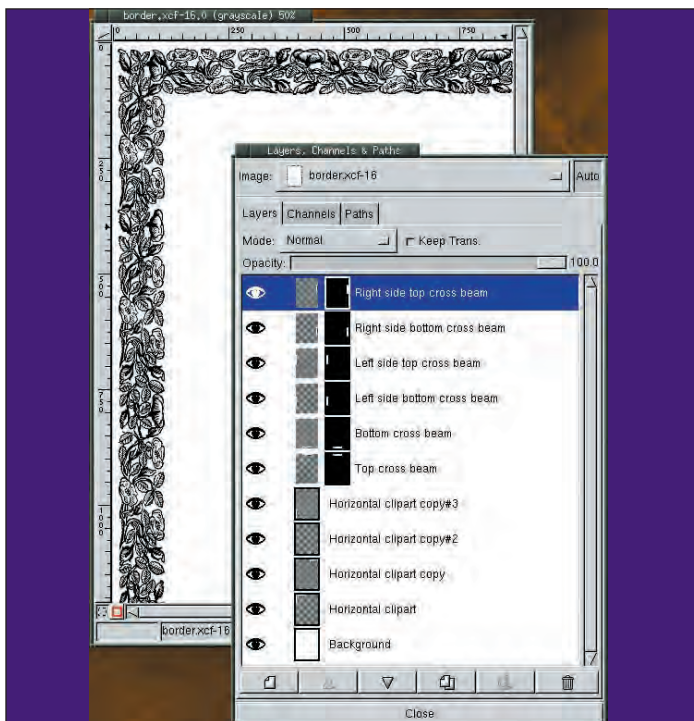
make it any standard size used for printing at home). The white background is used because our source clipart has a white background as well. If the clipart background is not exactly white, this doesn't present too much of a problem.. Simply use the steps in the previous *Isolating Complex Shapes* tutorial to adjust the image colours.



**1** With the clipart sized appropriately to work as a border on the page, copy the clipart (CTRL-C) into image (CTRL-V). Convert the floating selection to a new layer. Duplicate the layer and rotate the duplicate 90 degrees. Use **Layers>Align Visible Layers** to align both copies of the clipart to the upper left corner. Set the Horizontal Style to **Collect**, the Horizontal base to **Left Edge**, the Vertical Style to **Collect** and the Vertical Base to **Left Edge**. Disable the **Ignore Bottom Layer** button (the button should appear raised). Enable the **Use Bottom Layer** button (it should appear depressed). The setting for Grid size is irrelevant for this example. Finally, zoom in on the upper left corner.



**2** Turn off the background layer visibility. Add a white layer mask to the top layer. Select a brush that has a hard edge to it, make sure the foreground colour is black and start to paint in the layer mask. What you're looking for is a form that merges the two layers cleanly. Click on the layer mask while holding the ALT key (Shift-ALT may be necessary depending on your desktop configuration) to view just the layer mask. Use the Opacity slider to see through the top layer to what you might want to show through the mask. Turn the top layer visibility on and off quickly to view what is showing and what could be showing from the other layer. All the while, paint in black in the layer mask to remove parts of the top layer, and paint in white to bring parts back. There is no special trick here except patience and various brush sizes to merge the two pieces together. When you've got the mask set so the two layers composite nicely, merge the visible layers to make a corner piece for your border.



**3** Duplicate the corner piece three times, aligning each copy to one of the other corners of the image, rotating the copy as needed. With the background layer active, copy and paste the clipart again. The floating layer should be centered in the background. Make it a new layer and align it with the top of the background only (set the Horizontal Style to **None** in the **Align Layers** dialog). Create a mask on this new layer and paint it like you did in the previous step to merge it with the corner pieces. Note that if the clipart is not long enough to fill the space between the corner pieces you might need to scale the clipart before you start all this. Alternatively you can paste and align multiple cliparts between the corners and merge them using layer masks. Repeat the process for the other three sides of the image. **LXF**

## NEXT MONTH

Blend modes can lighten dark areas, add colour to lifeless images, or produce unusual affects, all without actually modifying the image. Well-lit images can be enhanced with Multiply and Divide, but what if you have an underexposed or poorly lit image? We'll show you what to do. Next month we'll also cover Screen, Overlay, Behind, Darken, Lighten, Value, Colour, Hue and Saturation



TAILOR-MADE GUIs

# Practical PHP Programming

This month, take your PHP to an all-new level by creating GUI applications. **Paul Hudson** shows you how...



**A**lmost ten issues ago, in issue 30 of this very magazine, two momentous events took place: firstly, it was the first part of *Practical PHP Programming*, and secondly, Charlie Stross wrote an excellent tutorial on how to lever the Tk graphical toolkit to create applications using Perl.

Having covered command-line applications last issue, this issue we're going to move onto a fairly similar topic to Charlie's, with the important exception that we shall, of course, be working with PHP. Yep, you guessed it: the topic of this monster eight-page tutorial is how to create graphical applications using PHP. This is quite a jump from all other sorts of PHP programming, so you may find you need to read through this article a few times to get the hang of things!

## Getting started

In order to create graphical PHP applications, you first need to install the PHP-GTK module – read the box titled *Installing PHP-GTK* for guidance.

PHP-GTK, as can be guessed from the name, is the combination of our favourite programming language and the GTK+ GUI toolkit. GTK, incidentally, stands for *GIMP Tool Kit*, as it was originally developed for use in the *GIMP* software. Since its creation, GTK+ has come a long way and is now used as a central part of GNOME and has been ported to Windows.

This cross-platform ability works perfectly with PHP's cross-platform nature, and the end result is that, as long as care is taken, you can create attractive and powerful applications that run on a wide variety of machines.

## Important warning

Working with PHP-GTK, as already mentioned, is entirely unlike writing PHP in other situations; a solid grasp of object-oriented programming is a must, and also you should be prepared for quite a bit of theory before you get to dig in with the code. I'm not kidding about the OOP requirement!

That being said, I have specifically tried to simplify matters as much as possible, at least to begin with, so that you get to implement cool things using PHP-GTK as quickly as possible.

## Graphical User Interfaces

GUIs form the core computing experience for the majority of users – even many Linux people today find themselves using KDE or GNOME for tasks that only a few years ago would have been done from a shell out of necessity. The key reason for this

is that GUIs are, generally speaking, designed to be easy to use, with the goal of allowing users to spend more time thinking about what they wish to achieve and less about how it actually needs to be achieved.

In order to minimise the learning curve required for new users to get to grips with an environment, GUIs use shared code to implement the graphical components that make it up; for example, the code to generate a toolbar would be the same across all programs written using a given GUI toolkit in order that all the programs share the same look and feel. Several GUI toolkits, including GTK and Trolltech's Qt (used in KDE) are written using C++ classes and objects, which means each graphical element of a program, known in the 'nix world as a *widget*, has its own class which inherits properties and methods from various ancestor classes.

For example, the `GtkButton` classes inherits from the `GtkBin` class, which in turn inherits from the `GktContainer` class, then the `GtkWidget` class, which inherits from the `GtkObject` class. Each sub-class adds new methods and properties to do a specific task for that widget, and if a programmer wished to create a specific kind of `GtkButton` – for example, a button that automatically played a sound when clicked – they would probably find it easiest to inherit from the `GtkButton` class.

After you've been using GTK for some time, you will likely come to appreciate its fine-grained class inheritance structure, as it allows you to create your own objects at all levels, and also provides you with some very powerful objects, like the `GtkCalendar` widget, which provides fairly good calendar functionality just by instantiating the class.

Beyond inheriting class variables ('properties') and class functions ('methods'), GTK widgets also inherit *signals*, which is a core topic in GTK. Simply put, signals are emitted when things happen in your GUI; usually this is the user interacting with your widgets. `GtkButtons`, to take the preceding example, emit a signal when the mouse moves over them, a signal when the mouse clicks them, a signal when the mouse is released from clicking them, and a signal when the mouse moves away from them, amongst other signals.

So each signal, as you can see, is sent out when a particular occurrence happens to a widget, and each widget has its own set of signals that it will emit as a result of user interaction. The magic comes when you tie a given signal, eg 'clicked' for a `GtkButton`, to a function you have written; this function is then called every time your user clicks the button you chose.

All clear on the theory aspect? Excellent; if so, then it's time to demonstrate the first piece of code...

## A Basic GUI

```
<?php
function doshutdown() {
    gtk::main_quit();
}

function btnClick($button) {
    echo "Hello, console!\n";
}

$window =& new GtkWindow();
$window->connect("destroy", "doshutdown");

$button =& new GtkButton("Hello, GTK!");
$button->connect("clicked", "btnClick");

$window->add($button);
$window->show_all();

gtk::main();
?>
```

Our first program, which you should save as **gtk1.php** in your home directory, is simply the "Hello, World!" of PHP-GTK. Yes, I realise that it's twelve lines longer than a simple **echo 'Hello, World!'**; would have been; this is because working with graphical interfaces requires a great deal more work: you must create and assign properties to widgets, set up signal handlers, and more. However, don't be put off by the length of the script – I want to go through it line by line in order to show quite how easy it is.

Ignore the two functions near the beginning for the time being; we will come to them shortly. Beyond them lies **\$window =& new GtkWindow()**, which is probably something quite new to you, depending on your experience with the language.

If you didn't know already, **=&** means 'assign to equals' rather than 'copy to equals'. When **new GtkWindow()** is called, PHP creates a new object of the class **GtkWindow**. If we just had **\$window = new GtkWindow()** then PHP *creates* the **GtkWindow**, then *copies* it into the **\$window** variable. So, behind the scenes, this process involves *two* **GtkWindows**, one that is created using **new** and one that is created using **=**. Using **=&** rather than **=** circumvents this problem; **\$window** becomes a *reference* to the same **GtkWindow()** created by the **new** operator, so the process involves only one **GtkWindow** being created. You're quite able to rewrite the line as **\$window = &new GtkWindow()** if you prefer; it's all down to your coding style.

The class **GtkWindow()** itself is a general-purpose application window, and it descends from the class **GtkBin**. I mention this specifically because descendants of the **GtkBin** container widget are only allowed to have one child widget inside them. For **GtkWindow**, this means that you are only allowed one widget (eg one button, one combo box) in your window, which may sound a little restrictive at first, but all will be revealed later on.

**GtkBin** itself descends from **GtkContainer** – it is a specific kind of container in that it allows only one child widget – and this provides it with **GtkContainer**'s automagic resizing of child widgets. This means that any widget created inside a **GtkBin** will take up all the free space inside the **GtkBin**, and will resize as the **GtkBin** is resized.

On the next line, we again use **=& new** to create a new widget: this time it's a **GtkButton**. When you create your button, you can pass in as a parameter the caption you wish to give the button; that is, what text you wish to appear upon it. In the above example, "Hello, GTK!" is passed for the caption of the button, and internally this is used to create a **GtkLabel** inside the button, to which the caption is assigned – this will become important if you later wish to change the caption of the button to something different.

Again, the **connect()** function is called, this time we connect the 'clicked' signal of our new **GtkButton** to the PHP function **btnClick**. This works in the same way as our previous called to **connect()**.

The **add()** method of our **GtkWindow** is native to all descendants of **GtkContainer**, and works as you would expect: the widget passed in as the first parameter is added to the container and placed where available. In the situation of **GtkWindow**, which, remember, is a descendant of **GtkBin**, this means that the widget being passed (our new **GtkButton**) will automatically take up all free space in the window.

The next line, **\$window->show\_all()** translates to "Show this window, and all its child widgets". An alternate method here is **show()**, which would have displayed the **GtkWindow** but not the **GtkButton**.

Finally, we come to the call to **gtk::main()**. If this line looks a little odd to you, don't worry: it's a remnant of PHP's mantra of "If we want to do something complicated, at least make it look like C++; that way at least *some* people will understand it". Technically speaking, **gtk::main()** is a call to the static member function **main** of class **gtk**. In normal circumstances, one creates an object of a class before calling a function of that class. However, sometimes it's not necessary or indeed it is counter-productive to use an object on some occasions, and so these functions are 'static': always in the same place.



## Installing PHP-GTK

The easiest part of it all

Debian users, type **apt-get install php-gtk** as root, and you're done. For everyone else, here's a step-by-step guide:

- 1) Pop over to [www.gtk.org/download/](http://www.gtk.org/download/) to download and install GTK+ 1.2. You *may* have this already installed for you by your system, depending on your particular distribution
- 2) Download the PHP-GTK source from <http://gtk.php.net/download.php> and extract it locally
- 3) Run from the directory you extracted PHP-GTK into, run **./buildconf**, then **./configure**
- 4) Run **make**, then **make install** to install the extension into your default PHP extension directory


5) Make sure you have the line **extension=php\_gtk.so** in your **php.ini** file

To test your installation, try some of the scripts from the **test** directory. My favourite is **gtk.php**, which shows off a wide selection of widgets to get you started. If you're interested in installing the *Glade* interface builder, then, if you're a Debian user, type **apt-get install glade** as root. Note that the **glade-2** package is for GTK+ 2 development, which is as yet unsupported.

For other distros, grab the source code from <http://glade.gnome.org/download.html>, then run **./configure**, **make**, and **make install** to compile and install the program. You can run *Glade* by simply typing **glade** from the command prompt.



```
paul@alice:~/gtk$ php4 -q first.php
Hello, console!
Hello, console!
Hello, console!
```



**All the hard work is over, and we finally have a simple GUI application. It's all easy from here – sort of...**

◀ If this all seems complicated, don't worry about it – it really is quite complicated, but you needn't understand any more about how it works other than that it's a function call you can use like any other.

What it *does*, though, is quite important, and I'm sorry but it includes *even more* theory! Put simply, owing to the fact that GUIs are signal-based – that is, they wait until told what to do – your control over the program ceases once you've finished creating your GUI and performing any startup tasks. Instead of running everything yourself, control gets passed onto GTK, which enters what's known as its *message loop*, which internally looks something like this:

```
while (1) {
    if user moves over widget {
        if widget has signal handler set {
            send signal to program that mouse is over widget
        } else {
            do nothing
        }
    }

    if user terminates program {
        send signal to program that it is being killed
        break out of while(1) loop
    }

    ...
}
```

Granted, that's quite a big simplification, however you should get the gist: GTK gets control of your GUI and does all the processing for it such as resizing buttons, dropping down combo boxes, highlighting buttons when the mouse is over, etc, and only passes control back to you by sending a signal which in turn calls your signal handler functions. While these functions are running, you are back in control, and you may run all the PHP code you like, including calling other functions. However control will eventually be passed back to GTK, again leaving you waiting for a signal to be passed.

So, calling **gtk::main()** instructs GTK that you're done setting up, and that you're ready for it to take control. Now, onto our two functions: **doshutdown()** and **btnClick()**.

**doshutdown()** was passed to our first call to **connect()** to tie it to the signal **destroy**. The end result of this is that when the **GtkWindow \$window** is being destroyed, that is, closed by the user, it will be sent the event **destroy** by GTK causing it to emit the signal **destroy**, which in turn will call the **doshutdown()** function. **doshutdown()** has just one line inside it: a call to another static member function, **gtk::main\_quit()**.

**gtk::main\_quit()** only has any use when **gtk::main()** has been called, because it instructs GTK that you're ready to exit its message loop and resume control of the application. Generally speaking, this is the end of your application, as control is passed back to you after **gtk::main\_quit()** has been called, PHP

continues to execute any code that lies after your original **gtk::main()** call.

Our **btnClick()** function was tied to the 'clicked' signal of our **GtkButton**, and so it will be called every time a user clicks **Hello, GTK!**. The 'clicked' signal is often sent part-way between two other signals, 'pressed' and 'released', which correspond to a mouse button being pressed on a button and a mouse button being released on a button. The 'clicked' signal is also sent when the button is activated by way of the keyboard (pressing **Enter**, etc), as is common in many programs.

Inside **btnClick()**, we again only have one line, which calls some relatively simple PHP. However, it's important to realise that calling **echo** from inside your PHP-GTK scripts allows you to write to the console just as easily as if this were a standard PHP script. A helpful bonus is that one signal can execute several functions simply by calling **connect()** multiple times with varying second parameters.

There you have it! Although it may seem like an awful lot of explanation is required for what is actually quite a short piece of script, the large percentage has been the theory behind how the script works. Go ahead and run your script with the following command line:

```
php4 -q gtk1.php
```

If your PHP CGI is named differently, you shall need to make the appropriate change. However, as long as you have installed PHP-GTK correctly, the end result is the same – all being well, you should see something similar to screenshot 1, above left. **-q**, as you may know, is 'quiet mode' for the PHP CGI, and it forces PHP not to emit its standard HTTP headers. You can exclude this parameter if you are using the PHP CLI SAPI.

## Multiple Windows

Once you are feeling confident with the whole situation so far, you can move onto a slightly more complicated script – here's the source code:

```
<?php

function doshutdown() {
    gtk::main_quit();
}

function btnClick($button) {
    $window =& new GtkWindow();
    $window->set_title("Spawning Windows 2");
    $window->set_default_size(300, 100);

    $label =& new GtkLabel("This is a new window");

    $window->add($label);
    $window->connect("destroy", "doshutdown");
    $window->show_all();

    return false;
}

$window =& new GtkWindow();
$window->set_title("Spawning Windows 1");
$window->set_default_size(300, 100);
$window->set_border_width(10);
$window->connect("destroy", "doshutdown");
```

## Get more out of GTK

Online resources to make your GTK code fly

The PHP-GTK documentation itself is a good enough place for some pieces of information, but, to be honest, it's rather weak on the whole; large chunks of code are left unexplained, various function calls are entirely undocumented, etc. However, it's a good place to start, so take a look at <http://gtk.php.net/manual/en/>.

<http://gtk.miester.org> is a fairly good website with regards to PHP-GTK, but there isn't much there, yet. This isn't terribly surprising because the GTK interaction with PHP is still working its way into the community. However, we anticipate this site will continue to grow healthily over time.

There is a long-running PHP-GTK mailing list available online at <http://marc.theaimsgroup.com/?l=php-gtk>. There's a lot of worthwhile information to be had here, but a great deal of it is repeated time and time again. However, you *can* search for particular terms, which increases its usefulness somewhat.

```
#include <gtk/gtk.h>
```

```
struct      GtkWidget;
GtkWidget*  gtk_window_new      (GtkWindowType type);
void        gtk_window_set_title (GtkWindow *window,
                                const gchar *title);

void        gtk_window_set_wmclass (GtkWindow *window,
                                const gchar *wmclass_name,
                                const gchar *wmclass_class);

void        gtk_window_set_focus (GtkWindow *window,
                                GtkWidget *focus);

void        gtk_window_set_default (GtkWindow *window,
                                GtkWidget *defaultw);
```

### The GTK+ C documentation. "It's documentation, Jim, but not as we know it."

What I believe will be the best resource for PHP-GTK in six months or so is the PHP-GTK Wiki. If you're new to the wiki phenomenon, then you've been missing out! "Wiki wiki" is Hawaiian for "quick", and it's basically a set of online documents that anyone – by default, absolutely everyone – can

add to, edit, and delete from. The PHP-GTK Wiki is an online FAQ where visitors can easily add and amend entries to make the best documentation available, and it is picking up pace as this is being written.

If you find yourself confounded to get the information you're looking for

about a particular widget or other GTK-related item, then you may find you have to bite the bullet and read the GTK C documentation available online at <http://developer.gnome.org/doc/API/gtk/>. This documentation is very thorough, however it's clarity leaves a little to be desired in places.

```
$button =& new GtkButton("Click Here");
$button->connect("clicked", "btnClick");
$button->set_relief(GTK_RELIEF_NONE);
$window->add($button);

$window->show_all();

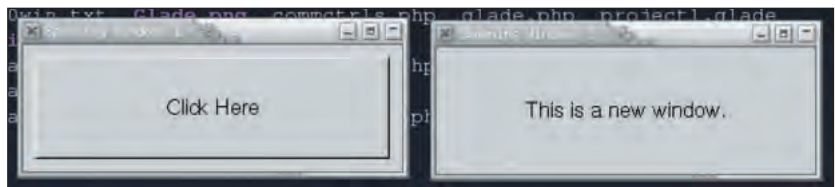
gtk::main();
?>
```

You should recognise about half of the code from the previous script. For now, again, ignore the two functions near the top and concentrate on the main body of code.

We create our `GtkWindow` in the same way as last time, however this time we follow up with three new methods: **set\_title()**, **set\_default\_size()**, and **set\_border\_width()**. These methods are all named quite clearly, but just to make sure we're on exactly the same wavelength, **set\_title()** sets the titlebar caption for this window, **set\_default\_size()** sets the initial width and height of this window, and **set\_border\_width()** sets the amount of margin space in pixels on each edge of the window that is unavailable to child widgets.

If **set\_border\_width()** had been used with a positive value in the prior example, the `GtkButton` being used would not have taken up *all* of the space in the window, just what was left after the border.

For all intents and purposes, the rest of the script is pretty much similar, with the most notable exception being the function `btnClick()`. This time the function is much longer, and also has a return value. If your signal handlers send back **false** as their return value, PHP-GTK fires the default signal handler as soon as your function finishes, whereas if you return **true**, it is assumed that you wish no more processing to take place for this signal beyond any other signal handlers you have defined.



At the start of `btnClick()`, a new `GtkWindow` is created with a new caption and a default size. Also, a new `GtkLabel` is created, which is a basic widget that allows you to display short amounts of text. As with `GtkButtons`, `GtkLabels` take the string they should display as a parameter when being created, and you can change this string at a later date by using the method **set\_text()**.

Continuing on in the function, the label is added to the window, our shutdown function is connected to the 'destroy' signal, and the new window is shown. Attaching `doshutdown()` to the destroy signal of each window being created means that if the user closes *any* window, the application will terminate – you may want a different situation in your own programs.

The other change that you'll notice in this script is the call to the **set\_relief()** method of our `GtkButton`. Like the **set\_title()**, **set\_default\_size()**, and **set\_border\_width()** calls in this script, this method isn't *necessary*, but I've included it to show you more of the GTK functionality. This method takes one of three special constants: `GTK_RELIEF_NORMAL` (the default setting), `GTK_RELIEF_HALF` (much lighter shading for buttons), and `GTK_RELIEF_NORMAL` (no shading for buttons unless mouse is over the button).

Now you understand what's going on in the script, go ahead and save it as **gtk2.php** and run it. I've included another screenshot at the top of this column of how this should look when your mouse is over the button, although my mouse cursor is invisible in the screenshot.

**A slightly more complicated PHP-GTK script, this time with multiple windows being created.**





## ◀◀ Handling popup menus

With over a hundred different widgets supported, GTK+ makes for a very rich programming environment. However, there is a generally accepted set of 'standard' widgets that are most commonly used in applications, one of which is the popup menu.

If you're a long-time shell person who really doesn't know much about GUIs, then you should understand that popup menus are also known as context-sensitive menus and generally appear close to the mouse pointer when the right mouse button is pressed. Anyway, here's the next script:

```
<?php

function doshutdown() {
    gtk::main_quit();
}

function show_popup($event, $menu) {
    if ($event->button == 3)
        $menu->popup(null, null, null, $event->button, $event->time);
}

function mnunew_click($new) {
    echo "New clicked!\n";
}

function mnuopen_click($new) {
    echo "Open clicked!\n";
}

function mnuexit_click($new) {
    echo "Exit clicked!\n\n";
    doshutdown();
}

$menu =& new GtkMenu();
$new =& new GtkMenuItem("New");
$open =& new GtkMenuItem("Open");
$sep =& new GtkMenuItem("");
$sep->set_sensitive(false);
$exit =& new GtkMenuItem("Exit");

$menu->append($new);
$menu->append($open);
$menu->append($sep);
$menu->append($exit);
$menu->show_all();

$window =& new GtkWindow();
$window->set_title("Using menus");
$window->set_default_size(300, 100);
$window->connect("destroy", "doshutdown");

$window->add_events(GDK_BUTTON_PRESS_MASK |
    GDK_BUTTON_RELEASE_MASK);
$window->connect_object('button-press-event', 'show_popup',
    $menu);

$new->connect("activate", 'mnunew_click');
$open->connect("activate", 'mnuopen_click');
$exit->connect("activate", 'mnuexit_click');
```

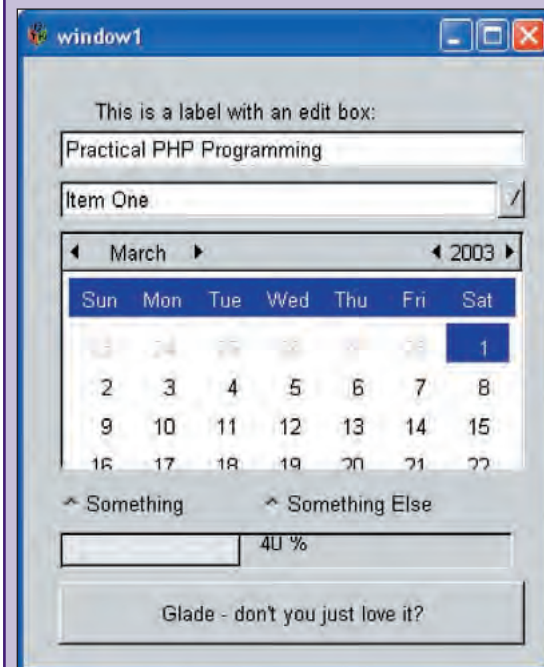
## Distributing your applications Platform-independent code?

PHP is cross-platform. GTK is cross-platform. So, surely it's easy to transfer your code from one platform to another, right? Well, sort of. In order to get PHP-GTK working, you need, unsurprisingly, PHP and GTK installed. Furthermore, if you wish to use *Glade*, you will also need *libglade* installed. Naturally, very few Windows users fit this criteria, so you have two options: either you provide very (very!) detailed installation instructions, or you package up the PHP interpreter, any DLLs you want to use, a pre-written php.ini file, and include it all in your installation. This is actually quite a good option, as you can

be sure it will work with no hassle for your users, and it only adds around 2MB to your package once compressed.

For Unix users, generally speaking we're used to software having a few installation requirements, so generally noting on your website and in the docs "Requires PHP4.x, GTK 1.x" etc should be fine.

However, as you can see in the screenshot below, our Glade script runs perfectly well on Windows. Skins are unavailable in the existing Windows port of GTK, however, so sadly Windows users have to live with a less-than-beautiful interface.



**Cross-platform coding in action. Yes, it's Microsoft Windows. Yes, you're still reading *Linux Format*...**

```
$window->show_all();

gtk::main();

?>
```

If you're wondering at the length of that script, don't worry – it's the longest in this article! You already know quite a bit of what goes on in there also, so it's probably not all that fearsome.

Starting at the line **\$menu =& new GtkMenu()**, a new class is introduced: *GtkMenu*. This widget is solely designed to host a popup menu, but it intertwines cleverly with *GtkMenuBar*, the widget designed to host the horizontal-style menu bar, and we'll discuss that later.

Each `GtkMenu` contains several `GtkMenuItem`s, and these are created similarly to `GtkButtons` and `GtkLabels` in that you passed the string you wish them to display.

Just before creating the last `GtkMenuItem`, I slipped a blank `GtkMenuItem` in there that some of you may not have noticed. This item, upon which I call the `set_sensitive()` method that is universal to all GTK widgets, is there to act as a separator between New/Open and Exit. Creating a `GtkMenuItem` with no text results in a blank menu item that may still be selected by users. Calling `set_sensitive()` on that widget and passing in `false` disables the widget.

Through the use of the `append()` method of `GtkMenu` we add our `GtkMenuItem`s to our popup menu then create and set-up the `GtkWindow` itself. Two new methods are called here: `add_events()` and `connect_object()`.

`add_events()` is a peculiar but very helpful function that allows you to modify which events a given object captures. In essence, you can make a widget listen to an event it ignores by default. The method takes one parameter (we OR two parameters into one in the example), which is a bit mask of constants from the `GdkEventMask` list. In the long piece of script on the preceding page, `GDK_BUTTON_PRESS_MASK` and `GDK_BUTTON_RELEASE_MASK` are combined into one bitmask before being passed in, which makes the widget calling `add_events`, our `GtkWindow`, respond to mouse buttons being pressed and released. In turn, our `GtkWindow` will emit the signal `button-press-event`, which we bind a function to in the next line.

Similar to the `connect()` method we've been using so far, the `connect_object()` method also connects signals to functions, with the key difference that the object passed into the handler function isn't the object you used to call the method on. Instead, the object passed in is the one you set as parameter three to `connect_object()`.

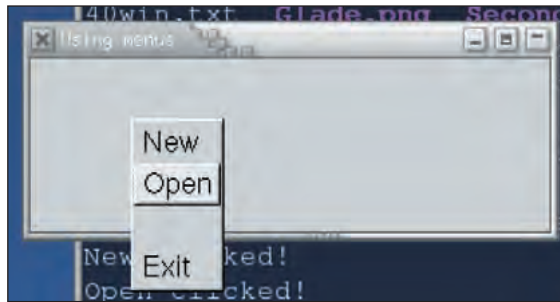
You might not think this is particularly helpful, but in the example we set `$menu` as the object to be passed to our `show_popup()` function. If `connect()` had been used as opposed to `connect_object()` we would have had to try to get a handle to `$menu`, because a pretty useless reference to the `GtkWindow` would have been passed in. Whilst this can be overcome with custom parameters (see the box about *Using Custom Parameters* overpage for more information), it's more logical to use `connect_object()`.

So, at the end of the day, we tie our `GtkWindow`'s `button-press-event` signal to our `show_popup` function. Note that the first parameter in `show_popup()` is of the type `GdkEvent`, which at the time of writing sadly seems entirely undocumented in the PHP-GTK documentation (please correct me if I'm wrong). However, it is documented in the GDK developer documentation, albeit in C++, which can be found at:

<http://developer.gnome.org/doc/API/gdk/gdk-event-structures.html#GDKEVENT>.

The particular `GdkEvent` type we're interested in in this situation is `GdkEventButton`, documented at <http://developer.gnome.org/doc/API/gdk/gdk-event-structures.html#GDKEVENTBUTTON>. This event is sent when buttons are clicked and released, which is what we're looking to handle. If your C is sketchy, never fear – here's a quick breakdown of some of the data included in this event:

**button** – the mouse button press (left=1, middle=2, right=3)  
**time** – the time, in milliseconds, that the event occurred



Popup menus are easy to create, and very powerful.

**x, y** – the x and y coordinate of the mouse

**state** – a bitmask of `GdkModifierTypes` (see the main PHP-GTK documentation) that describes whether Control was held down, etc  
**pressure** – generally only used for graphics tablets, this is a floating point value from 0 to 1 describing how "hard" the button was clicked. This defaults to 0.5 for mouse clicks

So, the first line of the function checks which button was pressed to generate the event, and, if it was button 3 (the right mouse button), we call the `popup()` method of our `GtkMenu`.

`popup()` takes a total of five parameters: the first two are generally null as they are only used when tying menus to `GtkMenuItem`s. Parameter three is null in the example, but can be the name of a function to call to return the x and y coordinates at which you wish your menu to appear as an array. So, for example:

```
function mnupos() {
    return array(50, 200);
}

$menu->popup(null, null, 'mnupos', $event->button, $event->time);
```

When the third parameter is null, the current mouse coordinates are used, which is usually the desired result. Parameter four is the button that was pressed to generate the event, and finally parameter five is the time the event took place, in milliseconds. As seen above, even though there are five parameters for the method, you'll see that it's actually quite straightforward to use.

The `mnuexit_click()` function could have been removed entirely because, as well the fact that multiple functions can be connected to a single signal, multiple *signals* can be connected to a single *function*. If there were no special processing to be run when Exit was clicked (in our example we echo to the console), then the activate `GtkMenuItem` signal could have been connected to `doshutdown()` as well as the `GtkWindow` destroy signal.

Save this script as `gtk3.php` and run it as before. Again, check my screenshot at the top of this column to make sure you got everything right.

As promised earlier in this tutorial, I want to briefly mention how `GtkMenu` bar works. A menu created with `GtkMenu` can be used as seen above, where it is activated in a floating space. It can also be used to provide the contents of a horizontal-style menu bar item, for example 'New, Open, Save, Exit' would be the `GtkMenu` that was attached to the 'File' `GtkMenuItem` of a `GtkMenuBar` widget.

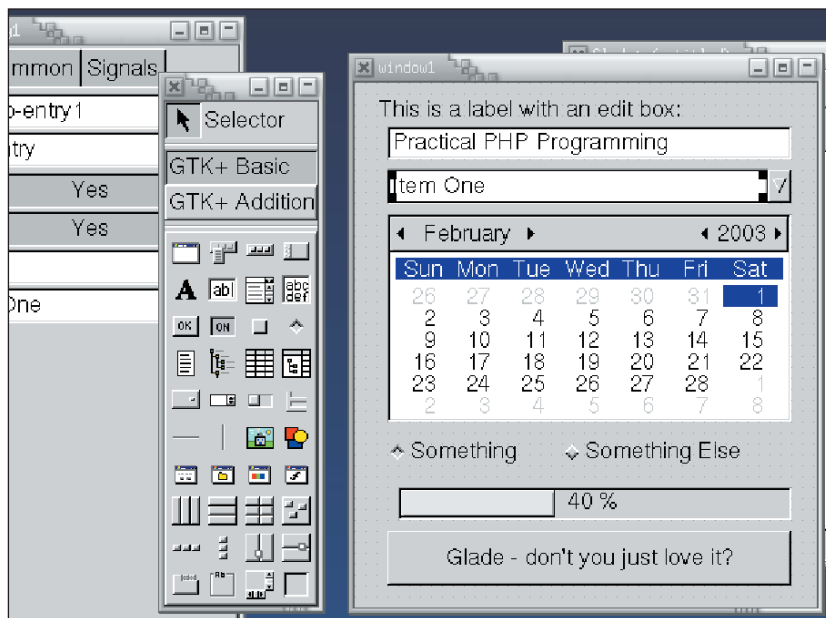
Does this make sense? If not, here's a quick piece of code to demonstrate what I mean:

```
$mainmenu =& new GtkMenuBar();
$filemenu =& new GtkMenuItem("File");
$mainmenu->append($filemenu);
```





# TutorialPHP



With *Glade*, interface design is a snap.



```
$filemenuoptions = &new GtkMenu();
$open =& new GtkMenuItem("Open");
$filemenuoptions->append($open);
$save =& new GtkMenuItem("Save");
$filemenuoptions->append($save);
$filemenu->set_submenu($filemenuoptions);
```

So, a `GtkMenuBar` is the menu strip along the top of your window. The top level items (eg: File, Edit, Document, Bookmarks, etc, in KDE's Kate 2.1) are `GtkMenuItems`, which each contain a `GtkMenu` of their contents. The 'Document' `GtkMenuItem` in *Kate* would contain a `GtkMenu` which itself contained `GtkMenuItems` for Back, Forward, and any open files.

## Advanced GUIs

There are so many possibilities using PHP-GTK that, sadly, I've had to pick and choose what I can cover here owing to space reasons – and that's despite the fact that this tutorial is extra long! So far we've looked at windows, buttons, labels, menus, and menuitems.

What we're going to look at now is an easy way to use all sorts of GTK widgets, perfectly lined up where you want them to be, with many widgets in the same window, and, surprisingly enough, with almost no work. This is the power of *Glade*.

Available from <http://glade.gnome.org>, *Glade* is a GPLed GTK+ user interface builder designed to allow you to design and build your GUI, including defining signal handler functions, with little work.

Take a look at the screenshot at the top of this column to see *Glade* in action. As you can see in the picture, you have a big toolkit available to you under the 'GTK+ Basic', and another large toolkit available under 'GTK+ Additional'. When you want to make use of a particular widget, you simply have to select from the toolbox and 'draw' on your window. Properties can be set from a property editor which is partly offscreen to the left. Once you're finished designing the parts of your GUI, you can even instruct *Glade* to generate source code for you, although sadly, this is not yet available in PHP.

However, there is still a way *Glade* can be used with PHP. Take a look at this final script:

```
<?php

function doshutdown() {
    gtk::main_quit();
}

$layout = &new GladeXML('complex_interface.glade');
$layout->signal_autoconnect();

$window = $layout->get_widget('window1');
$window->connect("destroy", "doshutdown");

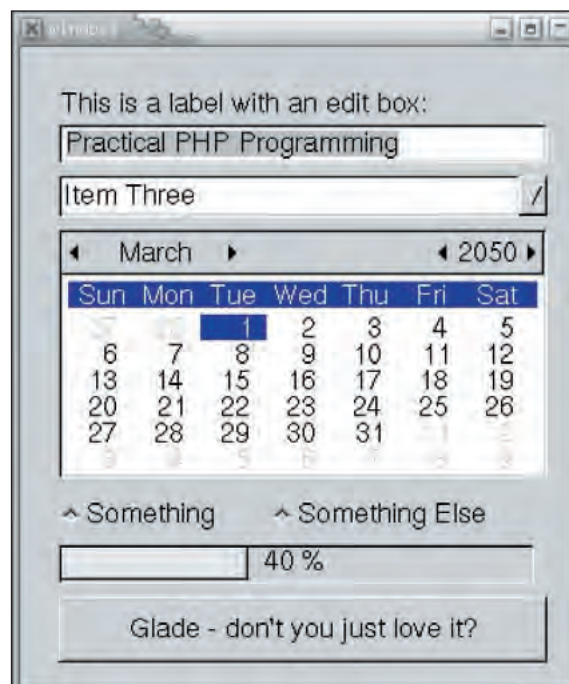
Gtk::main();
?>
```

Here, there is a new class available if you have *libglade* installed. That takes the *.glade* project file that *Glade* saves for its own purposes and translates that into a GUI. This GUI, stored in **\$layout** in the example above, can then have its signals connected using the `GladeXML` method **signal\_autoconnect()**.

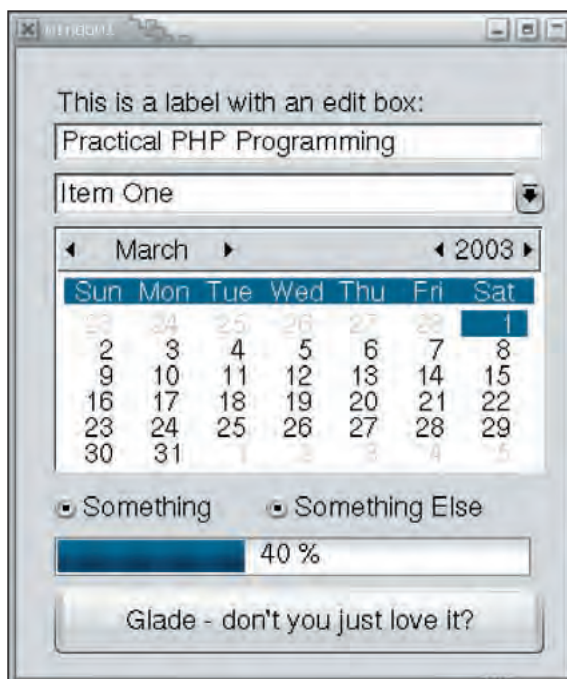
In order to provide a clean shutdown of the script, I have used the `GladeXML` method **get\_widget()** to grab the main window. Note that you may need to change this line if you have used a particular name for your window in *Glade*. **get\_widget()** takes just the one parameter, which is the name of the widget you wish to get from the layout, and returns the widget for you to use.

With our `GtkWindow` reference, I have connected the destroy signal to our usual **doshutdown()** function, and that's the end of the script.

If you save that as **gtk4.php**, you can then go experiment with *Glade* to see what you can make. You can see my interface live in action in the screenshot below, though you'll see that some bits are still labelled 'something'. As you can see, using *Glade* takes all the hard work away from designing a GUI. All that's left to do now is to write handlers for all the signals you wish to work with, and your interface is done.



Here is the script running, and it looks just like it did when it was being built in *Glade*. Magic.

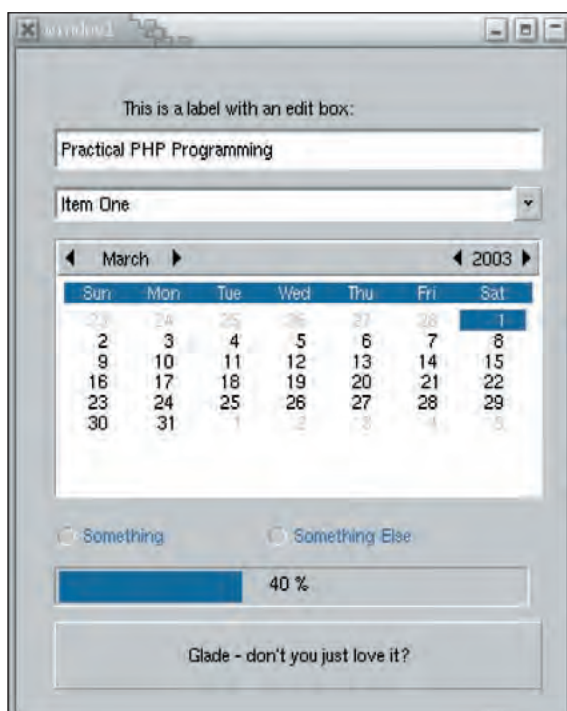


Our Glade-built GUI doing a KDE impression.

## Themes

Some people say that unthemed GNOME looks a little unsightly, and they might be right. However, luckily all applications made using PHP-GTK happily work with GTK+ 1.x themes with no additional work. I myself have GNOME 2.2 installed, and so I needed to specifically install a GTK1 theme as GTK2 themes do not work with PHP-GTK or other GTK1 apps.

Anyway, themes are all transparent to your scripts – you can't tell whether a particular theme is enabled or not, and neither should you need to as your scripts will adapt to whatever the user has selected.



And again, this time doing a Microsoft Windows impression.

## Using Custom Parameters

### Get connected

When you're connecting signals to functions, it is possible to add one or more custom parameters to the signal handler. As seen earlier, **connect\_object()** can be used to pass a particular widget into a function, however an alternative is to use **connect()** with extra parameters for the information you wish to use inside the function.

So, **connect()** could have been called like this:

```
$window->connect('button-press-event',
    'show_popup', $menu);
```

The handler function would then need to have accepted three parameters – the event, the **GtkWindow** object that emitted the signal, and the custom parameter **\$menu**. The only difference here is that behind the scenes a little more data needs to be passed around for the handler function to be called with the extra parameter.

Here's a complete code example you can try out to get the idea:

```
<?php
function btnClick($button, $window) {
    $window->destroy();
    gtk::main_quit();
}

$window =& new GtkWindow();
$btnquit =& new GtkButton("Quit");
$btnquit->connect("clicked", "btnClick",
    $window);
$window->add($btnquit);
$window->show_all();
gtk::main();
?>
```

As you can see, the parameter **\$button** isn't being used inside **btnClick()**, however that's hardly a major speed hit.

Take a look at screenshots on this page to see the same Glade GUI interface themed in different ways.

## Conclusion

If you've made it this far, the chances are the sun will be rising outside shortly and you had best get some sleep before morning! However, hopefully you will have learned a great deal about the coolest – and probably least-exploited – alternative use for PHP.

Creating graphical applications for PHP may, at first, not seem "right", which is quite true to some extent. After all, PHP was designed to be a language for general web use, and not for GUIs. However, once you get over the initial, and indeed inevitable shock of switching to signal-based programming, it is normally a pleasant experience.

The GTK version used with PHP-GTK is quite old, and recent releases have been far superior, with a lot of work done to GTK to make it more flexible, with more intuitive interaction between objects. While GTK2 support is not currently on the cards for PHP-GTK, I shan't imagine it will be long. Until then, there's more than enough information to help you get to grips with using the current release of PHP-GTK – good luck! **LXF**

## Make your mark

### Brainstorms 'R' Us

Would you like to get your name in the mag and learn about stuff you're most interested in?

We're looking out for ideas for new *Linux Format* Practical PHP tutorials, and where better to look than to you, the reader? If, while reading past issues of Practical PHP, you've thought "I wish they'd covered XYZ in more depth...", or "I really want to know how to use...", then now's the time to get your voice heard!

Send an email to [paul.hudson@futurenet.co.uk](mailto:paul.hudson@futurenet.co.uk) with your ideas – all the good ideas that you send in will be covered in future issues. So far, the topics we have covered in some depth include MySQL, XML, CLI, GUIs, media generation, templates, and more.

If you're short of ideas, you're certainly welcome to write in with comments about prior issues – we're always looking to improve the overall quality of tutorials.

## NEXT MONTH

Next month we'll be looking at potentially the most complicated aspect of PHP, and that is creating your own modules for the language. If you thought this month's topic was tricky, you've got another think coming – next month, knowledge of C is strongly recommended.

If you have any comments or suggestions about this series, please be sure to write in.



## LOVE LETTERS

# Containing your feelings for Python



**Patrick K. O'Brien** tries to contain his feelings for Python and ends up counting love letters.

**A**s you begin to write programs in Python, you'll soon find yourself dealing with a multitude of objects that must be kept track of and organised. In fact, your programs will often deal with so many objects that assigning individual variable names to each would be impractical and burdensome. The solution to this problem is **containers**: objects that are designed to hold a collection of other objects, without having to name each of them. In this issue we'll examine four Python object types that act as containers: strings, lists, tuples, and dictionaries.

Let's begin by looking at the basic differences between these four types. The first three (strings, lists, and tuples) all contain sequences of objects. Strings are limited to sequences of characters, while lists and tuples have no limitation and may contain objects of any type. The primary difference between lists and tuples is that lists are mutable and tuples are immutable. The dictionary is unique in that it does not contain a sequence. Instead a Python dictionary contains mappings, or associations, between keys and values, much like a paper dictionary contains mappings

between words and their definitions. Unlike the paper dictionary, however, a Python dictionary has no inherent order.

It's probably difficult to fully comprehend the descriptions in that previous paragraph without seeing these container-like objects in action. So let's take a look at each of them in detail, with plenty of examples. As I've done in previous issues, I'm going to recommend that you follow along using a Python shell. A graphical Python shell, such as my *PyCrust* program, can greatly increase your understanding of Python objects, especially the container-like objects we'll be discussing in this issue.

I've recently made some changes to the *PyCrust* program, including a new, tabbed interface. You can see the new interface in Figure 1, which shows a *PyCrust* session with a dictionary object selected in the namespace tree. By entering the code examples yourself, and examining the results using *PyCrust*, you'll quickly increase your proficiency with Python.

## Strings

We talked about strings in the last issue, but there we focused on strings as simple representations of textual information. Now we'll focus on strings as containers that hold sequences of characters.

As a container of characters, individual characters in a string can be accessed by their index value, a value that identifies a position in the sequence. To access a particular index position you simply specify the index value in square brackets following the string. Index values start at zero and increase by one:

```
>>> s = 'Love'
>>> s[0]
'L'
>>> s[1]
'o'
>>> s[2]
'v'
>>> s[3]
'e'
```

Indexing works the same on string literals as it does on string variables:

```
>>> 'Love'[0]
'L'
```

A character can also be accessed in relation to its position from the end of the sequence. Index values from the end start with **-1** and decrease by one:

```
>>> s[-1]
'e'
>>> s[-2]
'v'
>>> s[-3]
'o'
>>> s[-4]
'L'
```

Going too far will raise an `IndexError`:

```
>>> s[-5]
Traceback (most recent call last):
  File "<input>", line 1, in ?
IndexError: string index out of range
```

Indexing works well for individual characters, but sometimes you want to access an entire substring. To do that you can use slicing. Slicing looks a lot like indexing, except that you supply two index values separated by a colon. A substring is then extracted, starting at the first index value and including all the characters up to, but not including, the second index value:

```
>>> s[0:2]
'Lo'
>>> s[2:4]
've'
Negative index values also work with slicing:
>>> s[1:-1]
'ov'
>>> s[-4:-2]
'Lo'
```

Python allows for some useful slicing shortcuts. If you leave out the first index value, Python defaults to the beginning of the sequence; if you leave out the second index value, it defaults to the end:

```
>>> s[:1]
'Lo'
>>> s[1:]
'ove'
```

## Python strings

A Python string is an immutable, ordered sequence of characters.

Create string literals using matching pairs of single, double, or triple quotes:

```
>>> emptystring = ""
```

```
>>> s = "what's a string?"
```

```
>>> s = 'this is a string'
```

Extract individual characters using the index operator:

```
>>> s[0]
```

```
't'
```

```
>>> s[1]
```

```
'h'
```

```
>>> s[2]
```

```
'i'
```

```
>>> s[-1]
```

```
'g'
```

```
>>> s[-2]
```

```
'n'
```

```
>>> s[-3]
```

```
'i'
```

Extract substrings using the slicing operator:

```
>>> s[0:4]
```

```
'this'
```

```
>>> s[:4]
```

```
'this'
```

```
>>> s[-4:]
```

```
'this is a st'
```

```
>>> s[-4:]
```

```
'ring'
```

```
>>> s[2:7]
```

```
'is is'
```

```
>>> s[:]
'Love'
Python is also quite forgiving if your
slicing specifications are illogical or exceed
the size of the string:
>>> s[:12]
'Love'
>>> s[9:]
''
>>> s[3:1]
''
>>> s[-2:-4]
''
```

Because a string is immutable, you can't use indexing or slicing to change a string in place, the way you can with a list, for example. If you try, you'll get an error:

```
>>> s[0] = 'M'
Traceback (most recent call last):
  File "<input>", line 1, in ?
TypeError: object doesn't support item
assignment
But we can still achieve that kind of change
by binding the same variable name to the
result of a string operation, like this:
>>> s = 'Love'
>>> s = 'M' + s[1:]
>>> s
'Move'
>>> s = s.lower()
>>> s
'move'
```

Don't forget that a string is an object with many methods, such as the **lower()** method in the above example. One of the more useful of these is the **split()** method, which will split a string according to a specified separator. If no separator is

provided, the string is split along any whitespace. Here is an example:

```
>>> s = 'I will write to you every day.'
>>> s.split()
['I', 'will', 'write', 'to', 'you', 'every', 'day.']
The value returned by the split() method
is a list. In this case, a list of strings.
```

## Lists

Like strings, lists contain sequences of objects, but unlike strings, lists are not limited to sequences of individual characters. In fact, a list can contain any kind of Python object (even another list) in any combination. And while we couldn't make changes to a string in place, we can with a list. Lists are mutable: you can insert, replace, and delete elements of a list:

```
>>> s = 'I will write to you every day.'
>>> l = s.split()
>>> l[2] = 'sing'
>>> del l[-1]
>>> l.append(2.5)
>>> l.append('hours:')
>>> l
['I', 'will', 'sing', 'to', 'you', 'every', 2.5, 'hours:']
>>>
```

When you make changes to a list, you aren't limited to single item changes. You can insert, replace, and delete entire sections of a list, using the slicing operator:

```
>>> l[1:-2] = ['must', 'have']
>>> l
['I', 'must', 'have', 2.5, 'hours:']
>>> del l[-2:]
>>> l
['I', 'must', 'have']
>>> l[3:] = ['chocolate:']
```

```
>>> l
['I', 'must', 'have', 'chocolate:']
>>> l[1:2] = ['would', 'like', 'to']
>>> l
['I', 'would', 'like', 'to', 'have', 'chocolate:']
A list can contain other lists. In fact, a
list can even contain itself recursively,
which Python represents using an ellipsis:
>>> l = [1, 2, 3]
>>> l.append(l)
>>> l
[1, 2, 3, [...]]
>>> l[-1]
[1, 2, 3, [...]]
>>> l[-1][-1]
[1, 2, 3, [...]]
Because a list is mutable, changes to a list
will be visible anywhere that list is referenced:
>>> alist = ['a', 'b', 3, 4]
>>> blist = [7, 8, alist]
>>> blist
[7, 8, ['a', 'b', 3, 4]]
>>> alist.append('a new string')
>>> alist
['a', 'b', 3, 4, 'a new string']
>>> blist
[7, 8, ['a', 'b', 3, 4, 'a new string']]
```

There are many ways of creating a list, beyond typing in a list literal. We saw **split()** earlier, which returns a list of strings. Another useful way is the built-in **range()** function, which returns a list of integers:

```
>>> range(5)
[0, 1, 2, 3, 4]
```

A relatively new and fairly sophisticated way to create a list is with a list comprehension. Because a list comprehension is closely related to a

## Python dictionaries

A Python dictionary is a mutable, unordered mapping of key/value pairs.

Create a dictionary by enclosing any number of comma-separated key:value pairs inside curly brackets:

```
>>> emptydict = {}
>>> d = {'a': 'alpha', 'z': 'zeta', 'i': 'iota'}
>>> d
{'a': 'alpha', 'i': 'iota', 'z': 'zeta'}
Access, insert, or
modify dictionary values
by key:
>>> d['a']
'alpha'
>>> d['z']
'zeta'
>>> d['z'] = 'zippy'
>>> d['k'] = 'kappa'
>>> del d['i']
>>> d
{'a': 'alpha', 'k': 'kappa', 'z': 'zippy'}
```

Return lists of keys, values, and item tuples using dictionary methods:

```
>>> d.keys()
['a', 'k', 'z']
>>> d.values()
['alpha', 'kappa', 'zippy']
>>> d.items()
[('a', 'alpha'), ('k', 'kappa'), ('z', 'zippy')]
```

## Mutable vs Immutable

Which object type is which?

Some objects can be changed; others cannot. Those that can change are called 'mutable,' and those that cannot are called 'immutable.' You'll find many aspects of Python easier to understand once this distinction becomes ingrained in your mind. Let's look at a couple of examples to appreciate the ramifications of mutability.

Strings are immutable. In order to make a change involving a string, we need to assign the results of a string operation to a new variable (or bind the result to the same variable):

```
>>> s = 'some string'
>>> s.upper()
'SOME STRING'
>>> s # not changed
```

```
'some string'
>>> s = s.upper()
>>> s
'SOME STRING'
Lists are mutable. We can
change a list in place:
>>> l = [2, 3, 8, 5, 7]
>>> l.append(6)
>>> l
[2, 3, 8, 5, 7, 6]
>>> l.sort()
>>> l
[2, 3, 5, 6, 7, 8]
>>> l.append(4)
>>> l
[2, 3, 5, 6, 7, 8, 4]
```

Of the object types we've discussed so far in this tutorial, strings, numbers, and tuples are immutable; lists and dictionaries are mutable.

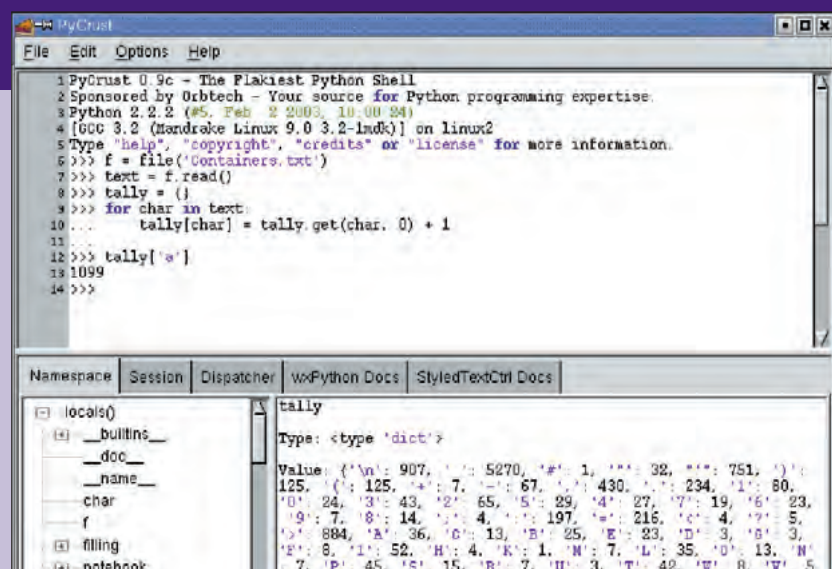


Figure 1: The namespace tree in PyCrust allows you to navigate Python objects. In this session I've written some code to count the occurrence of letters in this article. I store the results in a dictionary named **tally**, which is selected in the namespace tree.



for loop, we'll postpone discussing it in detail until the next issue. In the mean time, here is an example of a list comprehension being used to select and uppercase all the string items from another list:

```
>>> l = [8, 'one', 4.5, 'true', -3, 'love']
>>> [item.upper() for item in l if
instance(item, str)]
['ONE', 'TRUE', 'LOVE']
```

Because lists are mutable, they come with a number of methods to support the most common operations you would want to perform on a list. Let's look at a few of these. You can add onto the end of a list using the **append()** method:

```
>>> l = ['c', 'd', 'a']
>>> l.append('c')
>>> l
['c', 'd', 'a', 'c']
```

Use the **index()** method to get the index position of the first occurrence of an item:

```
>>> l.index('a')
2
>>> l[2]
'a'
```

And use the **remove()** method to remove the first occurrence of an item:

```
>>> l.remove('c')
>>> l
['d', 'a', 'c']
```

The **count()** method will tell you how many times an item appears in a list:

```
>>> l = [3, 2, 8, 2, 7, 2, 6]
>>> l.count(2)
3
```

We've seen several examples where we've appended to a list. But sometimes you need more control over where items gets placed into a list. For that, you use the **insert()** method, and you supply it with two pieces of information: the index value in front of which the item will be inserted, and the item itself:

```
>>> l = ['c', 'e']
>>> l.insert(0, 'b')
>>> l
['b', 'c', 'e']
>>> l.insert(2, 'd')
>>> l
['b', 'c', 'd', 'e']
```

The **pop()** method will return a list item and remove it from the list. You can pop the item at a specific index position, or the last item if you don't specify an index:

```
>>> l = ['a', 'b', 'c', 'd', 'e']
>>> l.pop()
'e'
>>> l
['a', 'b', 'c', 'd']
>>> l.pop(2)
```

```
'c'
>>> l
['a', 'b', 'd']
>>> l.pop()
'd'
>>> l.pop()
'b'
>>> l
['a']
```

A list is often more useful when it's sorted:

```
>>> l = ['c', 'd', 'a', 'b', 'e']
>>> l.sort()
>>> l
['a', 'b', 'c', 'd', 'e']
```

The order of a list can be reversed:

```
>>> l = ['c', 'd', 'a', 'b', 'e']
>>> l.reverse()
>>> l
['e', 'b', 'a', 'd', 'c']
```

If you want your list to be in reversed, sorted order, first sort it, then reverse it:

```
>>> l = ['c', 'd', 'a', 'b', 'e']
>>> l.sort()
>>> l.reverse()
>>> l
['e', 'd', 'c', 'b', 'a']
```

## Tuples

Tuples are almost identical to lists, with one significant exception – tuples are immutable: once defined, the shape of a tuple cannot change – you cannot add on to a tuple, or delete one of its items; and you cannot replace an object that is referenced in a tuple. Of course, if the referenced item happens to be mutable, you can still make changes to it. These rules will be easier to understand once we look at some examples.

Let's start by creating a few tuples. While it is common to enclose tuples inside parentheses, they aren't strictly necessary. The appearance of commas in the sequence tells Python we're creating a tuple:

```
>>> t = (1, 2, 3)
>>> t
(1, 2, 3)
>>> t = 1, 2, 3
>>> t
(1, 2, 3)
```

Tuples are immutable, so we can't change them the way we could lists. But a tuple can contain a list, which is mutable. And we can still make changes to the list, even though it is referenced in a tuple:

```
>>> l = ['a', 'b', 'c']
>>> t = (3, l, 4)
>>> t
(3, ['a', 'b', 'c'], 4)
>>> l.append('d')
>>> t
(3, ['a', 'b', 'c', 'd'], 4)
```

```
>>> t[1].append('e')
>>> t
(3, ['a', 'b', 'c', 'd', 'e'], 4)
>>> l
['a', 'b', 'c', 'd', 'e']
```

Another ramification of tuples' immutability is that tuples lack many of the methods available to list objects, such as **append**, **insert**, **remove**, **pop**, **reverse**, and **sort**.

Since lists can do everything a tuple can do, and then some, you might wonder why Python has tuples at all. The main reasons are size and speed; tuples require less memory and are faster than lists. So if you don't need mutability, use a tuple.

## Any sequence will do

Strings, lists, and tuples are all sequence types. That means certain operations work on all of them. We covered the operations that are unique to each of them earlier. Now we're going to look at some operations that work essentially the same for any sequence type.

In the last issue we saw that we could concatenate two strings together using the concatenation operator (+). We can also concatenate lists and tuples:

```
>>> [1, 2, 3] + ['a', 'b', 'c']
[1, 2, 3, 'a', 'b', 'c']
>>> (4, 5, 6) + ('d', 'e', 'f')
(4, 5, 6, 'd', 'e', 'f')
```

You can count the number of items in a sequence with the **len()** function:

```
>>> s = 'python'
>>> l = [1, 2, 3]
>>> t = ('L', 'O', 'V', 'E')
>>> len(s)
6
>>> len(l)
3
>>> len(t)
4
```

The **min()** function will return the smallest item in a sequence:

```
>>> min(s)
'h'
>>> min(l)
1
>>> min(t)
'E'
```

The **max()** function will return the largest item in a sequence:

```
>>> max(s)
'y'
>>> max(l)
3
>>> max(t)
'V'
```

You can also determine if an item is a member of a sequence:

## Python lists

A Python list is a mutable, ordered sequence of arbitrary objects.

Create a list by enclosing a sequence of comma-separated objects inside brackets:

```
>>> emptylist = []
>>> a = 16
>>> b = 6.5
>>> c = "some string"
>>> l = [a, 2, 'three', b, "Bob", c]
>>> l
[16, 2, 'three', 6.5, 'Bob', 'some string']
```

Access or modify individual list members using the index operator:

```
>>> l[0]
16
>>> l[-1]
'some string'
>>> del l[-1]
>>> l[2]
'three'
>>> l[2] = l[2].upper()
>>> l
[16, 2, 'THREE', 6.5, 'Bob']
Extract a portion of a list using the slicing operator:
>>> l[2:]
['THREE', 6.5, 'Bob']
>>> l[1:-1]
[2, 'THREE', 6.5]
```

```
>>> 'P' in s
0
>>> 'p' in s
1
>>> 4 in l
0
>>> 2 in l
1
>>> 'v' in t
0
>>> 'V' in t
1
>>> 'P' not in s
1
>>> 2 not in l
0
>>> 'v' not in t
1
```

When assigning variables to values from a sequence you can take advantage of a Python feature known as sequence unpacking. Sequence unpacking allows you to assign more than one variable at the same time, as long as you have matching numbers of variables on each side of the assignment operator:

```
>>> a, b, c = l
>>> a
1
>>> b
2
>>> c
3
>>> d, e, f, g = t
>>> d
'l'
>>> e
'o'
>>> f
'v'
>>> g
'e'
```

One common use of sequence unpacking is to swap variable values:

```
>>> a = 25
>>> b = 73
>>> a, b = b, a
>>> a
73
>>> b
25
```

## Dictionaries

A dictionary is not a sequence type. Instead, it is a collection of key, value pairs. Dictionaries provide a mapping between a key and a value: if you know the key you can locate the value associated with that key. A dictionary key must be an immutable object, but the dictionary value can be any type of Python object.

Instead of indexing by position, a dictionary is indexed by key:

```
>>> s = 'python'
>>> d = {'a': s, 'b': 3.5, 's': 7}
>>> d
{'a': 'python', 'python': 7, 'b': 3.5}
>>> d['a']
'python'
>>> d[s]
7
>>> d[s] = d['b'] + 18
>>> d[s]
21.5
>>> del d[s]
>>> d
{'a': 'python', 'b': 3.5}
```

As you can see in the first four lines above, Python does not necessarily maintain the order in which a dictionary was defined. Also note that dictionary keys must be unique. To associate more than one value with the same key, use a list or tuple for the values you wish to associate:

```
>>> d = {}
>>> d['alpha'] = ('a', 'A')
>>> d['beta'] = ('b', 'B')
>>> d
{'alpha': ('a', 'A'), 'beta': ('b', 'B')}
```

The **len()** function will tell you how many keys are in the dictionary:

```
>>> len(d)
2
>>> min(d)
'alpha'
>>> max(d)
'beta'
```

You can return a list of keys using **keys()**:

```
>>> d.keys()
['alpha', 'beta']
```

Similar methods return lists of values and items:

```
>>> d.values()
[('a', 'A'), ('b', 'B')]
>>> d.items()
[('alpha', ('a', 'A')), ('beta', ('b', 'B'))]
>>> 'alpha' in d
1
>>> 'zeta' in d
0
>>> 'zeta' not in d
1
```

By now you should be comfortable experimenting, and users of *PyCrust* will have autocompletion, calltips, and the namespace viewer to help them. Take a close look at dictionary objects, because they come with a host of methods we don't have room to discuss.

I will point out one method that will be used in our "love letters" program. The **get()** method returns the value associated with a key, if it exists, otherwise it returns a default value that you supply:


```
>>> d.get('alpha', ('', ''))
('a', 'A')
>>> d.get('zeta', ('', ''))
('', '')
```

The **get()** method is handy for creating an initial value, such as when you keep counts for items stored in a dictionary.

## Counting love letters

I think it's time to see how we can apply the concepts from this tutorial to a practical application. I figure that since I'm writing about my love for Python, the letters that make up this article must therefore be "love letters." And since there are so many, you might want to gauge my affection by counting these letters, perhaps to find out which are the most frequent letters of love. Of course, we'll let Python do the actual counting, using all of the container-like objects we just discussed:

```
>>> f = file('Containers.txt')
>>> text = f.read()
>>> tally = {}
>>> for char in text:
...     tally[char] = tally.get(char, 0) + 1
...
>>> l = [(num, char) for (char, num) in tally.items()]
>>> l.sort()
>>> l.reverse()
>>> for n in range(10):
...     t = l[n]
...     num = t[0]
...     char = t[1]
...     print n+1, 'Love letter', repr(char),
'appears', num, 'times.'
...
1 Love letter ' ' appears 5270 times.
2 Love letter 'e' appears 1711 times.
3 Love letter 't' appears 1361 times.
4 Love letter 'a' appears 1099 times.
5 Love letter 's' appears 1042 times.
6 Love letter 'i' appears 1011 times.
7 Love letter 'n' appears 978 times.
8 Love letter 'o' appears 941 times.
9 Love letter '\n' appears 907 times.
10 Love letter '>' appears 883 times.
```

It appears from this analysis that a space is the most frequent letter of love, proving the old adage that "absence makes the heart grow fonder." If the frequent absence of letters in this article has made your heart grow fonder, then I've succeeded as chaperone and caretaker of the budding romance between you and Python. 

## Python tuples

A Python tuple is an immutable, ordered sequence of arbitrary objects.

Create a tuple by enclosing a sequence of comma-separated objects inside parentheses:

```
>>> emptytuple = ()
>>> a = 1
>>> b = 6.5
>>> c = "some string"
>>> t = (a, 2, 'three', b,
'Bob', c)
>>> t
(1, 2, 'three', 6.5, 'Bob', 'some string')
```

Access individual tuple members using the index operator:

```
>>> t[0]
1
>>> t[-1]
'some string'
```

Attempts to change a tuple will raise an exception:

```
>>> t[2] = t[2].upper()
Traceback (most recent call last):
  File "<input>", line 1, in ?
TypeError: object doesn't support item assignment
>>> del t[-1]
Traceback (most recent call last):
  File "<input>", line 1, in ?
TypeError: object doesn't support item deletion
```

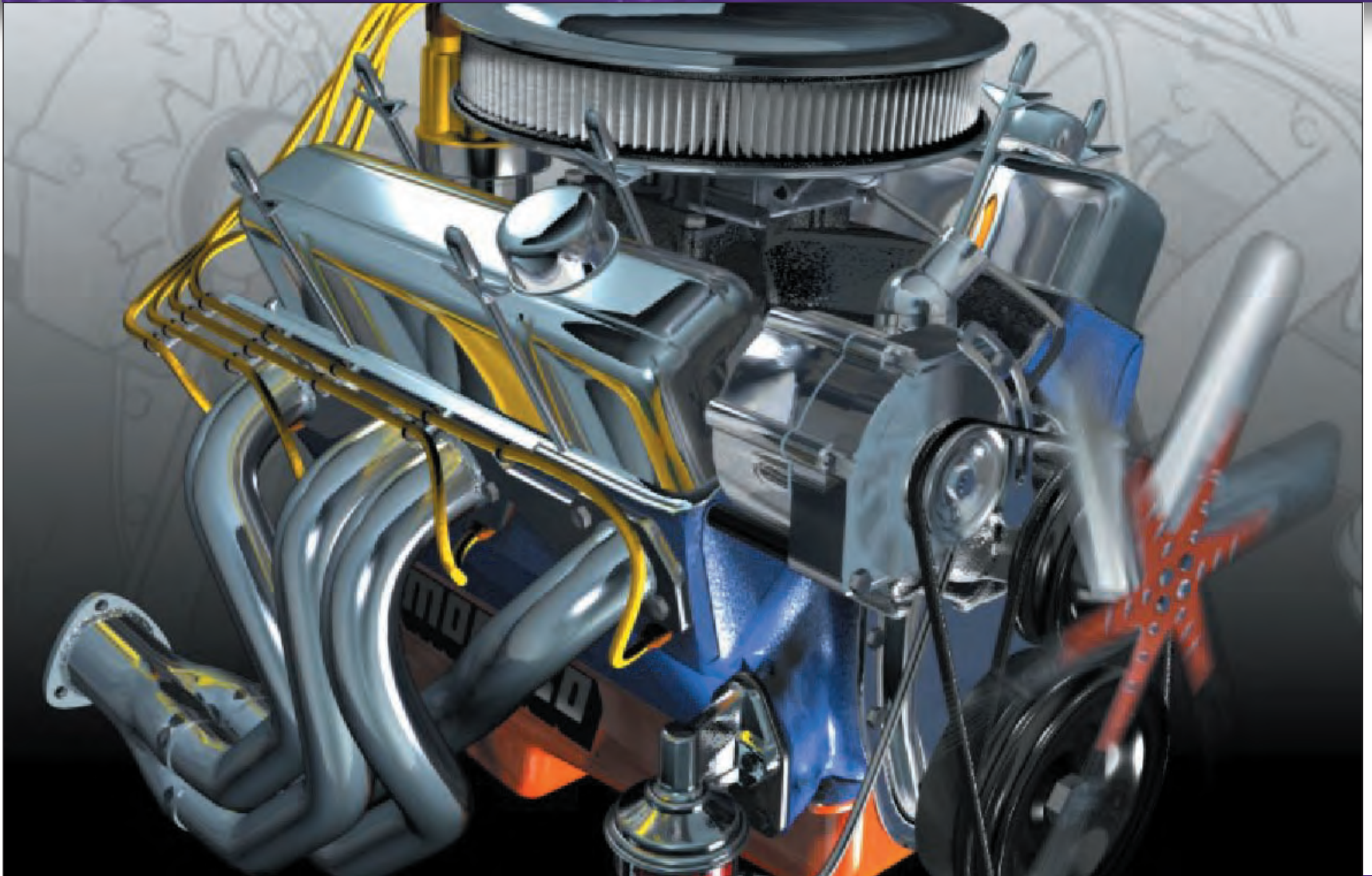
Extract a portion of a tuple using the slicing operator:

```
>>> t[2:]
('three', 6.5, 'Bob', 'some string')
>>> t[1:-1]
(2, 'three', 6.5, 'Bob')
```

## NEXT MONTH

If your love is True, we'll throw you for a loop while iterating over items of affection. In other words, we'll show you how to control the flow of your Python programs.





2D AND 3D ART

# Render with Blender

From game-building through animation to stills, this recently open-sourced app is an artist's dream, says **Jono Bacon**.



**W**ithin the Linux world, there are a number of applications that have gained a reputation for being the 'killer apps' for our lovable Linux OS. Among these reputable offerings are applications such as the *GNU Image Manipulation Program (GIMP)*, *Apache*, *Quanta*, *ImageMagick* and more. One of the lesser-known programs that is making major waves is *Blender*.

*Blender* is a free 3D modeling, rendering and compositing package that includes a huge raft of features that have been developed over the years. Although *Blender* is indeed free software, it would be wise for us to first cover how the now Open Source *Blender* got to where it is today.

## The History

*Blender* started out in life as a product that was developed for a small company called Not a Number (NaN). NaN spent time developing the *Blender* application, and a series of tools and

applications that were made available as part of the *Blender* platform. Although *Blender* was developed by a commercial company, NaN offered a free download of *Blender* available for Linux, IRIX, Windows and many other platforms.

Although *Blender* was a powerful and flexible application, unfortunately NaN could not make enough money and filed for bankruptcy. *Blender* was in a state of limbo; it was a popular free application, but could not be developed further due to the state of NaN. At this point the creator of *Blender* – Ton Roosendaal, approached the shareholders of NaN to see if a resolution could be made to make *Blender* a truly free software application by open-sourcing it. Ton came to an agreement with the shareholders and he needed to raise 100,000 Euros to make the source publicly available. Luckily the generosity of the free software and *Blender* communities pulled together and the €100,000 was raised thus sealing *Blender* as an application we can use freely forever.

It is only recently that *Blender* has made its first Open Source release – 2.26 – and the hard work of the *Blender* community is ensuring that *Blender* will continue to grow and evolve. It won't be a 'lesser known' killer app for much longer...

## The feature set

First and foremost, *Blender* is a 3D modeler. It has facilities for modeling a variety of shapes from simple primitives to complex structures such as heads and buildings. This modeling functionality is accompanied with a powerful rendering infrastructure including materials and textures and definition of materials is highly configurable. *Blender* includes a powerful lighting facility in which scenes can be lit by a variety of different types of lights, volumetric lighting and more.

Although *Blender* is very capable for still imagery, it is also extremely capable as an animation studio with a raft of facilities for animating scenes, characters and more. *Blender* includes basic frame animation, movable joint animation and many other facilities that could be realistically used for creating animated shorts. These animations can also be outputted to a variety of video formats.

One facility that many people do not realise is a part of *Blender* is its game engine. *Blender* includes a powerful engine in which interactive 3D games can be developed with little or no programming. For those codehounds among us though, *Blender* does have a Python-based scripting interface for fine tuning your games, and the game engine has been used effectively by many *Blender*heads.

## Installing Blender

OK, we are now fully aware of just how cool *Blender* is so now is the time to get it and start *Blending*! *Blender* is fairly easy application to install, but installation can vary for your distribution.

The main site for obtaining *Blender* is [www.blender.org](http://www.blender.org), but your distributor may have made a *Blender* package available for your distribution. *Blender* is on the LXF coverdiscs this issue too!

### RPM Based Distributions (Red Hat, Mandrake, SuSE etc)

**1** You can obtain a *Blender* RPM from your distributor or from one of the *Blender* packagers (see [www.blender.org](http://www.blender.org) for more details).

**2** When you have obtained the RPM, install it with:

```
rpm -i blenderpm.rpm
```

### DEB Based Distributions (Debian, Progeny, Xandros)

**1** Check with your distributor for the availability of a *Blender* DEB package.

**2** You can install the package with:

```
dpkg -i blenderdeb.deb
```

### Debian specific

**1** You can install *Blender* using the **apt-get** tool by simply typing:

```
apt-get install blender
```

## Starting Blender

Many distributions add *Blender* to the application starter menu in the desktop/window manager that you are using. *Blender* can also be started by opening a command line terminal and typing **blender**.

This will open *Blender* in its default mode which has no window decorations surrounding the *Blender* environment. Although many prefer this approach, others prefer to have window decorations surrounding the *Blender* application like any other application. You can achieve this with:

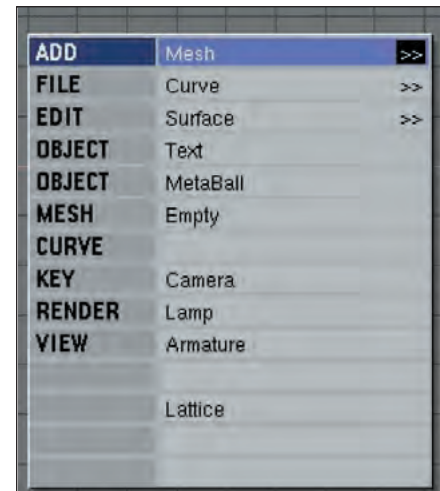


Figure 2: **The Blender Toolbox; a place to create objects in your scene. Adding a sphere to the screen is a basic operation that you'll use regularly.**

## Getting Started

### Blender basics

When you fire up *Blender* for the first time, you may be overwhelmed with the apparent complexity of the environment. Although it does indeed look complicated, rest assured the *Blender* interface when mastered is one that is both efficient and powerful.

The *Blender* interface has a number of major parts to it. These are:

- The main part of the screen with the gridlines is the Canvas. This area is where you perform the modeling and create your 3D scenes.
- The first strip of buttons immediately below the canvas is the Window Header Buttons. These buttons are used in relation to the canvas and can be used to change the views and other manipulations.
- The strip of buttons below the Window Header buttons are the Function Buttons. These buttons are used to select different functionality within *Blender*. The area below these buttons is where each function can be configured.
- Although we have focussed on the different sections of the screen, you may notice a number of other items on the Canvas part of the screen. These are:
  - The red dashed circle is the 3D cursor. This cursor is used to pinpoint where on the canvas items are created.
  - The square in the center of the canvas is the Default Plane. This flat object is always present when you begin a new *Blender* project. This is a flat square that is present on the canvas.
  - The triangle below the Default Plane is the Camera.

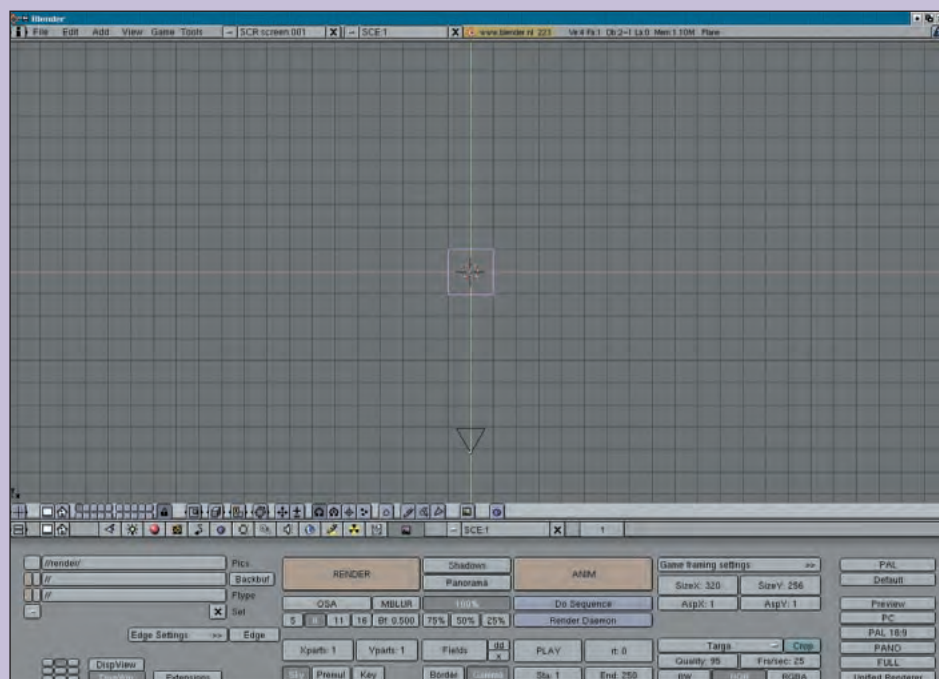


Figure 1: **What the interface first looks like when you load Blender.**



# TutorialBlender

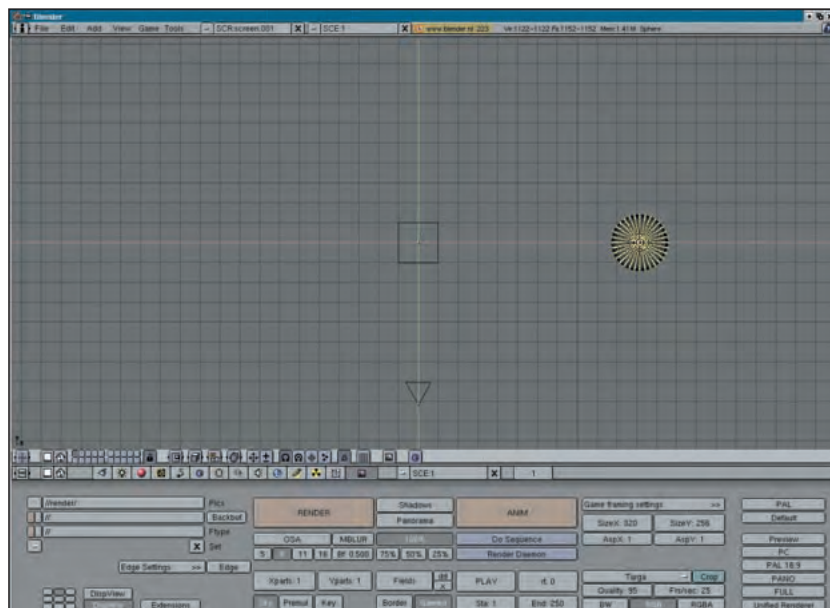


Figure 3: Moving the mouse from left to right increases the number of segments in your virtual 'orange'.

## blender -w

It is worth taking a look at the blender man page to investigate the other command line switches that can be used to adjust how *Blender* is loaded. You can see the man page with:

```
man blender
```

## Creating Objects

We will now get started at creating our very first object within *Blender*. First, ensure that you have a new project open, and that your canvas looks like **Figure 1** on the previous page. Next, move the 3D cursor to the right of the Default Plane by clicking in the area to the right of it with the left mouse button.

To create an object, press the Spacebar and window will pop up like the one shown in **Figure 2**. This is the Toolbox menu and is used to add various objects to the scene. There are a number of categories down the left side of the toolbox, and then each category has various suboptions on the right.

We will add a sphere to the scene. To do this click on the Add category in the Toolbox (this should be already selected if this is the first time you are opening the Toolbox. Next, click on Mesh and then select 'UVSphere'.

A small box will then pop up asking how many Segments you wish to use. This button is a special kind of button in *Blender* and is worth mentioning. If you click on the button and move the mouse left, the number of segments will reduce, and if you move the mouse right the number will increase. You can also type in a value by holding the Shift key when clicking on the button with the left mouse button.

Click on OK to accept the number of segments (similar to orange segments, fruit fans!) and you will then be asked for the number of rings. Again, click OK to accept the value, and you should see something similar to **Figure 3**.

## Editing Objects

Objects within *Blender* can be edited in a huge number of ways, and we will covering a number of methods over the course of this series. When we are editing objects in *Blender* we will be dipping in and out of a mode known as Edit Mode. This mode is toggled using the Tab key and switches between the object being treated as the entire object and the object being editable by editing each vertice in the object. When the object is pink it means you are in non-edit mode (ie the object is being treated as the entire object). If the object is a dark purple colour, it means you are in edit mode, but no vertices are selected. When you see the object as yellow or parts of it are yellow, those parts are the selected vertices.

We will now have a little play with the object and move and resize it. First, get out of Edit mode by pressing the Tab key until the object turns pink. You can then use the G key when hovering the mouse over the sphere and the object will move when you move the mouse. You can place the object very simply by moving the object to the place you want to put it and pressing the left mouse button.

We can now scale the object by again ensuring it is pink first, and then pressing the S key and moving the mouse to scale the

## Getting to know the Function buttons

Make *Blender* work for you



### VIEW BUTTONS

Used to adjust the view in *Blender*. We will look into this more later when we explore views.



### LAMP BUTTONS

Add, adjust and fine tune all light based objects in your scene.



### MATERIAL BUTTONS

Add and edit materials for individual objects and faces in your scenes. This feature is often combined with Textures to create realistic looking objects.



### TEXTURE BUTTONS

Textures can be applied to objects and faces as well as materials. Textures are important in ensuring that your objects look authentic. We will look into this more later.



### ANIMATION BUTTONS

These buttons can be used to set up your

objects so they can be animated using a variety of different techniques.



### REALTIME BUTTONS

These buttons are used to configure the realtime aspects of your *Blender* scenes. You can use these buttons to implement realtime processing.



### EDIT BUTTONS

Make changes to your meshes, models and objects in your scene. This is one of the most commonly used buttons and we will be looking into this area in great detail later.



### CONSTRAINT BUTTONS

These buttons allow you to attach and edit constraints to your scene.



### SOUND BUTTONS

These buttons deal with attaching sound events to your scenes and animations.



### WORLD BUTTONS

These buttons let you set up world settings such as sky colour, mist colour etc.



### PAINT BUTTONS

These buttons are used for fine-tuned painting at the smallest level



### RADIOSITY BUTTONS

These buttons control the radiosity renderer within *Blender*. We will cover this more later.



### SCRIPT BUTTONS

These buttons deal with the python based scripting interface within *Blender*.



### DISPLAY BUTTONS

These buttons are used to configure how your animations and images are displayed and where they are saved to.

## Window Header Buttons

Customise each view with its own buttons



### CURRENT WINDOW TYPE

This button lets you select what kind of information is displayed in the window. At the moment it is set to 3D space view.



### CREATE FULL SCREEN

If you click on this button the window will take up the full extent of the screen.



### HOME

Can be used to jump to the selected scene within the window. This is useful when looking at another part of the canvas within this view.



### LAYERS

This set of small buttons are different layers that can be used to build a scene up.



### LOCK LAYERS

This will be looked at in later tutorials in this series when we discuss layers.



### LOCAL VIEW

Lets us select how we look at our object.



### PERSPECTIVE MODE

Selects which perspective we can use to look at the scene. We can also select the camera view from this button.



### TOP/FRONT/SIDE VIEW

Look at the scene from the top, the side and the front.



### DRAWTYPE

Select how the object is rendered in the window and if it is to be shaded.



### TRANSLATE VIEW

Used to navigate the scene.



### ZOOM

This button can be used to zoom in and out of the window.



### ROTATION/SCALING

The next four buttons deal with rotation and scaling around different parts.



### EDIT MODE

Toggles the edit mode (equivalent of pressing Tab).



### VERTEX PAINT

This button lets you paint on individual vertexes.



### TEXTURE PAINT

Sets up the painting of textures.



### FACE SELECT MODE

Toggles the face select mode on the object.



### RENDER VIEW

This button will render this specific view.



### START GAME

Lets you start a game when using the game engine in this view.

You can now use these buttons to adjust the different settings of the window. As an example click on the Top/Side/Front view button and play with the different views. You can also use the Perspective Mode view to select the camera and see what the camera is looking at.

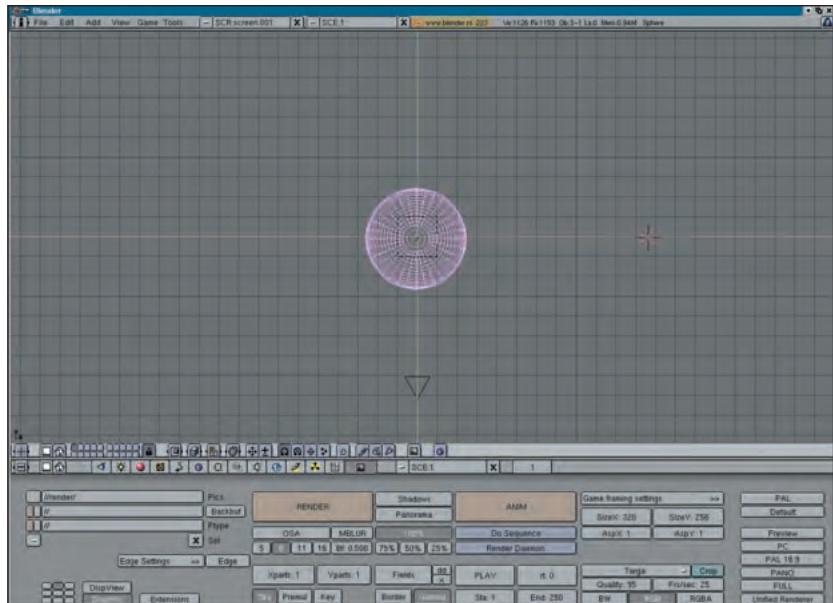


Figure 4: **Scaling** can be executed with the mouse – no need for using tables of numbers.

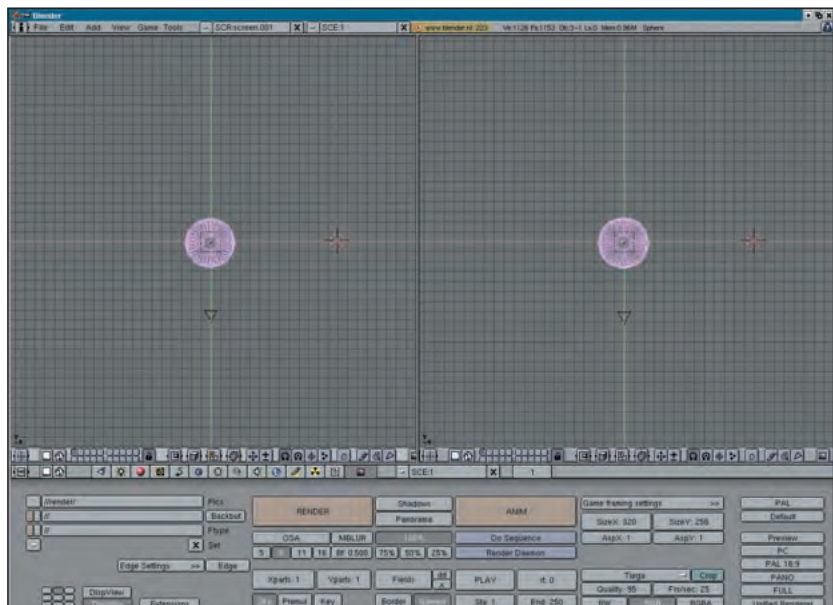


Figure 5: **Two sets of buttons** makes editing each window less confusing than having to remember which one is currently selected.

## Adjusting the views

Although we still only have a simple object on the canvas, we will now create some different views and perspectives on our scene. *Blender* can support numerous different views at the same time, and these different views can be very useful when modeling complex scenes.

To create new scenes we need to split the current screen into different views. To split the current view in half, right-click on the border between the Window Header Buttons and Function Buttons and a little menu will pop up. Select 'Split Area' from the menu and you will be able to select where the window is split. Click the left mouse button to select the split. When the screen is split into the two views you can see that each view has its own set of Window Header Buttons like in **Figure 5** above.

If you have any comments or questions on this *Blender* tutorial series, please send them in to the usual address. [lxf@linuxformat.co.uk](mailto:lxf@linuxformat.co.uk)

## 3D Acceleration

Like with most 3D based software, *Blender* can make use of 3D acceleration through the use of the *Mesa OpenGL* libraries. If you have a 3D accelerated graphics card, it is recommended that you ensure your drivers are updated and functioning correctly to get the most out of *Mesa*.

It is also important to note that 3D acceleration is not required for *Blender* to run, as software rendering will work fine.

sphere. Click the left mouse button to select the size. When you have made the sphere bigger, move it over the Default Plane using the **G** key again and your screen should similar to **Figure 4**.

## NEXT MONTH

We've taken a look at what *Blender* is, how to install it, the interface and how to create an object and adjust its views. These baby steps will be built on next month with a deeper look at views and perspectives, and begin looking at more complex modeling. Until then we suggest you experiment with *Blender* and explore what it can do.



# Answers

If you are really stuck and the HOWTOs yield no good result, why not write in? Our resident experts will answer even your most complicated problems!

## Our experts

Whatever your question is, we can find an expert to answer it – from installation and modem woes to network administrations, we can find the answer for you – fire off an email to [lxf.answers@futurenet.co.uk](mailto:lxf.answers@futurenet.co.uk) or send a letter by snail mail and it'll all be taken care of.

LXF answers guy

**David Coulson** is a networking and security guru with plenty of sysadmin experience to boot.



**Nick Veitch** is the editor of the magazine, and answers your easy questions! Or indeed anything to do with *Grub*, *LILO*, *netatalk*, vi...



**Hans Huberland** is Rackspace Managed Hosting's Linux expert. Send any Linux system admin questions to [sysadminqa@rackspace.co.uk](mailto:sysadminqa@rackspace.co.uk).



Not pretty – the JPEG compression algorithm causes artifacts and data loss, so you can't get the original back.

## Images don't match

**Q** I am stuck with a problem with my project that I have available for other Linux users to use. I have finished the first stage of a duplicate file finder which now successfully finds identical binary files in given search paths. It is still a little rough around the edges and I am planning some changes to it (interface, probably changing from Java to either Kylix or C, and sorting out the licence under which I am making it available), however the next step, image matching, is where I am completely stuck. I have searched on and off for over a year for some way to do image matching, that is find images that are similar to the eye, but completely different in their binary make up, but come up empty-handed. At the moment I am mainly interested in JPEG and

GIF images, but may later extend it to include TIFF, PNG and some others if the demand is there. I want to continue with the idea of using a type of database to store and use only a fingerprint of images (as I have done with checksums on binary files) rather than having to look at the images every time a search is done. I know this is possible because other proprietary programs do it. Do you guys have any idea on how I can do this?

*Ian B, via email*

**A** A utility such as *ImageMagick* may be useful for what you are trying to do. However, if you're looking at JPEG, GIF and PNG images, due to the compression in the first and the colour limitations of the second, you're not going to end up with the same image at any rate. There's no easy way to do what you're trying to accomplish without a

comprehensive tool which intelligently matches images. Unless you have the original pristine image available for comparison, you're likely going to be chasing a goal which is realistically out of reach.

## Modem detection

**Q** Unfortunately, after having installed Red Hat 8.0 recently, I have found that it does not auto-detect my Kingston Technologies modem. I followed your advice from *LXF37*, and visited [www.linmodems.org](http://www.linmodems.org) to search for Kingston. They don't have any real info, so I was wondering if you could give advice on probing the device to find out if the modem is actually a winmodem (strongly suspect this as it was cheapo) or if I've not set something up properly.

Also, as a little quickie, I found something in my user account

which starts a terminal window automatically when I log in, but I can't find it again, as I would like to add to it. The official Red Hat documentation and help don't seem to be too helpful. Could you tell me where I would find this?

Keith Wyse, Edinburgh

**A** It is fairly straightforward to find out if a modem is a winmodem or not, as hardware modems are detected by the kernel as a serial port. By running **dmesg** following a boot up and grepping for **ttys**, you should see something along the lines of:

```
$ dmesg | grep ttyS
```

```
ttyS00 at 0x03f8 (irq = 4) is a 16550A
```

```
ttyS01 at 0x02f8 (irq = 3) is a 16550A
```

Generally IRQ 3 and IRQ 4 are for the on-board serial ports, so any other serial ports will belong to the modem device. Chances are, if you can't find the serial port, then it's going to be a winmodem. You can also cat `/proc/pci` to find out what PCI devices exist on your system, which should give a reasonable description of the device.

Assuming you perform a graphical login with *gdm*, *kdm* or *xdm*, you can modify `~/.xsession` and add a line like:

```
xterm &
```

```
exec startkde
```

You must remember to include the **&** after your terminal program, otherwise it will block the startup of your desktop environment until you quit your terminal. Also, depending on your distribution and choice of desktop, you may be able to store session information and have it start an xterm for you without having to modify any configuration files yourself.

## Missing headers

**Q** I was made up to see the Nvidia drivers on your coverdisc. As whilst I have

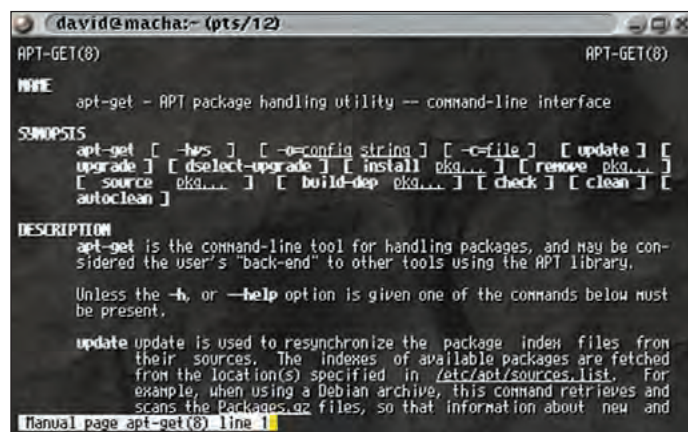
SuSE 7.3 running fairly well, the Nvidia drivers improved it. I am however, trying to get Debian 3 up and running. The problem is that the Nvidia kernel module will not compile in Debian 3. The last part when doing 'make' is as follows:

```
nv-linux.h:24: linux/modversions.h:
no such file or directory
make: *** [nv.o] ERROR 1
```

which means nothing to me. Have you any ideas why the module won't compile? The GLX part went in no problem.

John Bywater, via email

**A** It appears that you are missing the Linux kernel headers. There is a Debian package called *kernel-headers* which provides these for you, although you



**apt-get** a front-end to the *dpkg* package manager in Debian, which manages dependencies for administrators.

will need to check your kernel version with **uname -r** to ensure that you install the correct version of *kernel-headers*. Once *kernel-headers* is installed, the Nvidia kernel module should compile correctly. There is also a *nvidia-kernel-src* module in Debian which provides the same functionality as the code you're trying to compile, except it's in a nice package form, rather than source code.

## RPM commands

**Q** With regard to the Mandrake 9 feature/review in LXF35; what is really the difference between the RPM

command **urpmi <name>** and Debian's **apt-get install <name>**? It seems to me that Debian is so very easy to look after, yet when I tried version 2.2rc5, I had difficulties with (USB) printing and (cs46xx) sound output. I use Mandrake 8.0 and very much like its hardware detection and configuration ability/frontend, but in MDK 8.0 I am constantly plagued by RPM dependency issues. That's why I'm considering a newer distro. It's got to be either Debian or Mandrake, I think.

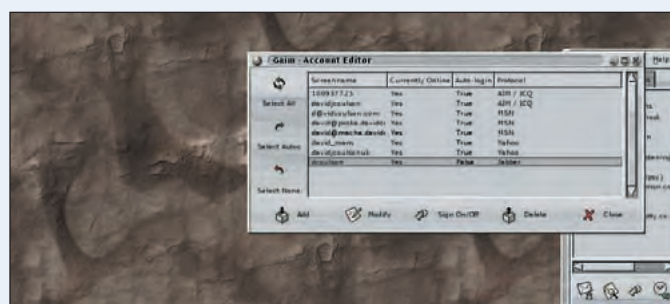
I know several people who have told me different things about



## A QUICK REFERENCE TO: INSTANT MESSAGING

For those of us who consider email to be slow, there are many instant messaging protocols out there, from those provided by corporations, such as MSN, Yahoo and AIM, to open protocols, with IRC and *Jabber* in the latter group. While many provide closed-source clients for Linux, it's not exactly useful for those running non-i386 systems, plus you need one for each protocol. If we're connected to the most common services, then we might have five or six little windows cluttering our desktop.

Somewhere down the line, there was the bright idea that having one client handle more than one protocol would be a smart idea. *Jabber* does this in some respect, since you can connect to another protocol via the *Jabber* server, however this can cause problems if



**gaim** is a wonderful Linux instant messenger client, supporting a wide variety of protocols.

the *Jabber* server is unavailable. Instead, most choose to use a client which contains protocol code for the most popular IM systems.

Two of the most popular IM clients on Linux are *gaim* and *everybuddy*. The former originally only supported AIM, but it now has plug-ins for everything from MSN to IRC, with plenty of others

in between. Both support almost all protocols known to man, along with a few others no one ever uses. Of course, since those running the servers can modify the protocols as a whim, it's worth keeping up with the updates to both of these clients. It's not uncommon for AOL to suddenly not like other clients and block them.

IRC is a little different, since it's a group based chat system, rather than a IM service. There are hundreds of clients for IRC on the Internet, both X and console based and <http://freshmeat.net> contains a comprehensive list of what is available.

On the server side, it's somewhat difficult to supply your own MSN, Yahoo or AIM services because there is no public code for the service. Since individuals have managed to reverse engineer the protocol to write clients such as everybuddy and gaim, it is practical to produce a server, although no one seems to have taken the time to do this. Both IRC and *Jabber* have Open Source servers available, so one can easily setup a private IRC or *Jabber* network.



# FREQUENTLY ASKED QUESTIONS **KERNEL**

## **FAQ I HAVE JUST INSTALLED LINUX BUT WANT TO UPGRADE THE KERNEL. HOW DO I FIND OUT WHAT KERNEL I'M ACTUALLY RUNNING?**

Most current distributions ship with 2.4 kernels, although some use 2.2 kernels instead. You can easily check what kernel you're running with the `uname` command. Simply do:

```
$ uname -a
```

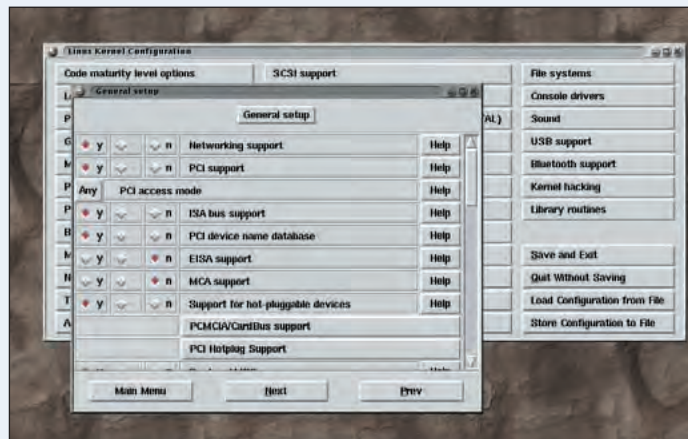
```
Linux tailtiu 2.4.19-rc1-pe-lb-brnf #1
SMP Wed Jun 26 18:48:32 BST
2002 i686 unknown
```

This system is running 2.4.19-rc1, with some extra patches. Likely your distribution creators will have applied their own patches so it won't be a simple 2.x.y kernel version.

## **FAQ WHERE DO I DOWNLOAD A KERNEL FROM AND WHAT DO I HAVE TO GET?**

You can download a tarball of the current kernel from <http://www.kernel.org>.

Generally one will simply want to grab **linux-2.4.19.tar.gz** from `linux/pub/linux/kernel`, which is the complete tarball of the kernel source. If you already have a kernel source tree available, you can obtain the patch between that and the next revision, saving users some time downloading the entire source tree.



**The Linux kernel is easily extended with kernel patches or loadable modules.**

## **FAQ I RUN THE KERNEL THAT CAME WITH MY DISTRO. DO I HAVE TO USE THEIR KERNEL OR CAN I BUILD MY OWN?**

Red Hat distributes a kernel which is patched beyond all acceptable limits. However, some of their tools and utilities fail to function properly with a kernel which does not have their patches. Likewise, some third-party patches will conflict with the contents of the Red Hat kernel, as they will be expecting a clean kernel tree.

Unless you're in need of specific kernel capabilities which render a Red Hat kernel useless, due to incompatibility, it's often good to stick

with distribution kernels. Of course, if you don't need any of the patches within the distribution's kernel, then a vanilla kernel tree should work perfectly well.

## **FAQ I'VE GOT MY KERNEL TAR FILE. HOW DO I GO ABOUT COMPILING IT?**

The general build location is `/usr/src/linux`, so you need to **cd** into `/usr/src` then untar it with:

```
tar xzf ~/linux-2.4.19.tar.gz
```

Once the kernel tarball is unpacked, you will have a 'linux' directory which contains the source code. The general routine to configure,

build and install a kernel is:

```
cd linux
make mrproper xconfig dep clean
bzliilo modules modules_install
```

First, you will need to configure your kernel. Generally keeping the defaults will result in something that will boot, but you will likely need to select other options for hardware devices, such as sound cards, SCSI interfaces and everything else which everyone won't be using. As long as there is a backup kernel available, accessible via `lilo`, then it should not be a major issue to try different options to decide which supply the required capabilities for the system.

**bzliilo** will build the kernel, copy it to `/vmlinuz`, then run `lilo` to install the new kernel. This will require `lilo` to be modified to use `/vmlinuz` as the image, rather than `/boot/vmlinuz`, although one can hack the Makefiles to install the kernel into `/boot/`.

The last two make options build the loadable modules and plonk them in the right place for the kernel to use. Once you have a kernel built with loadable module support, you can just go in and reconfigure the kernel and install the modules.

## **FAQ CAN I ADD CAPABILITIES TO THE KERNEL SOURCE TREE?**

modules: If I hard-compile everything the system will be a little "tighter" and faster – must be good? – but apparently some

## Posting to the forum The LXF online community

Got a technical question? Other LXF readers may be able to help!

The forums at [www.linuxformat.co.uk](http://www.linuxformat.co.uk) have a section dedicated to technical queries, hardware, programming languages and general help. As well as being able to call on **lxfadmin** (when there's no deadline!) and the ever-present 'anonymous', the forums are also frequented by Linux heroes like **Jeremy, Nelz, Fingers99, Rhakios, Erin** and many others brimming with knowledge and experience of using Linux in a wide variety of situations.

things *need* to be compiled as modules; and too much hard-compilation will make the kernel too big for a floppy disk. Also; if this is true, can I not create a boot-CD instead of a diskette-based one? Thank you for all your help (fingers crossed)...

James Thompson, via email

Mandrake's RPM is a different package manager to `apt-get`, which is a front-end to `dpkg`.

Debian uses *Debian Package* with the *Advanced Package Tool* as a front-end. RedHat, Mandrake, SuSE and many others use *Red Hat Package Manager* to keep track of packages on the system.

The version of Debian you tried is particularly old and supported only the 2.2 kernels. The current 3.0



Debian lacks some of the graphical niceties of Mandrake that novices like.

There are many patches available for the Linux kernel which enable it to do other things. One of the main projects which distribute combined kernel patches is WOLK, the Working Overloaded Linux Kernel. Rather than having to apply individual patches one at a time, WOLK provides a single patch which adds a whole bunch of different patches, which saves time for experienced users, and saves hassle for novices.

WOLK is available from <http://sf.net/projects/wolk/>. There is also the -ac kernel series, distributed by Alan Cox. These contain many updates which have not yet made it into the mainstream kernel, along with some basic fixes for things broken in existing kernels. Many people use -ac kernels instead of the regular kernel tree, although as with distribution kernels, other kernels may object to being applied to an -ac tree.

### FAQ ARE 2.5 KERNEL STABLE ENOUGH TO USE, OR SHOULD I WAIT FOR 2.6?

2.5 kernels are getting better, but they're certainly not ready for end-user usage right now. Each release still has numerous flaws which are fixed in follow up releases. If you value your data and want to get stuff done, stay well away from 2.5 kernels!

release of Debian has improved kernel support, which should help you with your USB and sound issues. Debian is a far more complex distribution to install and manage than Mandrake as it lacks many of the nice GUI aspects which make the latter easier for a beginner. RPM dependencies are some things which are not going to go away. Package managers are designed to ensure dependencies are met, although apt generally does a nicer job than RPM due to having an intelligent system to ensure that package dependencies are met correctly.

As for your modules question – there is really no performance benefit in compiling everything into a kernel, other than that it won't have to load it from disk once when it's first needed.

There are very few things which need to be built as modules, and certainly everything in the current 2.4 kernel tree can be compiled into the kernel without any problem. The only noteworthy problems are with the Linux Userland File System (<http://lufs.sf.net>) and the ppp\_mppe.o module from PoPToP [www.poptop.org](http://www.poptop.org). Compiling everything into the kernel will make it larger than a floppy, but you need to question what you need in a rescue floppy. Chances are you're not going to need sound or USB support, so you can save some space on the disk by only compiling in what you really need to get the system up and running to a state when you can recover the regular boot loader. One could indeed make a boot-CD, but if the kernel is 3MB, it's a nice waste of a CD-R just to have sound support when you boot off it.

### Mozilla muddle

**Q** I am currently using Mandrake 8.2 (from Linux Format coverdiscs). I seem to have a slight problem with Cut and Paste.

Often when I am browsing with Mozilla I may see something that I copy and paste into a separate text editor either the *Advanced Editor* or *KEdit*. Now Copying from Mozilla to either of these editors works fine, and I can copy and paste within these editors and between these editors with no problem. BUT if I want to copy from either of these editors and paste to Mozilla, it does not work.

What I see instead in the Mozilla buffer is the last thing that Mozilla copied. At first, I thought that maybe this was a Mozilla bug, BUT I found that I can copy and paste fine (in either direction) if I use *GVIM* and Mozilla.

So then I experimented with *GVIM* and *KEdit* and I found that *GVIM* relates to *KEdit* the same way Mozilla does. That is to say, I can copy from *GVIM* and paste it to *KEdit* but not vice-versa.

Any thoughts on this?

Carl Hoffman, via email

**A** It sounds as if your consistent factor is *KEdit*. Most editors behave like Windows editors and require you to do <Ctrl-C> before dumping the selected text into the X clipboard buffer for pasting



Long-ago 'last updated' info indicates a problem or a migration to SourceForge.

elsewhere. No doubt, when you then try to paste into *KEdit*, it will paste it directly into the program itself, rather than to its own clipboard and then expect you to do a <Ctrl-V> to paste it into the document.

A similar effect is seen with *StarOffice* and *OpenOffice*, so it's not unusual for UNIX applications to behave this way. However, it is inconsistent with other X programs making it somewhat irritating to those of us who are used to the way the X clipboard functions.

### Crashing WM

**Q** I have installed Star Office in Mandrake 9.0. It has a tendency to make the window manager crash when ever I try to open a template or an MS Office document. This happens whether I use KDE or GNOME. However, *IceWM* is fine – which makes the 9/10 you award it in February's magazine get my hearty endorsement!

I have tried the Mandrake Club users' forum to get an answer, but no one has come up with anything. Brian Blandford, via email

**A** Sounds like a bit of a crazy one to us. There's no obvious reason why the window manager should die when you perform an operation within an X client, so there isn't anything specific which you can do to further debug this problem. Short of running *kwm* under *gdb* and waiting for it to segfault, you're probably going to rely on someone else being able to reproduce this problem. Of course, as it works fine under *IceWM*, you could

run KDE or GNOME applications under *IceWM* without any problems.

### Trig tribulations

**Q** I'm having a problem with gcc 2.95.3 20010315 (SuSE)

When trying to compile the attached source using the command gcc 09104.c I get the following compiler errors:

```
/tmp/ccboQBqX.o: In function 'main':
/tmp/ccboQBqX.o: undefined
reference to 'sin'
/tmp/ccboQBqX.o: undefined
reference to 'cos'
/tmp/ccboQBqX.o: undefined
reference to 'tan'
```

collect2: ld returned 1 exit status

I used #include <math.h> which should give me access to trig functions. I used the -M option with gcc to see if it was looking in the right places for header files, it was. The file math.h and a bunch of other stuff is there in /usr/include. What am I doing wrong?

I managed to compile the dreaded 'Hello World' program with no problems. Will I have to write my own trig functions? I've decided to abandon C for the time being and have switched to Perl.

Pete Czerwinski, via email

**A** If you're wanting to use C math functions, you'll need to enable libm, which actually supplies those functions.

This can be accomplished by adding -lm to your command line. The headers declare the functions for the compiler, but the library is still required for compilation and execution.





## Sendmail stopped

**Q** I hope you can help? I am running Red Hat 7.3 and recently upgraded sendmail to 8.11.6 after CERT announced the vulnerability that exists within all versions prior to 8.6.10. I actually did a downgrade according to the documentation, I think!

I have two major issues after doing this upgrade. Firstly, my server no longer accepts connections, *sendmail* is started but I cannot even use telnet to port 25 so it is not receiving any mail.

Secondly, every email the mailing program try to send I get this error:

Warning: Option: CACERTPath requires TLS support

Warning: Option: CACERTFile requires TLS support

Warning: Option: ServerCertFile requires TLS support

Warning: Option: Serverkeyfile requires TLS support

Warning: Option: ClientCertFile requires TLS support

Warning: Option: Clientkeyfile requires TLS support

The documentation I have mentions creating a new *sendmail.cf* file using *m4* but I am not sure how to proceed with this. Also how do I get rid of these errors?

*Tom Berkman, via email*

**A** The *m4* macro pre-processor program is used by *sendmail* V8 to produce a *sendmail* configuration file. The *m4* program will produce the */etc/mail/sendmail.cf* configuration file by processing a file whose name ends in *.mc*, usually *sendmail.mc* situated in */etc/mail*. Although it is quite acceptable to make changes to *sendmail.cf* it is not very straightforward. The *m4* macros make it far easier to make modifications to *sendmail*.

It appears that the RPM upgrade of *sendmail* has replaced your *sendmail.cf* file with a default *sendmail.cf* in which all external connections are closed. You need to edit the file */etc/mail/sendmail.mc* and

look for a line that reads:

```
DAEMON_OPTIONS(`Port=smtp,Addr=127.0.0.1, Name=MTA')
```

This forces *sendmail* to only listen on 127.0.0.1. To comment this line out you will need to place a *dnl* in front of this line so that it reads:

```
dnl DAEMON_OPTIONS(`Port=smtp,Addr=127.0.0.1, Name=MTA')
```

Do the same with the following lines as you are most probably not using the certificates in *sendmail*:

```
define(`confCACERT_PATH',`/usr/share/ssl/certs')
```

```
define(`confCACERT',`/usr/share/ssl/certs/ca-bundle.crt')
```

```
define(`confSERVER_CERT',`/usr/share/ssl/certs/sendmail.pem')
```

```
define(`confSERVER_KEY',`/usr/share/ssl/certs/sendmail.pem')
```

```
define(`confCLIENT_CERT',`CERT_DIR/cert.pem')
```

```
define(`confCLIENT_KEY',`CERT_DIR/key.pem')
```

You are now ready to use *m4* to create the *sendmail.cf* file. Issue the following command:

```
m4 /etc/mail/sendmail.mc > /etc/sendmail.cf
```

This will generate a new *sendmail.cf* file from the *sendmail.mc* file. If you are modifying a working configuration I would highly recommend changing the last argument to something like

*/etc/sendmail.cf.new* and backing up the old file in case you break something inadvertently

## Student server

**Q** First off I just want to say GREAT magazine. I'm a college student in the USA studying web development. I have decided after some rummaging around the web that I would really like to run a Linux web server (with Perl, PHP, MySQL, ASP support, etc). But I know very little about Linux and need some help! I have searched online and tried to find a book or tutorial on how to setup a web server in Linux, but haven't found anything that looks good. I need something along the lines of a *Dummies Guide To...* type set of instructions. If you have any suggestions or can help out in anyway please let me know.

*Michael, via email*

**A** Welcome to Linux Michael. You will find that *Apache* generally uses a single flat configuration file which is divided into several sections. This file is usually named *httpd.conf* and is located somewhere under the */etc* directory, in newer versions of Red Hat the file is */etc/httpd/conf/httpd.conf*. This file has many directives which allow *Apache* to behave in a myriad of ways. An excellent reference guide to these directives is O'Reilly's *Apache - The Definitive Guide* ISBN 0-5960-0203-3. There is also excellent searchable online documentation at <http://httpd.apache.org/docs/>. The main *httpd.conf* file is usually very well documented with comments too.

If you choose to install *Apache* when you have installed Linux you will probably find that it is serving a default page on port 80. You can verify this by connecting to the server from a browser on the local machine and then remotely. You can change the directory from which pages are hosted by editing the **DocumentRoot** directive in *httpd.conf*. Remember that any documents you create should be readable by the user *Apache* runs as.

There are many modules available to support the requirements you have mentioned. You need to be sure *Apache* has been compiled with DSO support (most modern distributions do this). In *httpd.conf* you can use the **LoadModule** Directive to load a module for whichever feature you have chosen to install. Most distributions will install Perl and PHP, you will need to look for an after market add-on for ASP such as *ChilliSoft ASP* or its open source alternative *Apache::ASP*.

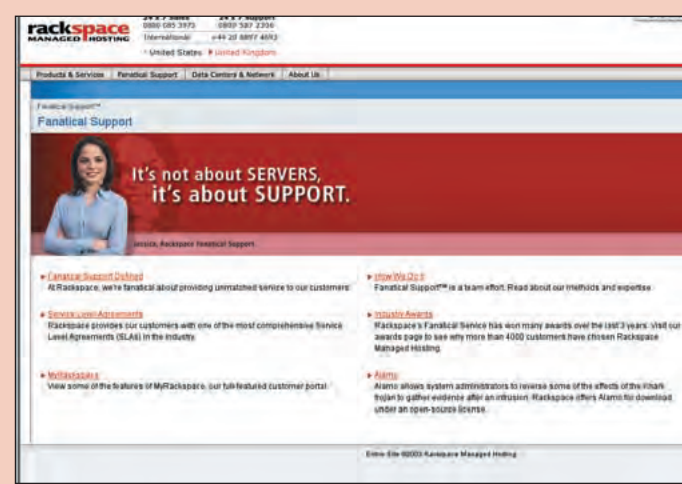
You have not mentioned virtual hosting but it's something worth mentioning. Due to IPV4 addressing limitations the most common method is name-based virtual hosting. The directive to enable name-based virtual hosting is **NameVirtualHost**, it is followed by the IP address (or \* for all interfaces) and optionally the port eg:

```
NameVirtualHost 10.2.10.33:80
```

This will enable *apache* to serve virtual hosts from this IP and port. You can now configure virtual hosts using the **<VirtualHost 10.2.10.33:80>** and **</VirtualHost>** tags. Any directive between these tags will apply only to that virtual host. Remember that the

## About Rackspace Managed Hosting

Rackspace Managed Hosting™ provides complete managed hosting solutions with Fanatical Support to small to medium-sized customers as well as large enterprises. All customer platforms include state-of-the-art data centres, customised servers, burstable connectivity, 99.999% uptime SLA, a dedicated account manager, instant emergency response and access to live expert technicians 24x7 for support of all hardware and core software. Founded in 1998 with locations in San Antonio, Texas and London, the company manages over 7,000 servers for customers in more than 80 countries. For more information, visit [www.rackspace.co.uk](http://www.rackspace.co.uk)



## WIN A SHARP ZAURUS PDA FROM RACKSPACE MANAGED HOSTING

[www.rackspace.co.uk](http://www.rackspace.co.uk)

**Every month, the best question** related to Systems Administration that a LXF reader sends in wins a Sharp Zaurus SL-5500 running Linux.

With industry support from the likes of IBM, SAP, Oracle, and Sybase, the Zaurus features a QWERTY keyboard behind a sliding cover. The 65,536 colour screen is 240x320 resolution. An Intel® StrongARM® 206MHz processor and 64MB memory makes the Zaurus ideal for downloading software or playing MP3 and MPEG1. The docking station that fits into its USB connector, allowing users to synch with their desktop,

schedules, address book and ToDo lists. The email app supports POP3 or IMAP email servers. It offers the ability to read, write and send emails and browse the Internet with a compatible modem card, or using the infra-red port and a mobile phone; or any replied, forwarded, or composed emails are transferred to your outbox the next time you synch. SD and Compact Flash card slots allow upgrades and addition of software.

Email all your sysadmin questions and problems to Hans Huberland: [sysadminqa@rackspace.co.uk](mailto:sysadminqa@rackspace.co.uk) Every published query = a Rackspace T-shirt!

It's only 74mm wide by 138mm long and a svelte 18mm thick – so the Sharp Zaurus really is Linux in your pocket!



arguments for **NameVirtualHost** and **<VirtualHost>** must match exactly.

Here's an example:

```
NameVirtualHost 10.1.1.10
```

```
<VirtualHost 10.1.1.10>
```

```
ServerName domain.com
```

```
ServerAlias www.domain.com  
support.domain.com
```

```
DocumentRoot /var/www/sites/  
domain.com/html
```

```
</VirtualHost>
```

Above I have specified a server name. This is the domain which will match the host header in the request from the browser. **ServerAlias** allows you to specify other names which this virtual host will respond to. Any directives you specify within the **<VirtualHost>** block will override the defaults set up elsewhere in the httpd.conf file. A **DocumentRoot** has been specified here to point apache to a different path to serve the pages for this virtual host.

This may seem like a lot to take in at once but bear in mind that you will probably have a working configuration from your distribution which you can experiment with. I have only covered a handful of the directives which are available but with these you should be able to modify the basic behaviour and get yourself started.

### Permissive PHP

**Q** I would like to know if there is a correct approach to setting directory/file permissions for PHP programs that reside in my webserver directories.

I am using SuSE 8.1, and it created a directory structure of `/srv/www/htdocs`. For instance, I have the popular phpgroupware program located at `/srv/www/htdocs/phpgroupware`.

The install instructions are usually sparse for PHP programs; they usually tell one to untar or copy the directory to the webserver and set the proper permissions. I typically use *Konqueror* to extract the tar kit in `/home/winter` and copy it using *File Manager – Super User Mode* to `/srv/www/htdocs`. Next, I invoke *Mozilla* and type:

```
http://home7/phpname.php.
```

Sometimes I get file permission errors, so I simply use file 'Properties' of *Konqueror* to set ownership to root/root and every access permission of every file to 666. But content management php programs that create files in their subdirectories do not like this, and they refuse to operate properly. What should I be doing to properly set the file ownership and access permissions?

*Kevin Winter, via email*

**A** Using a default SuSE 8.1 installation PHP is implemented as an *Apache* module and PHP pages inherit the permissions of the *Apache* user (in SuSE this is the user **wwwrun** but in other distributions this is normally *apache*, *httpd* or *nobody*). As long as the *Apache* user has permission to read these files the pages should

display correctly. I would not recommend setting permissions to 666 as this gives anybody access to modify your files.

I would never give more than read access to anyone who does not own the file or belong in the group that owns it. I would suggest setting the permissions to 644 and you should be able to give ownership to any user on the system and still be able to run the php pages.

There will be exceptions to this but the program's docs should cover this (remember often the online docs are far better than the readme files in a tarball). The extracted tarball should also set all the correct file permissions for you. You may just need to change the ownership.

### DNSreport.com

**Q** I have a Red Hat 8.0 server with one primary domain. A friend of mine recommended I check out [www.DNSreport.com](http://www.DNSreport.com) who perform a variety of useful tests on the DNS records and the server itself. Everything went through fairly well but my domain failed on one test.

The following is from DNSreport.com: **ERROR: One or more of your mailservers does not accept mail in the domain literal format (user@[0.0.0.0]).**

**Mailservers are required RFC1123 5.2.17 to accept mail to domain literals for any of its IP addresses.**

I'm not sure how to go about fixing this or if it's even worth fixing.

*Legolas, via email*

**A** The ability to use domain literals (ie using [s and ]s) to specify the IP address of a mail server and bypass normal DNS mechanisms is required by RFC1123. For security and for spam prevention reasons, not all mail servers are configured with it enabled by default. If you would like to have your *sendmail* daemon server accept mail sent to it in this way, you can add a line containing only **[10.10.10]** to `/etc/mail/local-host-names` where **10.10.10** is the IP address you would like *sendmail* to listen on.

### Memory-intensive

**Q** When I run the free command my system always reports almost no free memory, even if there is almost nothing running. There is usually plenty of swap available. If I start some memory-intensive programs no more swap space gets used and they seem to run pretty quick. Is this a bug? Why do I have no free RAM?

*Jedi Knight, via email*

**A** In Linux any 'unused' memory is assigned to buffers and caches. This improves the system's efficiency and the space is released as soon as a program requires it. You can get more detailed info by executing:

```
cat /proc/meminfo
```



# Answers

```

david@macha:~ (pts/12)
$ cat /etc/resolv.conf
search i.davidcoulson.net dmz.davidcoulson.net davidcoulson.net
nameserver 10.2.1.3
nameserver 10.1.1.5

"/etc/resolv.conf" [readonly] 3L, 104C
1,1 811

```

`/etc/resolv.conf` is used to configure *glibc*'s resolver, which performs DNS lookups for applications such as web browsers and mail clients.

## Nvidia niggles

**Q** I have been having this problem for two months now and have yet to find a solution. First a list of my hardware and software so I may help you to help me:

Asus a7v333 512 DDR +1700XP  
GeForce 4 8X 4200 (also Asus).  
Software-wise... Running MDK 9 pretty much default and I'm using the latest Nvidia drivers (provided BTW by your coverdiscs, thank you very much!)

OK... Here is the problem. Though the Nvidia driver both Kernel and GLX install when I run `nv_check` it gives me this output:

```

the NVdriver kernel module does
not seem to be loaded. The 3D
drivers will not work without it...
please compile it for your
kernel, set it up to insert the module
on boot, then run this script again.

```

Remember if you are having trouble compiling the module, try adding `-D_LOOSE_KERNEL_NAMES` to the Makefile.

I have searched every forum you can think of (including LXF), and have not found a solution to this problem. I have even tried un installing MDK and reinstalling again. I have also tried with an old GeForce 2MX400 and the same output from `nv_check`. Don't know what else to do and really hate switching to XP just to play *Half-Life* or *UT2k3*. Oh, I have also listed the loaded modules and *Nvidia.o* is loaded as well. Please Help!  
*Digit0, from the forum*

**A** Thanks for the hardware and software details – it *does* help! The `nv_check` script will check for 'NVdriver' in `/proc/modules`. However, you appear to have *Nvidia.o* loaded instead. You may wish to locate the *Nvidia.o* file in your `/lib/modules` directory then rename it to *NVdriver.o* to ensure the `nv_check` utility can detect it. The `nv_check` script is simply a *bash* script, and it's fairly straight forward to check the code and figure out what it's trying to do. Hopefully once that you have *NVdriver* loaded, rather than *Nvidia*, then the check should return a positive result and allow you to play the games.

## A few questions...

**Q** I could put these headings in different letters, but I really don't know where to go with them on an individual basis! My SuSE support has expired, I'm confused as to which distro does what, and about the time I finally get settled into which programming language I would like to "hobby" with, my pot gets stirred.

### 1 SuSE 7.3

I purchased a copy of SuSE shortly after its release. The install was fairly trouble-free, even for a newbie like myself. But, this was just the base system with X Windows. This install didn't get me surfing. I did find that my WinModem was just not going to play nice. I needed to find a driver. Voila, a Linux driver for a Conexant winmodem. Set the driver on the desktop, click, boom, installed. Impressive, it even dials out, after some prodding in *Yast1*.

The current source of my distress with all of this is, it won't let me surf. I have picked the dial-up provider off of the list, given the phone number, and adjusted the settings that I was able figure out. It really seems like something is turned off. Would it be possible to give a dunce like me a step-by-step explanation for getting this mess straightened out?

### 2 Flavour of Linux

To add insult to injury with the above section, I made the basic mistake of trying to install Debian first. I broke down and purchased "Debian GNU/Linux Unleashed 2.1" from the "get me outta here" bin at Borders books. I think I paid \$10.00 for the book/CD combo. But, after failure, after failure, I found out that my video card wasn't happy with X Windows. (It took a failed install of SuSE to figure this out).

Now I get to the question: Which distribution of Linux will play nice with my win modem, surf nicely, give a basic 'Plug & Play' functionality, and an install package that truly works out of the box?

I had to find the "magic" button in the SuSE installation that really allowed me to partition the hard drive the way I wanted. I like *cf-disk* in the Debian package much better than SuSE's. I can tell you how many Meg's should go to which partition, but the sectors thing is really just a shot in the dark for me.

SuSE likes to fly onto the computer with X Windows, but about the time that the Debian installer is trying to load X Windows things get real bad, all the way down to me having to unplug the computer from the wall, because it is trapped in a loop and it won't let me out. Every time; 2.1, 2.2r2, 2.2r5, 2.2r6. You can forget 3.0. *Jigdo* and I refuse to get along together. I can't even get *jigdo* to work with an LXF DVD, and I have 3-1/2 other computers that don't have DVD ROM drives.

Neither distribution has ever gotten me on the Internet. I did find out that my life would have been easier with a serial modem, but that would really take all of the fun out of my frustration. But, with my luck, I still wouldn't be able to surf, so I stick it out with the winmodem.

*John R. Klaus, Bartlett, TN, USA*

**A** To answer your questions in the same order in which you asked them:

**1** Without any information as to what exactly does not work, we can only make guesses as to the problem. The first thing to check is `/etc/resolv.conf` to ensure that the nameservers are selected correctly. Usually these are sent to your system during the PPP negotiation with your ISP, but it requires a specific option to be set when the PPP session is started. You may also be missing a 'default route', which ensures that traffic not destined for your LAN is sent out of the dial-up connection.

**2** I would suggest that you look at Mandrake or Red Hat as alternatives to SuSE. Both have good support for various video adaptors, plus they have nice graphical configuration utilities. We've never had any problems with Debian's installer, although it might be a specific issue with the video card you are using. You didn't state the specific issues you've had with *jigdo*, so we can't help you out there, but a quick search on Google with any error message your seeing should help you find an answer. [LXF](#)

## Submission advice

We are happy to answer all sorts of Linux related questions. If we don't know the answer, we'll find out for you! But in order to give you the best service, it helps a lot if you read the following submission advice.

- Please be sure to include any relevant details of your system. 'I can't get X to work' doesn't really mean anything to us if we don't know things like what version of X you are trying to run, what hardware you are running on.
- Be specific about your problem. Things like 'it doesn't work' or 'I get an error' aren't all that helpful. In what way does something not work? What were you expecting to happen? What does the error message actually say?
- Please remember that the people who write this magazine are NOT the authors or developers of Linux, any particular package or distro. Sometimes the people responsible for software have more information available on websites etc. Try reading the documentation!

We will try and answer all questions. If we don't answer yours specifically, you'll probably find we've answered one just like it. We can't really give personal replies to all your questions.

**WRITE TO US AT:**  
Linux Format, Future Publishing, 30  
Monmouth Street, Bath BA1 2BW or  
email: [lxf.answers@futurenet.co.uk](mailto:lxf.answers@futurenet.co.uk)

# missed one?

## LINUX FORMAT BACK ISSUES

Every month *Linux Format* brings you the best tutorials, the essential reviews and the latest news. But if you've missed out on a must-read feature or a vital programme from our expertly compiled CDs and DVDs, order your back issue NOW! And remember, you need never miss an issue of your favourite Linux mag, if you subscribe to *Linux Format* (see overleaf for more details).



**April 2003**

Product code:  
**LXFB0039**(cd)  
**LXFD0039**(dvd)

**DVD HIGHLIGHTS:**  
CensorNet, K3b, VIPs, FrozenBubble, Linux Open Media Box, IDMS Linux, WhiteDune, OBM, AstaroSecurityLinux

**MAGAZINE FEATURING:**  
Customising Linux, Evesham Linux PCs, Open publishing with Wiki, Linux PIMs and PDAs, SME Server review, File Managers Roundup, Serial Networking, GnuCash

**CDs HIGHLIGHTS:**  
SME Server, TightVNC, GNOME 2.2, VOPC CoyoteLinux, UT2003 update, Blender, Qt, BootEverywhereLinux



**March 2003**

Product code:  
**LXFB0038**(cd)  
**LXFD0038**(dvd)

**DVD HIGHLIGHTS:**  
Python, The GIMP, Lycoris, Yellowdog, Knoppix, Cygwin, DansGuardian, Senken, AbiWord, X Northern Captain, Eclipse

**MAGAZINE FEATURING:**  
You, Copyright & The Law, X apps on Mac OS with Fink, SVG versus Flash, Audio Editors roundup, Python tutorial, GIMP secrets, SuSE Office Desktop examined

**CDs HIGHLIGHTS:**  
KDE 3.1, UNIX Tools For Windows, Konstruct, phpMyAdmin, X Net Strength, AutoUpdate



**February 2003**

Product code:  
**LXFB0037**(cd)  
**LXFD0037**(dvd)

**DVD HIGHLIGHTS:**  
IPCop, Postnuke, Llypynd, Squid, Allegro, Cocoon, Fandango, gPhoto, Xrmap, Ted, AdvancedPHPDebugger, Flightgear updates

**MAGAZINE FEATURING:**  
Lindows on test, Serial ATA, Window Managers roundup, Arkeia review, Xft2, build your own Linux backup box, Java compiling, Jool Kwartz review, New Larger Linux Pro mag

**CDs HIGHLIGHTS:**  
Yoper, KOffice 1.2.1, Parted, DOM Tooltip, Guarddog, GL-117, PythonImagingLibrary



**January 2003**

Product code:  
**LXFB0036**(cd)  
**LXFD0036**(dvd)

**DVD HIGHLIGHTS:**  
Evolution PIM/calendar/email client, Freevo personal video recorder, OpenZaurus, PilotLink

**MAGAZINE FEATURING:**  
Optimisation tips and tricks, Netbox Cubit review, build your own desktop, Xandros, Cubit, CORBA, Realsoft 3D, Instant Messenger roundup

**CDs HIGHLIGHTS:**  
Racer, Freeduc, Linux From Scratch, GNOME Meeting, Opera 6.1, Film GIMP, phpOpenTracker, Tuxpaint, Snort



**Christmas 2002**

Product code:  
**LXFB0035**(cd)  
**LXFD0035**(dvd)

**DVD HIGHLIGHTS:**  
Mandrake 9.0 - 3 CD download edition, jEdit, BastilleLinux, Phoenix, VideoLAN, KDE 3.0.4

**MAGAZINE FEATURING:**  
Huge C++ IDE RoundUp, LXF Awards nominations, gadgets for Xmas, User Mode Linux, compiling code, process accounting

**CDs HIGHLIGHTS:**  
3CDs, including 2CD Mandrake 9.0, SpamX, Stellarium, Krename, ImagePress, Nogger, DigitalMusicCentre



**December 2002**

Product code:  
**LXFB0034**(cd)  
**LXFD0034**(dvd)

**DVD HIGHLIGHTS:**  
KDE 3.1beta2, Scribus, Smoothwall2, Movix, GNUPrivacyGuard, GCompris, Drip

**MAGAZINE FEATURING:**  
Distro reviews of SuSE 8.1, Red Hat 8.0, Lycoris and a mini-distro RoundUp, scanning, PHP acceleration, Linux Expo UK 2002

**CDs HIGHLIGHTS:**  
Unreal Tournament 2K3 (Demo), DemoLinux, Cardfile, RUTE, OoDictionaries, Mozilla binary packages

Other back issues also available, please call for further information, giving the issue number and/or date required

# To order, call the back issues hotline on **0870 4448645**

or **+44 1458 271178** for overseas orders

Remember to quote the issue code shown above and have your credit card/switch card details ready.

**Prices (inc post and packing):** CD – UK£5.99,  
US & Eur £6.99, rest of world £7.99  
DVD – UK £6.99, US&Eur £7.99, rest of world £8.99.

# LINUX

www.linuxformat.co.uk **FORMAT**



# SUBSCRIBE AND

## ON A YEAR'S SUPPLY OF LINUX FORMAT

# That's 3 issues free!



Title \_\_\_\_\_ Initials \_\_\_\_\_ Surname \_\_\_\_\_

Address \_\_\_\_\_

Postcode \_\_\_\_\_ Country \_\_\_\_\_

Email address \_\_\_\_\_

Daytime telephone number \_\_\_\_\_



Media with passion  
Future Publishing Ltd, Cary Court,  
Somerton, Somerset, TA11 6BR

Instruction to your Bank or Building Society to pay Direct Debits



Originator's Identification Number

7 6 8 1 9 5

Please send this form to address shown

1. Name and full postal address of your Bank or Building Society branch

To: The Manager

Bank/Building Society

Address

Postcode

2. Name(s) of  
account holder(s)

3. Branch sort code

(from the top right-hand corner of your cheque)

4. Bank or Building Society  
account number

5. Instruction to your Bank or Building Society

Please pay Future Publishing Direct Debits from the account detailed on this Instruction subject to the safeguards assured by the Direct Debit Guarantee. I understand that this instruction may remain with Future Publishing and if so, details will be passed electronically to my bank or building society.

Signature(s) \_\_\_\_\_ Date \_\_\_\_\_

Ref No (Office use only)

Banks and Building Societies may not accept Direct Debit Instructions for some types of account.

You may photocopy this blank form and then complete it. You cannot fax or e-mail Direct Debit forms.

QUOTE ORDER No. **LXFP40**

# LINUX FORMAT

THE UK'S BEST-  
SELLING LINUX  
MAGAZINE

Please choose your method of payment **1** or **2**

**1.** ☐ Direct Debit (UK only – double CD Edition) – I wish to pay £27.45 every 6 months

or ☐ Direct Debit (UK only – DVD Edition) – I wish to pay £29.95 every 6 months

**Please complete the form on the left**

I understand that I will receive 13 issues during the next 12 months.

**2. Cheque/credit card 13 issues**

	CD	DVD
<b>UK</b>	<input type="checkbox"/> £54.90	<input type="checkbox"/> £59.90
<b>Europe (including Eire)</b>	<input type="checkbox"/> £78.90	<input type="checkbox"/> £83.90
<b>North America</b>	<input type="checkbox"/> £78.90	<input type="checkbox"/> £83.90
<b>Rest of the World</b>	<input type="checkbox"/> £91.90	<input type="checkbox"/> £96.90

☐ Cheque (payable to Linux Format. Sterling cheques drawn on a UK account.)

☐ Visa

☐ MasterCard

☐ American Express

☐ Switch

Issue No/Valid date \_\_\_\_\_

Card No \_\_\_\_\_ Expires \_\_\_\_\_

Signature \_\_\_\_\_ Date \_\_\_\_\_

**UK READERS RETURN THIS COUPON TO:**

Linux Format, Subscriptions, Future Publishing Ltd, Freepost BS4900, Somerton, Somerset, TA11 6BR.

**OVERSEAS READERS RETURN (POSTAGE PAYABLE) TO:**

Linux Format, Subscriptions, Future Publishing Ltd, Cary Court, Somerton, Somerset, TA11 6TB, UK

**WEB:** [www.linuxformat.co.uk](http://www.linuxformat.co.uk) then click on **Subscribe**

☐ Please tick this box if you would prefer not to receive information on other offers.

**ORDER HOTLINE ☎ 0870 444 8645**

**OR FROM OVERSEAS ON +44 870 444 8645**

**OFFER ENDS: 22.05.03**

# SAVE £24.47!\*

## – THE LEADING LINUX MAGAZINE



**13 x**

**= SAVING OF £24.47\***

- **THE ESSENTIAL** Linux read for all Linux users – can you afford to miss out?
- **EVERY ISSUE** comes with the ideal supplement for IT professionals – *Linux Pro* covers trends, technology, security issues, industry comment and thought-provoking case studies
- **GUARANTEED DELIVERY** direct to your door so you never miss an issue!

### RISK-FREE GUARANTEE!

You can cancel your subscription at any time and receive a full refund on all unmailed issues, no questions asked

## SPREAD THE COST OF SUBSCRIBING

For under £15.00\* every 3 months you can pay by convenient Direct Debit. Simply complete the form overleaf and return it back to us today.

**Call now on 0870 444 8645 to make sure your receive the next issue**

\* Prices and savings based on DVD edition.

[www.linuxformat.co.uk](http://www.linuxformat.co.uk)

LXF40 MAY 2003 **97**





# Essential disc info

Read this important information before you use your *Linux Format* coverdisc – CD or DVD. We've collated some helpful info to help you get the most from these jewels of data!

## Finding the essentials

### Missing something?

As many of the programs on our discs are the very latest releases, they are often built on the very latest libraries and may depend on other packages your current Linux setup does not contain. We try to provide you with as many of these important supporting files and libraries as possible, though obviously we don't have space to include absolutely everything.

In many cases, the latest libraries and

other packages you might need will be included in the "essentials" folder on the disc, so if you are missing dependencies, this is the first place to look.

### Package formats

Wherever possible, we try to include as many different types of package for an installation as possible, whether that be distribution specific RPMs, debs or whatever. Please bear in mind that we can only do this where space permits and when the packages are available.

We will, apart from exceptional or legally restricted situations, include the source files for any package, so that you can build it yourself.

### Documentation

These pages provide helpful information on how to install and use some of the packages on the CD. Please note that many of the applications come with their own documentation, and there are additional notes and files in the relevant directories.

## What are all these files?

If you are new to Linux, you may find the profusion of different files and extensions confusing. As we try to give as many packages as possible for compatibility, there will often be two or three files in a directory covering different types of Linux, different architectures and usually source and binary versions – so which do you install? They can be identified by their filenames, and usually just by the file extensions.

**Someap-1.0.1.i386.rpm** – This is probably a binary rpm, designed to run on x86 systems.

**Someap-1.0.1.i386.deb** – The same, but a debian package.

**Someap-1.0.1.tar.gz** – This is usually source code.

**Someap-1.0.1.tgz** – Same as the above, tgz is abbreviated form of tar.gz

**Someap-1.0.1.tar.bz2** – Same, but uses bzip2 compression instead of zip

**Someap-1.0.1.src.rpm** – This is also source code, but supplied as an rpm to make it easier to install

**Someap-1.0.1.i386.RH7.RPM** – A binary, x86 RPM designed specifically for Red Hat Linux

**Someap-1.0.1.ppc.Suse7.rpm** – A binary RPM designed specifically for SuSE7.x PPC Linux.

**Someap-devel-1.0.1.i386.rpm** – A development version.

## Installing from tarballs

A tar ball is a two stage archive. First the files are archived into a single file with tar and then compressed with Gzip or Bzip2. To unpack, cd to the directory you want to unpack it, usually your home directory and type one of the following two lines:

```
tar xzvf /mnt/cdrom/Desktop/progname/progname-2.1.0.tgz
tar xvf -bzip2 /mnt/cdrom/Desktop/progname/progname-2.1.0.tar.bz2
```

Use the first for Gzipped files, those ending in .tar.gz or .tgz, and the second for Bzipped files, ending in .tar.bz2 or .tbz2. Naturally, you change the paths to suit the location and name of the archive. and replace /mnt/cdrom with whatever is applicable to your system (eg /cdrom). This normally unpacks the archive into a directory of the same name, enter that directory with:

```
cd progname-2.1.0
```

To compile and install the software, type the following three commands:

```
./configure
make
su -c "make install"
```

The last line will prompt you for the root password, as this stage must be run as root. If you are already logged in as root, just type **make install**. This will give you a default installation. If you want to change any aspect of the install, type **./configure --help** to see the options available. For example, you are usually able to change the default location with the **PREFIX** argument. When you have finished installing, you may remove the source files with:

```
cd ..
rm -fr progname-2.1.0
```

You should also log out as root, before you do anything you may later regret.

## DEFECTIVE CDs

In the unlikely event of your CD/DVD being physically damaged we'll send you a new, working version within 28 days. Send your defective disc – complete with your name, address, and a description of the fault – to:

**Linux Format, Future Publishing Disc Department, 3B  
Athena Avenue, Elgin Industrial Estate, Swindon, SN2 8HF.**

## Creating install CDs with cdrecord

The quickest way to burn an ISO image to CD is with *cdrecord*. You need to be root to do this. First find the address of your CD-writer with

```
cdrecord -scanbus
```

This will show the devices connected to your system. The SCSI address of each device is the three numbers in the leftmost column, say 0,3,0. Now you can burn a CD with

```
cdrecord dev=0,3,0 -v  
/path/to/image.iso
```

You can simplify the command by saving some default settings in /etc/default/cdrecord. Add a line for each CD writer on your system (usually one) like this

```
Plextor= 0,3,0 12 16M
```

The first item is a label, after the SCSI address you put the speed and the buffer size to use. You can now replace the SCSI address in the command line with the label, but it gets even easier if you add

```
CDR_DEVICE=Plextor
```

Now you can burn an ISO image to disc with

```
cdrecord -v/path/to/image.iso
```

If you really don't want to use the command line, *gcombust* will do the job for you. Start it as root, select the "Burn" tab and the "ISO 9660 Image" gadget near the top of the window. Put the path to the image file in the gadget and press "Combust!". Now put on the kettle while the CD is created for you.

## Other OS?

You do not have to use Linux to burn the ISO to a disc. All the Linux-specific bits are already built into the image file. Programs like *cdrecord* simply dump it to the disk. If you don't have a CD-writer, find someone who does have one, and a DVD drive, and use the CD burning software on their computer. It can be Windows, MacOS, AmigaOS whatever.

## No CD burner?

What if you have no CD writer? Do you know someone else with one? You don't have to use Linux to burn the CDs, any operating system that can run a CD-writer will do the job (see above).

With some distributions it is also possible to mount the images and do a network install, or even a local install from another disk partition. The methods often vary between distributions, so check on the distro vendors website for more information. [LXF](#)

# Coverdisc



**Neil Bothwick** is your guide through the stuffed-to-the-gills *Linux Format* DVD. Get your Linux box DVD-enabled *right now* – next month's disc is double-sided...

**Y**ou're spoilt for choice on the distro front this month – there's eleven of them on the DVD for you to choose from!

Though 19 year-old 'DVD Jon' (Jon Lech Johansen) may be facing a barrage of repeated charges in the Norwegian courts for his part in bringing DVD functionality to the Linux desktop (assisting *DeCSS* to overcome copy-prevention) we bring you *Goggles* and *QuickRipDVD*. It's up to you whether you use them to annoy Hollywood or not – we'd never condone such activity, of course.

On the system side of things, *BaitAndSwitchHoneypot* is a great rerouter for any hostile traffic that you might be experiencing; and

*Makepasswd* and *MondoRescue* can add some extra secondary security to your setup by generating random passwords and helping with disaster recovery respectively.

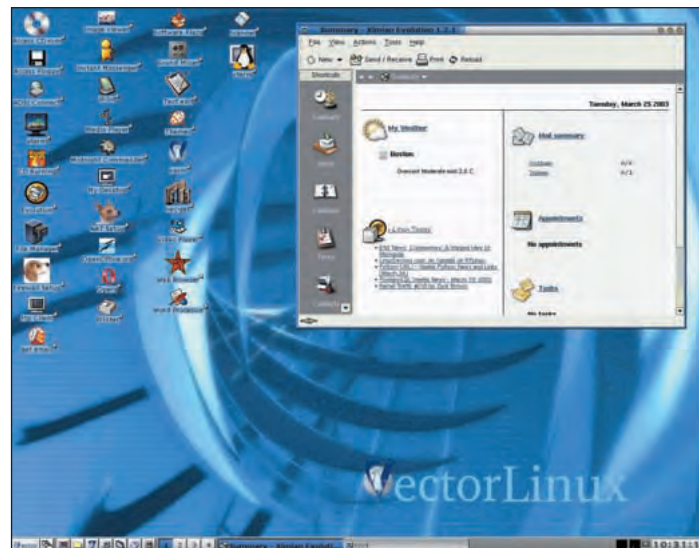
PDA users are also in for a treat this month – there's over a dozen titles for you. Exclusive to the DVD is the infuriatingly addictive *zTappy*. A selection of different tiles appear in a 5x5 matrix, and all you have to do is match them up to win the game. Sounds easy? You'd better carry a spare battery or mains cable if you need to get any work done afterwards! Want to win your own Sharp Zaurus PDA running Linux? Turn to page 92.

## DISTROS VECTORLINUX

There are two versions of Vector Linux on the DVD. The SOHO edition installs from the DVD exactly as it does from the CD, see the CD pages for more information. The standard edition is much smaller, and less demanding. It doesn't have KDE or the office packages, which are what mainly sets the SOHO edition apart. The default desktop is a customised version of *IceWM*, that is more useful than the standard *IceWM* while just as fast. To install this version you need to burn Distros/VectorLinux/VL3.2dwnld.iso to a CD-R (or CD-RW) and boot from it. The installation process is much the same as for the SOHO edition, except it only takes half as long.

## DESKTOP DROPLINEGNOME

Vector Linux SOHO edition comes with the KDE desktop environment, in addition to Vector's standard *IceWM*. Of course, many people prefer GNOME to KDE, and who are we to disappoint them? *Dropline* is an installer for GNOME 2.2.1 on any Slackware-based Linux distribution, which includes Vector. Installation is done in two stages, first you install *Dropline* itself, then you use it



**Vector Linux, running in a fast and light fashion on IceWM.**

to install GNOME. To install *Dropline*, do this as root:

```
cd /mnt/cdrom/Desktop/Dropline
GNOME
```

```
installpkg dropline-installer-2.0.5-
i386-1.dl.tgz
```

You also need to install one or two packages that are needed by *Dropline*, but not installed by Vector. If you are using the standard Vector, installed from the ISO image on the DVD, you need to type both of these lines. If you are using the SOHO version, you only need the first.

```
installpkg dependencies/libpng-1.2.1-
i386-1.tgz
```

```
installpkg dependencies/cdparanoia-
IIIalpha9.8-i386-1.tgz
```

In order to install the packages from the DVD, you need to copy them to where *Dropline* expects to find them, in its cache directory, creating it first if necessary:

```
mkdir -p /var/cache/dropline-installer
cp -p packages/* /var/cache/dropline-
installer
```

Now all you need to do is run the installer with

```
dropline-installer
```

Select the option to install from

hard disk and choose whether you want to install all packages or a selection. Once you've made your choices, *Dropline* will install GNOME for you. All that remains is to configure your display manager to load GNOME.

## INTERNET LOPSTER

Point-to-point file sharing is a legal minefield. However, it exists, and there are legitimate uses for P2P file exchange. Lopster is one of the less well known programs in this field, but a highly regarded one. Until recently, the latest versions have only been available as CVS builds, hardly suitable for inclusion on a cover disc. Now version 1.2.0 has been released, and here it is. Lopster is still supplied as source code, but it only requires the standard `./configure && make && make install` process to install. Lopster provides chat facilities as well as file exchange, and even includes basic IRC support.

If you prefer to use the eDonkey network, we have two programs for you. The eDonkey network is intended for transferring large files, and



Wherever you see this logo it means there's related stuff on the DVD

## IMPORTANT NOTICE

**Before you even put the DVD in your drive, please make sure you read, understand and agree to the following:** The *Linux Format* DVD is thoroughly tested for all known viruses, and is independently certified virus-free before duplication. We recommend that you always run a reliable and up-to-date virus-checker on ANY new software. While every care is taken in the selection, testing and installation of DVD software, Future Publishing can accept no responsibility for disruption and/or loss to your data or your computer system which may occur while using this disc, the programs or the data on it. You are strongly advised to have up-to-date, verified backups of all important files. Please read individual licences for usage terms.



*MLDonkey* is a clone of the standard eDonkey client. It works as a daemon that can be accessed through telnet, a web browser or through the included GTK client. *KMLDonkey*, as the name implies, provides integration for *MLDonkey* into KDE3. This includes a *kicker* applet, *ed2k* protocol handling for *Konqueror* and an applet for the *KDE Control Center* to make everything work the way you want it to.

*KMLDonkey* is supplied as source code, to be installed in the usual way. *MLDonkey* comes with tarballs of pre-compiled binaries. These require no installation, just unpack the appropriate archive and type

```
cd mldonkey-distrib-2.04rc1-0
./mldonkey > mldonkey.log &
./mldonkey_gui &
```

## DESKTOP QTPARTED

One of the main obstacles to installing a new Linux distribution is the need to repartition your hard disk. While some of the distributions provide partitioning tools as part of the installation process, many others do not, or rely on *parted*. *Parted* is a good program in terms of its abilities for creating, resizing and deleting partitions, but the command line interface can be rather daunting for many users, especially as the consequences of giving the wrong instructions to the program could wipe out your data.

*QTParted* uses the *parted* library, but works via a more accessible GUI interface. The functionality is identical to *parted*, they both use the same backend, but it is far easier to see what you are doing, especially for new users. As *QTParted* uses *parted*, you will need to install *parted* first, if your distribution doesn't install it for you,

and a copy of the latest version is in the dependencies directory.

Whether you use *QTParted*, *parted*, a distribution installer or a commercial program like *Partition Magic* to repartition your hard drive, you should back up any important data first. Repartitioning a drive is an inherently risky process. If anything interrupts it, like a system crash or a power failure, you could lose the contents of any partitions you are working on.

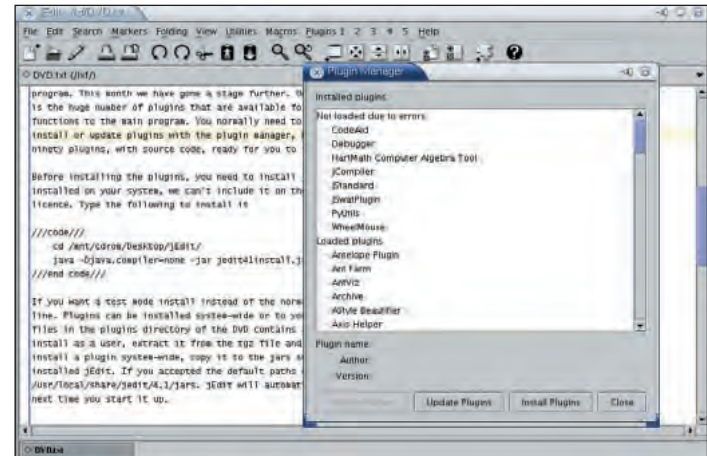
## DESKTOP JEDIT

*jEdit* has been on *Linux Format* cover discs before, because it is a powerful and useful program. This month we have gone a stage further. One of the reasons for *jEdit*'s power is the huge number of plugins that are available for it. These add a diverse range of functions to the main program. You normally need to be connected to the Internet to install or update plugins with the plugin manager, but we have included more than ninety plugins, with source code, ready for you to install.

Before installing the plugins, you need to install *jEdit*. You need to have Java installed on your system, we can't include it on the DVD because of the terms of Sun's licence. Type the following to install *jEdit*

```
cd /mnt/cdrom/Desktop/jEdit/
java -Djava.compiler=none -jar
jedit41install.jar
```

If you want a text mode install instead of the normal GUI, add **text** to the second line. Plugins can be installed system-wide or to your user directory. Each of the *tgz* files in the plugins directory of the DVD contains a *.jar* file, which is the plugin. To install as a user, extract it from the *tgz* file and copy it to *~/jedit/jars*. To install a



*jEdit* is powerful, made even more so by the 90+ plugins on the DVD.

plugin system-wide, copy it to the *jars* sub-directory of wherever you installed *jEdit*. If you accepted the default paths during installation, this would be */usr/local/share/jedit/4.1/jars*. *jEdit* will automatically detect the new plugins the next time you start it up.

## DISTROS ADIOSBOOTCD

Bootable "live" CDs are a popular addition to the *Linux Format* DVDs. They provide an easy way to try out various types of distribution without

the need to repartition a drive or install anything. *Adios Linux* is another such live distribution, based on Red Hat, with a slightly unusual feature. It comes with User Mode Linux, a means of running one instance of Linux inside another. Running *Adios* involves no more than burning the ISO image to a CD and booting from it.

When you get to the login window, the username is **adios** and the password is **12qwazsx**. Yes, the password really is that easy to remember! **LXF**

## Creating your own ISOs

### Essential reading

In the past we have used *jigdo* to help you create ISO images for our bootable DVDs, so that you may install onto computers without DVD drives without losing the convenience of bootable DVDs for other users. There have been some teething problems with this system, it wasn't the most intuitive of processes and it hasn't worked for everyone. We have changed the system for this DVD, it's still based on *jigdo*, but much easier to use, and thoroughly tested on various Linux and Windows setups.

To create an ISO image of Vector SOHO when running Linux, type

```
sh /mnt/cdrom/Distros/VectorLinux/mkiso
```

The ISO image will be created in the current directory and compared with the original's MD5 checksum to make sure it is good. If you want to create the image in another directory, give it as an argument, such as

```
sh /mnt/cdrom/Distros/VectorLinux/mkiso /home/me/iso-images/
```

You can even run the script by clicking on its icon in KDE, and probably other window managers but it was only tested with KDE, but you won't see any messages indicating success or otherwise.

If you get a "Permission denied" error message when trying to run **mkiso**, it means your DVD has been mounted with the **noexec** option. This is the default when set up to allow normal users to mount devices, as a security measure. The quickest way to fix this is to type, as root:

```
mount /mnt/cdrom/ -o exec,remount
```

## Creating the ISO with Windows

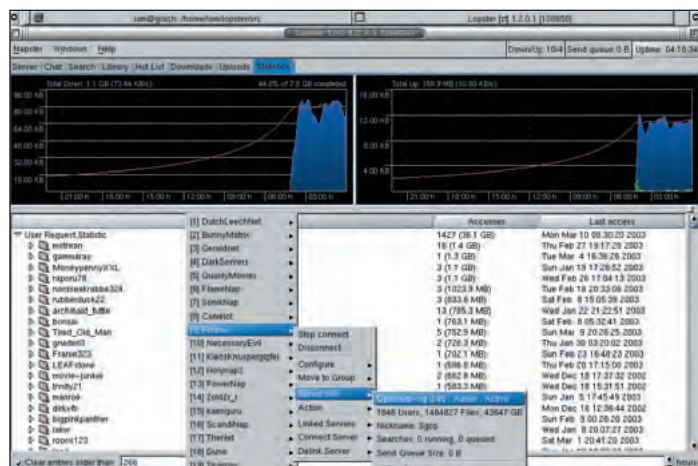
To build the ISO image when running Windows, open an MSDOS shell and type

```
D:\
```

```
cd Distros/VectorLinux
```

```
winmkiso
```

The ISO image will be created in C:. This assumes that your DVD drive is D:, if this is not the case, simply change the first line.



Lopster is one of the best P2P file sharing clients around...

# CoverdiscDVD

## DVD CONTENTS AT A GLANCE

### Desktop

<b>Bochs</b>	Portable x86 PC emulation software package
<b>Bochs-Tools</b>	Collection of tools for the Bochs emulator
<b>CapiSuite</b>	CAPI ISDN application for fax and voice services
<b>COnfiguratorForGnome</b>	Edit advanced GNOME settings
<b>Delimiteaze</b>	A delimited file conversion tool with multiple options
<b>DropLineGNOME</b>	GNOME-based Desktop for Slackware Linux
<b>FCKeditor</b>	The text editor for Internet
<b>GNOME-docs</b>	Useful documentation for GNOME
<b>jEdit</b>	Powerful text editor
<b>KdeCommander</b>	A KDE file browser
<b>KDEKontakt</b>	Integrated personal information management application
<b>KServiceMenu</b>	Editor for Konqueror's servicemenus
<b>MisterHouse</b>	Home automation with Perl
<b>QTParted</b>	Partition Magic clone
<b>Vilistextum</b>	A fast HTML to text converter
<b>Xbrightness</b>	Brightness controller for XFree86

### Development

<b>FileType</b>	File type detection library and application
<b>FingerprintVerification</b>	Biometric library used for fingerprint verification
<b>PatchUtils</b>	Programs for manipulating patch files
<b>Progressbar</b>	Monitor progress of computations with a progress bar
<b>PyInstallShield</b>	Cross-platform installation program
<b>Shorthand</b>	Powerful scripting language
<b>Squeak</b>	Portable Smalltalk 80-based language

### Distros

<b>ADIOSBootCD</b>	Boot CD with UML virtual machines
<b>BBLagentRouter</b>	Single-floppy router for broadband Internet connection
<b>Byzantine</b>	Software Internet Appliance with a home entertainment bias
<b>FdLinux</b>	Small, networkable mini Linux distribution
<b>RPM-livelihood</b>	RPM-based bootable live Linux system
<b>SoL-diag</b>	Diskless Linux distribution for rescue and analysis
<b>VectorLinux</b>	Small, fast, Slackware-based Linux distribution

### Games

<b>ArcCrosswordCompiler</b>	Crossword compiler using arc-consistency algorithms
<b>Childsplay</b>	Suite of educational games for young children
<b>FlightGearSceneryDesigner</b>	Program to design custom sceneries for FlightGear
<b>Kalyp</b>	A roguelike game written in Java
<b>Xname</b>	An arcade game emulator with support for over 2000 games

### Graphics

<b>Goggles</b>	A frontend for the Ogle DVD player
<b>MoviX</b>	Small but powerful console distro for multimedia playback
<b>Ogle</b>	DVD player for Solaris, BSD, and Linux
<b>oKle</b>	A KDE frontend to the Ogle DVD player
<b>QuickRipDVD</b>	Simple DVD ripper that focuses on ease of use
<b>Xine</b>	A Unix video player

### Internet

<b>Balsa</b>	GNOME mail client for local mailboxes, POP3 and IMAP
<b>BKedit</b>	Bookmark editor for a variety of different file formats
<b>Galeon</b>	GNOME Web browser
<b>KMLDonkey</b>	KDE integration for MLDonkey
<b>Lopster</b>	Napster client
<b>MLDonkey</b>	Multi-network file sharing client
<b>OpenBottle</b>	An anti-spam authenticated whitelist email system

### Mobile

<b>Catalist</b>	Shopping/stocktaking application
<b>OpenZaurus</b>	An alternative ROM image for the Sharp Zaurus
<b>ZEthereal</b>	Packet sniffer
<b>zTappy</b>	A simple action game

### Office

<b>OpenLearningManagement</b>	A learning management system
<b>TaskJuggler</b>	Project management tool

### Server

<b>GangplankConferencing</b>	Real-time Internet conferencing system
<b>GraphicalVoterInterface</b>	A voting GUI
<b>H2ORotisserie</b>	Structured discussion system
<b>LinuxBandwidthArbitrator</b>	A turn-key tool to distribute bandwidth on busy networks

**PHPWeather**  
**PureLoadBalancer**  
**SynchronizingKeyServer**  
**Tnftpd**  
**VeniVidiVotiLibrary**

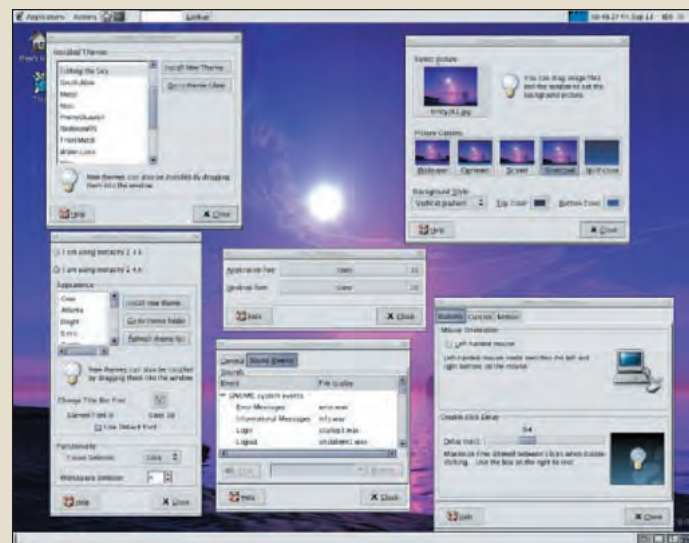
Shows the current weather conditions on your Web page  
 A free high-performance HTTP load balancer  
 OpenPGP Key Server with reliable replication  
 Port of the NetBSD FTP daemon  
 A collaborative writing web site

### Sound

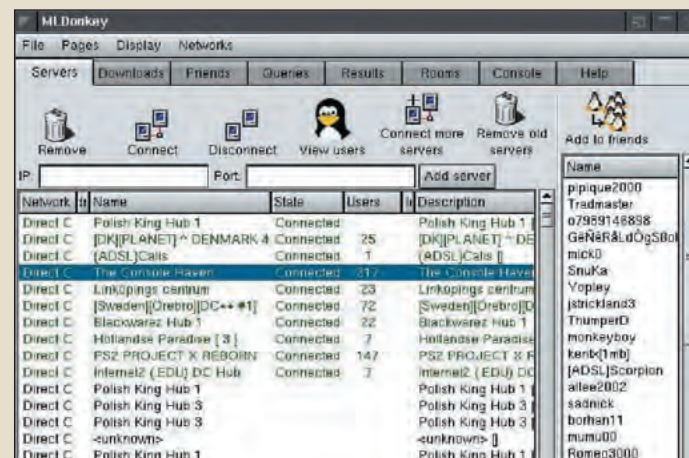
<b>CDLoop</b>	A CD player with special looping capabilities
<b>DJBorg</b>	A text-to-speech MP3 DJ
<b>KdeRadio</b>	Listen to the radio
<b>MakeMP3</b>	A fast CD to MP3 ripper and encoder with CDDb capabilities
<b>Manauton</b>	Records sound to disk in manual or autonomous mode
<b>NoteEditor</b>	Editor for music notation with playback/import/export
<b>RandCD</b>	Make random ISOs of MP3s from a large MP3 collection
<b>ReZound</b>	A graphical audio editor

### System

<b>at76c503Driver</b>	A Linux driver for at76c503-based wlan USB adapters
<b>BaitAndSwitchHoneyPot</b>	Reroutes hostile traffic to a honeypot
<b>FauBackup</b>	Stores incremental and full backups in a file system
<b>Keasyrestore</b>	Backup and restore tools for frequently changing data
<b>Makepasswd</b>	Generates random passwords
<b>MondoRescue</b>	Generates disaster recovery CDs or tapes
<b>PAMdotfile</b>	Allows users to have more than one password per service
<b>WebLog</b>	A Web interface to Syslogd



**GNOME the easy way. DropLine installs GNOME 2.2.1 on any Slackware-based system.**



**... but if you prefer the eDonkey network, there's MLDonkey and KMLDonkey for you to use as well.**



# User Groups

LUGs worldwide are full of members keen to help with your problems, discuss ideas, and generally natter about all things Linux. You can find lots more information online at: [www.lug.org.uk](http://www.lug.org.uk)

## 1 HAMPSHIRE

URL [www.hants.lug.org.uk](http://www.hants.lug.org.uk)  
Contact Hugo Mills

## 2 BRISTOL & BATH

URL [www.bristol.lug.org.uk](http://www.bristol.lug.org.uk)

## 3 SCOTTISH

URL [www.scottish.lug.org.uk](http://www.scottish.lug.org.uk)

## 4 OXFORD

URL [www.oxford.lug.org.uk](http://www.oxford.lug.org.uk)  
Contact Alasdair G Kergon

## 5 KENT

URL [www.kent.lug.org.uk](http://www.kent.lug.org.uk)  
Contact Kevin Groves

## 6 BRIGHTON

URL [www.brighton.lug.org.uk](http://www.brighton.lug.org.uk)  
Contact Johnathan Swan

## 7 WORCESTERSHIRE

URL [www.worcs.lug.org.uk](http://www.worcs.lug.org.uk)

## 8 NORTHANTS

URL [www.northants.lug.org.uk](http://www.northants.lug.org.uk)  
Contact Kevin Taylor

## 9 ANGLIAN

URL [www.anglian.lug.org.uk](http://www.anglian.lug.org.uk)  
Contact Martyn Drake

## 10 MILTON KEYNES

URL [www.mk.lug.org.uk](http://www.mk.lug.org.uk)  
Contact Denny De La Haye

## 11 DONCASTER

URL [www.doncaster.lug.org.uk](http://www.doncaster.lug.org.uk)  
Contact Andy Smith

## 12 MORAY

URL [www.moray.lug.org.uk](http://www.moray.lug.org.uk)  
Contact Stewart Watson

## 13 WEST WALES

URL [www.westwales.lug.org.uk](http://www.westwales.lug.org.uk)  
Contact Dan Field

## 14 WOLVES

URL [www.wolveslug.org.uk](http://www.wolveslug.org.uk)  
Contact Jono Bacon

## 15 PETERBOROUGH

URL [www.peterboro.lug.org.uk](http://www.peterboro.lug.org.uk)  
Contact Steve Gallagher

## 16 EDINBURGH

URL [www.edinburgh.lug.org.uk](http://www.edinburgh.lug.org.uk)  
Contact Alistair Murray

## 17 TYNESIDE

URL [www.tyneside.lug.org.uk](http://www.tyneside.lug.org.uk)  
Contact Brian Ronald

## 18 LEICESTER

URL [www.leicester.lug.org.uk](http://www.leicester.lug.org.uk)  
Contact Clive Jones

## 19 GREATER LONDON

URL <http://glug.linux.co.uk/>  
Contact John Southern

## 20 SURREY

URL [www.surrey.lug.org.uk](http://www.surrey.lug.org.uk)  
Contact Jay Bennie

## 21 CAMBRIDGE

URL [www.cam-lug.org.uk](http://www.cam-lug.org.uk)

## 22 DEVON & CORNWALL

URL [www.dclug.org.uk](http://www.dclug.org.uk)  
Contact Simon Waters

## 23 FALKIRK

URL [www.falkirk.lug.org.uk](http://www.falkirk.lug.org.uk)

## 24 MANCHESTER

URL [www.manlug.mcc.ac.uk](http://www.manlug.mcc.ac.uk)  
Contact John Heaton, Owen Le Blanc

## 25 HERTFORDSHIRE

URL [www.herts.lug.org.uk](http://www.herts.lug.org.uk)  
Contact Nicolas Pike

## 26 WEST YORKSHIRE

URL [www.wylug.lug.org.uk](http://www.wylug.lug.org.uk)  
Contact Jim Jackson

## 27 SHEFFIELD

URL [www.shefflug.co.uk](http://www.shefflug.co.uk)  
Contact Richard Ibbotson

## 28 STAFFORDSHIRE

URL [www.staffslug.org.uk](http://www.staffslug.org.uk)

## 29 NORTH EAST

URL [www.shofar.uklinux.net/NELUG](http://www.shofar.uklinux.net/NELUG)

## 30 LONDON

URL [www.lonix.org.uk](http://www.lonix.org.uk)

## 31 BERKSHIRE & THAMES VALLEY

URL [www.sclug.org.uk](http://www.sclug.org.uk)

## 32 LIVERPOOL OPENSOURCE

URL [http://linux.liv.ac.uk/\\_liv\\_linux\\_ug/](http://linux.liv.ac.uk/_liv_linux_ug/)  
Contact Simon Hood

## 33 DEAL AMIGA CLUB

Email [superhighwayman@hotmail.com](mailto:superhighwayman@hotmail.com)  
Contact John Worthington

## 34 CHESTERFIELD

Email [spirelug@yahoo.co.uk](mailto:spirelug@yahoo.co.uk)  
Contact Robin Needham

## 35 SOUTH DERBYSHIRE

URL [www.sderbylug.org.uk](http://www.sderbylug.org.uk)  
Contact Dominic Knight

## 36 BELFAST (BLUG)

URL [www.belfastlinux.cx](http://www.belfastlinux.cx)  
Email [russell@belfastlinux.org](mailto:russell@belfastlinux.org)

## 37 WILTSHIRE

URL [www.wiltshire.lug.org.uk](http://www.wiltshire.lug.org.uk)  
Contact Jason Rudgard

## 38 SOUTH LONDON

URL [www.sl.lug.org.uk](http://www.sl.lug.org.uk)  
Email [edo@perceptiondm.com](mailto:edo@perceptiondm.com)

## 39 CHESHIRE

URL [www.sc.lug.org.uk](http://www.sc.lug.org.uk)  
Contact Anthony Prime – [enquiry@sc.lug.org.uk](mailto:enquiry@sc.lug.org.uk)

## 40 NORTH WALES

URL [www.northwales.lug.org.uk](http://www.northwales.lug.org.uk)  
Contact Jonathan Cole

## 41 MIDLANDS

URL <http://midlandslug.port5.com/>  
Contact Pete Thompson

## 42 CUMBRIA

URL [www.cumbria.lug.org.uk](http://www.cumbria.lug.org.uk)  
Contact Jamie Dainton

## 43 DORSET

URL [www.dorset.lug.org.uk](http://www.dorset.lug.org.uk)  
Contact John Robinson

## 44 SHROPSHIRE

URL [www.shropshire.lug.org.uk](http://www.shropshire.lug.org.uk)  
Email [shropshire@lug.org.uk](mailto:shropshire@lug.org.uk)

## 45 SOUTH WEST

URL [www.southwest.lug.org.uk](http://www.southwest.lug.org.uk)  
Email [southwest@lug.org.uk](mailto:southwest@lug.org.uk)

## 46 SOUTH WALES

URL [www.swlug.org.uk](http://www.swlug.org.uk)

## 47 NORTH LONDON

URL [www.kemputing.net/lug/anlug-aims.html](http://www.kemputing.net/lug/anlug-aims.html)  
Email [jason@voyagercomputers.co.uk](mailto:jason@voyagercomputers.co.uk)

## 48 MALVERN

URL [www.malvern.lug.org.uk](http://www.malvern.lug.org.uk)  
Contact Greg Wright

## 49 HUDDERSFIELD

URL [www.hud.lug.org.uk](http://www.hud.lug.org.uk)  
Contact Dave Naylor – [knocker@caramboo.com](mailto:knocker@caramboo.com)

## 50 NOTTINGHAM

URL [www.nottingham.lug.org.uk](http://www.nottingham.lug.org.uk)

## 51 ST ALBANS & LUTON

URL [www.lust.lug.org.uk](http://www.lust.lug.org.uk)  
Contact Michael Culverhouse – [mike@easily.co.uk](mailto:mike@easily.co.uk)

## 52 WREXHAM

Contact Paul Kersey-Smith  
Email [paul@pkls.fsnet.co.uk](mailto:paul@pkls.fsnet.co.uk)

## 53 PRESTON & LANCs

URL [www.preston.lug.org.uk](http://www.preston.lug.org.uk)  
Contact Phil Robinson

## 54 DERRY

URL [www.derry.lug.org.uk](http://www.derry.lug.org.uk)

## 55 ISLE OF WIGHT

URL [www.iow.lug.org.uk](http://www.iow.lug.org.uk)  
Contact David Groom – [info@iow.lug.org.uk](mailto:info@iow.lug.org.uk)

## 56 SCARBOROUGH

URL [www.scarborough.lug.org.uk](http://www.scarborough.lug.org.uk)

## 57 BLACKBURN

Email [matt@consultmatt.co.uk](mailto:matt@consultmatt.co.uk)

## 58 YORK

URL [www.york.lug.org.uk](http://www.york.lug.org.uk)

## 59 Lincs

URL [www.lincs.lug.org.uk](http://www.lincs.lug.org.uk)

**60 HULL**URL [www.hull.lug.org.uk](http://www.hull.lug.org.uk)**61 WALTON-ON-THAMES**Contact William Mutch  
Email [rael@freeuk.com](mailto:rael@freeuk.com)**62 GLOUCS & COTSWOLDS**URL [www.gloucs.lug.org.uk](http://www.gloucs.lug.org.uk)**63 WEST OF SCOTLAND**URL [www.wos.lug.org.uk](http://www.wos.lug.org.uk)**64 SOUTH STAFFORDSHIRE**URL [www.staffs.lug.org.uk](http://www.staffs.lug.org.uk)**65 MANSFIELD**URL [www.mansfield.lug.org.uk](http://www.mansfield.lug.org.uk)**66 BORDERS**URL [www.linux.bordernet.co.uk](http://www.linux.bordernet.co.uk)**67 BIRMINGHAM**URL [www.sb.lug.org.uk](http://www.sb.lug.org.uk)**68 COVENTRY**Email [info@coventry.lug.org.uk](mailto:info@coventry.lug.org.uk)**69 NEWARK & LINCOLN**URL [www.newlinc.lug.org.uk](http://www.newlinc.lug.org.uk)**70 BEDFORDSHIRE**URL [www.beds.lug.org.uk](http://www.beds.lug.org.uk)**71 LINCOLN**URL [www.lincoln.lug.org.uk](http://www.lincoln.lug.org.uk)**72 LOUGHBOROUGH**URL [www.loughborough.lug.org.uk](http://www.loughborough.lug.org.uk)**73 EXETER UNIVERSITY**Email [N.J.Murison@exeter.ac.uk](mailto:N.J.Murison@exeter.ac.uk)**74 SUNDERLAND**Email [thomas.croucher@sunderland.ac.uk](mailto:thomas.croucher@sunderland.ac.uk)**75 EAST YORKSHIRE**Email [sharkonline@whatemail.com](mailto:sharkonline@whatemail.com)**76 CLEVELAND OPEN SOURCE GROUP**Email [openlug@digitalmedia.co.uk](mailto:openlug@digitalmedia.co.uk)**77 BEVERLEY**Email [vladimir\\_lukyanov@hotmail.com](mailto:vladimir_lukyanov@hotmail.com)**78 DUNDEE & TAYSIDE**URL [www.dundee.lug.org.uk](http://www.dundee.lug.org.uk)**79 SUSSEX**URL <http://sussex.lug.org.uk/>**80 WIGAN & ST HELENS**Email [paulf.johnson@ukonline.co.uk](mailto:paulf.johnson@ukonline.co.uk)**81 BRIXTON**URL [www.communitytechnology.org.uk/~linuxhome](http://www.communitytechnology.org.uk/~linuxhome)**82 ST.ANDREWS, FIFE**URL [www.standrews.lug.org.uk](http://www.standrews.lug.org.uk)Email [stuart@nx14.com](mailto:stuart@nx14.com)**83 NUNEATON**URL [www.nuneaton.lug.org.uk](http://www.nuneaton.lug.org.uk)**84 ISLE OF MAN**URL [www.iom.lug.org.uk](http://www.iom.lug.org.uk)Email [helix@manx.net](mailto:helix@manx.net)**85 AYLESBURY**URL [www.aylesbury.lug.org.uk](http://www.aylesbury.lug.org.uk)Email [drbond@educational-computing.co.uk](mailto:drbond@educational-computing.co.uk)**86 LANCASHIRE**URL [www.lancasterlug.org.uk](http://www.lancasterlug.org.uk)**87 EAST LONDON**URL [www.eastlondon.lug.org.uk](http://www.eastlondon.lug.org.uk)

Contact Jonathan Spriggs

**88 ORMSKIRK**Email [rob@northwestlinux.co.uk](mailto:rob@northwestlinux.co.uk)**89 HEREFORD**URL [www.hereford.lug.org.uk/](http://www.hereford.lug.org.uk/)Email [rbjh@good-news.fsnet.co.uk](mailto:rbjh@good-news.fsnet.co.uk)**90 EAST HERTS**Email [madtom1999@yahoo.com](mailto:madtom1999@yahoo.com)**91 SWINDON**Email [nick.trueman@ntword.com](mailto:nick.trueman@ntword.com)**92 MENAI**URL [www.menai.lug.org.uk](http://www.menai.lug.org.uk)**93 ABERDEEN**Email [mark@arricc.net](mailto:mark@arricc.net)**94 SHETLAND**URL [www.shetland.lug.org.uk](http://www.shetland.lug.org.uk)Email [c\\_s\\_s\\_butler@yahoo.com](mailto:c_s_s_butler@yahoo.com)**OTHER GROUPS:****B-LUX**URL [www.b-lux.org/](http://www.b-lux.org/)Email [blux-admin@lineone.net](mailto:blux-admin@lineone.net)**FUENGIROLA GNU/LUG**URL [www.fuengirola.lug.org.uk/](http://www.fuengirola.lug.org.uk/)Email [fuengirola@lug.org.uk](mailto:fuengirola@lug.org.uk)**SCHOOLS**URL [www.schools.lug.org.uk/](http://www.schools.lug.org.uk/)**TR LINUX – TURKISH LINUX**URL [www.trlinux.lug.org.uk/](http://www.trlinux.lug.org.uk/)Email [c\\_s\\_s\\_butler@yahoo.com](mailto:c_s_s_butler@yahoo.com)**YOUNG LINUX**URL [www.young.lug.org.uk/](http://www.young.lug.org.uk/)



# LinuxUserGroups

## LUG OF THE MONTH

### Myanmar

#### Matt Nailon writes:

Myanmar is a country that most people would probably be hard-pushed to point to in an atlas, but it boasts a Linux-powered and Open Source-savvy web presence that shames that of many supposedly more technological nations.

[www.mmlinux.org](http://www.mmlinux.org) and [www.schwenet.com](http://www.schwenet.com) take browsers to the same online resource – a Linux-enabled news and links front page with a search engine specific to

Myanmar (formerly Burma). Apparently, the name of the site comes from the Myanmar word for Gold ('Shwe'), which is often used to depict the people of Myanmar and the country itself, due to the abundance of golden pagodas and monuments throughout the country.

SchweNET has branch offices in both Yangon, capital of Myanmar and Auckland, New Zealand. SchweNET is dedicated to IT-industry, cultural, educational and tourism development

in Myanmar; all three sites here are inappropriate places for discussion of politics or government.

Myanmar also has a LUG at [www.myanmarlug.org](http://www.myanmarlug.org). Set up initially in 2001, members have assisted in many projects, including producing fonts of characters for the Myanmar language. Both Myanmar LUG and SchweNET are very keen to hear from Linux users and other LUGs from all over the world.



## Worldwide Linux User Groups

Free Software users across the globe

### Africa

#### EGYPT

URL [www.linux-egypt.org](http://www.linux-egypt.org)

Contact Hesham Bahram

#### GAUTENG, SOUTH AFRICA

URL [www.glug.org.za](http://www.glug.org.za)

Email [glugmin@revolution.org.za](mailto:glugmin@revolution.org.za)

### Australia

#### ADELAIDE

URL [www.linuxsa.org.au](http://www.linuxsa.org.au)

Email [mtippet@anu.edu.au](mailto:mtippet@anu.edu.au)

#### ALICE SPRINGS

URL [www.aslug.org.au](http://www.aslug.org.au)

#### MELBOURNE, VICTORIA

URL [www.luvasn.au](http://www.luvasn.au)

Contact [luc-committee@luvasn.au](mailto:luc-committee@luvasn.au)

#### PERTH

URL <http://plug.linux.org.au/>

#### SYDNEY

URL [www.slug.org.au](http://www.slug.org.au)

### Europe

#### AUVERGNE

URL [www.linux-arverne.org](http://www.linux-arverne.org)

Email [Cyril.Hansen@wanadoo.fr](mailto:Cyril.Hansen@wanadoo.fr)

#### COSTA DEL SOL (English speaking)

URL [www.fuengirola.lug.org.uk](http://www.fuengirola.lug.org.uk)

#### DENMARK

Alssund [www.alslug.dk](http://www.alslug.dk)

Esbjerg [www.eslug.dk](http://www.eslug.dk)

Fyns [www.flug.dk](http://www.flug.dk)

Midt-og Vestjylland [www.mvjlug.dk](http://www.mvjlug.dk)

Nordjylland [www.njlug.dk](http://www.njlug.dk)

Skåne Sjælland [www.sslug.dk](http://www.sslug.dk)

Trekantsområdet [www.tlug.dk](http://www.tlug.dk)

Vest-fyn [www.haarby-net.dk/vflug](http://www.haarby-net.dk/vflug)

Århus [www.aalug.dk](http://www.aalug.dk)

#### EIRE

URL [www.linux.ie](http://www.linux.ie)

Email [root@linux.ie](mailto:root@linux.ie)

URL [www.dilu.org](http://www.dilu.org)

Email [glossary@dilu.org](mailto:glossary@dilu.org)

#### GOTHENBURG

<http://nain.oso.chalmers.se/LUG/>

### Middle East

#### ISRAEL

URL [www.iglu.org.il/IGLU/](http://www.iglu.org.il/IGLU/)

Email [webmaster@iglu.org.il](mailto:webmaster@iglu.org.il)

#### PALESTINE

URL [www.lugps.org](http://www.lugps.org)

Email [isam@planet.edu](mailto:isam@planet.edu)

### Asia

#### HONG KONG (multilingual)

URL [www.linux.org.hk](http://www.linux.org.hk)

#### SINGAPORE – SLUG

URL [www.lugs.org.sg](http://www.lugs.org.sg)

#### SRI LANKA

URL [www.lklug.pdn.ac.lk](http://www.lklug.pdn.ac.lk)

#### MYANMAR (formerly BURMA)

URL [www.myanmarlug.org](http://www.myanmarlug.org)

Email [aftyde@balug.org](mailto:aftyde@balug.org)

#### PAKISTAN

URL [www.linuxpakistan.net](http://www.linuxpakistan.net)

Email [tux@clug.org](mailto:tux@clug.org)

#### HYDERABAD, SINDH, INDUS VALLEY

URL [www.geocities.com/slug\\_pk/](http://www.geocities.com/slug_pk/)

### China

#### BEIJING (GB encoding, but mostly written in Chinese)

URL <http://mud.263.net.cn/~linux>

#### CHINESE LINUX USER GROUP

URL [www.linux.org.cn](http://www.linux.org.cn)

#### NANJING

URL <http://jllib.jionline.com/njlug>

### India

#### LINUX INDIA

URL <http://linux-india.org>

#### ALIGARH LUG

URL <http://linux.amupost.com>

#### BOMBAY

URL [www.ilug-bom.org.in](http://www.ilug-bom.org.in)

#### CHANDIGARH

URL [www.geocities.com/vipinb](http://www.geocities.com/vipinb)

#### CHENNAI AND MADRAS

URL [www.chennaiug.org/](http://www.chennaiug.org/)

#### CYBERABAD (CLUG)

URL <http://seeknew.freesevers.com/clug/>

#### DELHI

URL [www.linux-delhi.org](http://www.linux-delhi.org)

#### KOLKATA

URL [www.ilug-cal.org](http://www.ilug-cal.org)

#### MADURI

URL <http://linuxmadurai.tripod.com>

#### Northern India Linux

URL <http://groups.yahoo.com/group/lug-northindia>

## Spreading the word

Presenting TCO figures and concerns to non-techies is *the* crucial part of advocating Linux adoption, says **Jono Bacon**

Recently we have been discussing the Total Cost Of Ownership (TCO), and though we have discussed how to start calculating TCO, it is important to present the TCO in a structured and positive format.

It is important to remember that TCO is not a straight-forward single theory subject; many people use different TCO calculations to get their data. It is important when showing the TCO that you divulge how you got the figures and what your process is. Remember that if you are honest and detailed in your TCO assessment, you will get more punters taking an interest in your assessment and Linux.

Linux is an OS that scores on a number of areas when it comes to TCO but it does not necessarily mean it is cheap to 'own'. The ownership of an OS and solution is highly dependent on the context, and it is important that you don't make a punter feel that the TCO of Linux compared to another is a one-size-fits-all situation; it certainly is not, and

although some concepts match different contextual situations for a Linux-based solution, these should only be partially mentioned.

The actual presentation of your TCO assessment should be concise and direct and not be too laborious in its explanation – just present the information in a factual and direct way. The reader may not have time to sit and read a 250-page report on TCO and hence a 6-page report may be a little more approachable and not filed in the circular trash cabinet. Presentation is indeed the key, and if the reader feels that it is directed to their current situation and context it will be read more eagerly.

Next month we will look at how to actually get your advocacy material into the hands of people, how to push it and how to manage feedback and queries. Remember to use the LXF forums at [www.linuxformat.co.uk](http://www.linuxformat.co.uk) to let me know what you think of this series and any issues that you would like to be covered!

## Linux User Group organisers

If you're not listed here, or we have your details wrong, please contact us at: **LUGS!, Linux Format, 30 Monmouth Street, Bath, BA1 2BW** or email your details to: [linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

**EDITORIAL****Editor** Nick Veitch [nick.veitch@futurenet.co.uk](mailto:nick.veitch@futurenet.co.uk)**Reviews Editor** Paul Hudson [paul.hudson@futurenet.co.uk](mailto:paul.hudson@futurenet.co.uk)**Production Editor** Matt Nailon [matthew.nailon@futurenet.co.uk](mailto:matthew.nailon@futurenet.co.uk)**Art Editor** Julian Jefferson [julian.jefferson@futurenet.co.uk](mailto:julian.jefferson@futurenet.co.uk)

**Editorial Contributors** Jono Bacon, Neil Bothwick, David Cartwright, Andy Channell, Amias Channer, David Coulson, Rie Dodsworth, Richard Drummond, Hoyt Duff, Michael J Hammel, David Harris, Maurice R Kelly, Ian Kilpatrick, Peter Mackley, Patrick K O'Brien, Mike Saunders, Paul Sellars

**COVER CD PRODUCTION****CD Editor** Neil Bothwick**ART CONTRIBUTORS****Art assistance** Martin Mulchinock**Photography** Amanda Thomas, James Wilson, Corbis, Photodisc, Eyewire, Powerstock, Getty**Illustration** Paul Bateman, Chris Winn, Shane Collinge, Fudge Studios**ADVERTISING SALES****Key account manager** George Gill [george.gill@futurenet.co.uk](mailto:george.gill@futurenet.co.uk)**Business development manager** Damian Hughes**Senior Sales Executives** Giles Crosthwaite-Scott, Harry Maltby**Classified Sales Executive** Stephen Hall**MARKETING AND PROMOTIONS****Marketing Manager** Mike Hawkins**Subscriptions Executive** Marie Spicer**PRODUCTION****Production Co-ordinator** Diane Ross**Group Production Manager** Clare Tovey**Advertising Production Co-ordinator** Jo Crosby**MANAGEMENT****Publisher** Kelley Corten**Group Publisher** Dave Taylor**Publishing Director** John Weir**Managing Director** Colin Morrison**DISTRIBUTION AND CIRCULATION****Circulation Manager** Jamie Malley[jamie.malley@futurenet.co.uk](mailto:jamie.malley@futurenet.co.uk)

**Distributed by** Seymour Distribution, 86 Newman Street, London W1T 3EX **Tel** +44 (0)207 3968000

**Overseas Distribution** by Future Publishing Ltd. **Tel** 01225 442244.

**Overseas Licences****International Licensing Director** Simon Wear[simon.wear@futurenet.co.uk](mailto:simon.wear@futurenet.co.uk) **Tel** +44(0)1225 822798**Fax** +44 (0) 1225 788105 **Cell phone** +44(0)777 5641493**Contact Details**

Linux Format, 30 Monmouth Street, Bath BA1 2BW

**Tel** +44 (0)1225 442244 **Email** [linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)**Subscriptions and Mail Order****Phone** +44 (0)1458 271178. See page 96**Email** [linuxformat.subs@futurenet.co.uk](mailto:linuxformat.subs@futurenet.co.uk)

**Copyright** No part of this publication may be reproduced without written permission from our publisher. We assume all letters sent – by email, fax or post – are for publication unless otherwise stated, and reserve the right to edit contributions. All contributions to Linux Format are submitted and accepted on the basis of non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance in writing. Linux Format recognises all copyrights in this issue. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight. We cannot be held responsible for mistakes or misprints.

**All CD-ROM** demos and reader submissions are supplied to us on the assumption they can be incorporated into a future covermounted CD-ROM, unless expressly stated to the contrary.

**Disclaimer** All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

**Printed** in the UK by Midway Clark (Holt) and TPL

© Future Publishing Ltd 2003

**LINUX** is a trademark of Linus Torvalds, GNU/Linux is abbreviated to Linux throughout for brevity. All other trademarks are the property of their respective owners

**Future Publishing Ltd. is part of The Future Network plc.**

The Future Network produces carefully targeted specialist magazines and websites for groups of people who share a passion. We aim to satisfy their passion by creating titles that offer superb value for money, trustworthy information, multiple ways to save time and money, and are a pleasure to read or visit. Today we publish more than 80 magazines and over 30 magazine websites and networks from offices in five countries. The company also licenses 32 of its titles resulting in over 60 local editions in a further 23 countries.

The Future Network plc is a public company quoted on the London Stock Exchange (symbol: FNET).

**Non-executive Chairman** Roger Parry**Chief Executive** Greg Ingham**Chief Operating Officer & Managing Director, UK** Colin Morrison**Group Finance Director** John Bowman**Tel** +44 (0)1225 442244

BATH • LONDON • MILAN •  
NEW YORK • PARIS •  
SAN FRANCISCO • WROCLAW  
[www.thefuturenetwork.plc.uk](http://www.thefuturenetwork.plc.uk)



Media with passion

# NEXT MONTH

## Issue 41 on sale Thursday 22 May 2003

# HAMMER TIME!

As AMD launch their first 64-bit processors, we look at the server-oriented Opteron and whether the loose collaboration with the Linux community has paid off.

## LAPTOPTASTIC!

Linux on the move is easier than ever. We road test a range of off-the-shelf laptops and deliver our findings on their suitability as mobile Linux machines.

## ON TEST:

Find out whether Mandrake 9.1 or Red Hat 9 is the best desktop distro, plus AC3D modelling software, NEC servers and more reviewed and rated.

## PLUS:

Steel yourself for another mega **DOUBLE-SIDED** DVD! If you are a CD subscriber, now 's the time to upgrade your system in time to get hold of this amazing disc, containing **full versions** of Mandrake 9.1 and Slackware (both installable direct from DVD)!

## and LINUX PRO

24 pages of real-world Linux news, views and case studies for IT professionals

### DON'T MISS YOUR COPY OF LINUX FORMAT!

Get it delivered to your door every month – subscribe on page 96



The exact contents of future issues are subject to change



# Welcome

Twenty-four pages of real-world Linux for IT professionals

As you probably saw, the main cover story in *Linux Format* is SCO's legal action against IBM. This story and the merits of the case are discussed in detail there, so we won't be going into the details in *Linux Pro*. But what we did think would be timely and useful is a look at the liabilities faced by developers working with Free and Open Source software. The legal position on many of the subjects isn't clarified, as the GPL has never been fully tested in court. However, we can take a look at the type of action which could be taken, and steps you could take to reasonably make sure you are staying legal. It's important to point out that this isn't just for part-time coders, but equally has implications for companies using or developing software (even to use in-house) too.

We also like to get a fair amount of entirely practical pieces into *Linux Pro*. This time, our storage section is an overview of *LVM*, and how it can be used to save time, effort and money on systems big and small. There is a common misconception that *LVM* isn't much use if you already have RAID solutions – this simply isn't so. Find out why on page 13.

It's always nice to cover stories that show how Linux is making unique technology solutions viable. In this spirit, we have taken a look at West Yorkshire Police's VIPER system, which is rapidly changing the way identity parades are handled in this country, and saving taxpayers millions in the process.

There's also a look at the newly formed Desktop Linux consortium, aimed at promoting acceptance of Linux as a desktop OS, and two interesting comment pieces in our regular security spot this issue.

As ever, thanks for the feedback we get on *Linux Pro*. If there are topics you would like to see us cover in future, or have comments and suggestions on articles we have already run, I'd love to hear them – email me at the address below...

**Nick Veitch** Editor  
nick.veitch@futurenet.co.uk



**“Legal liability isn't a topic that only bedroom hackers need to know about, it should concern all businesses developing or deploying code too.”**

## Contacts

### EDITORIAL

Editor  
**Nick Veitch** [nick.veitch@futurenet.co.uk](mailto:nick.veitch@futurenet.co.uk)  
Reviews editor  
**Paul Hudson** [paul.hudson@futurenet.co.uk](mailto:paul.hudson@futurenet.co.uk)  
Art editor  
**Julian Jefferson** [julian.jefferson@futurenet.co.uk](mailto:julian.jefferson@futurenet.co.uk)  
Production editor  
**Matt Nailon** [matthewnailon@futurenet.co.uk](mailto:matthewnailon@futurenet.co.uk)  
Group publisher  
**Dave Taylor**  
Publisher  
**Kelley Corten**  
Contributors  
**Barbara Gengler, David Harris, Peter Mackley, Rie Dodsworth**

Photography  
**Photodisc, Powerstock, Corbis**

### ADVERTISING

Key Account Manager  
**George Gill** [george.gill@futurenet.co.uk](mailto:george.gill@futurenet.co.uk)  
Business Development Manager  
**Damian Hughes**  
[damian.hughes@futurenet.co.uk](mailto:damian.hughes@futurenet.co.uk)  
Senior Sales Executives  
**Giles Crosthwaite-Scott, Harry Maltby**  
Classified Sales Executive  
**Stephen Hall**

LinuxPro is a part of



magazine

Published by



Media with passion

30 Monmouth Street, Bath BA1 2BW  
Tel +44(0) 1225 442244

## CONTENTS

### DESKTOP LINUX

The Desktop Linux Consortium **p4**

### LIABILITY FOR CODERS

Will code land you in court? **p6**

### STORAGE: LVM

Save time and money **p13**

### LINE UP FOR LINUX

Linux solves street crime **p16**

### CROSS PLATFORM GUI

Pros and cons of GUI toolkits **p18**

### SECURITY

Two stories on security policy **p22**



## DESKTOP LINUX



# Scramble for the Desktop

**A** vendor-neutral association formed by a raft of leading vendors and Open Source organisations has launched a consortium to bring Linux on the desktop to the wider computing public.

The Desktop Linux Consortium, (DLC), which also has the backing of Linux founder and chief developer of the Linux kernel Linus Torvalds, is developing the technology they claim will speed the adoption rate of Linux to the desktop of computers everywhere.

"We already have all of the tools, in Open Source software, necessary for 80 per cent of office workers in the world, an office suite including spreadsheet, word processor and presentation program; a web browser, graphical desktop with file manager and tools for communications, scheduling and personal information management," Torvalds said in a statement. "The Linux desktop is inevitable!"

Still in the planning stages, the committee has set up a formation committee to jump-start the venture. A 90-day timeline is now in place with the goal of bringing in OEMs and major technology companies to expand its reach. Jeremy White, interim chairman, said the initial intentions are very clear. "Linux is firmly established in the server space and now desktop Linux is coming of age," he said.

## **BARBARA GENGLER of Edittech introduces the Desktop Linux Consortium.**

"The ultimate beneficiary of the consortium is the computing public, which will be assured a open, stable alternative to closed proprietary systems."

Organising DLC members include SuSE, MandrakeSoft, Lycoris and a host of Linux application companies that include CodeWeavers, Ximian and NeTraverse as well as Open Source organisations such as Debian, Samba and OpenOffice.org.

## **Open Standards**

The DLC founding members said open standards, lower licensing fees, proven and reliable technologies, along with the worldwide developer base, underscore GNU/Linux as the ideal software platform for end user computing. The DLC hopes to boost adoption of Linux on the desktop through initiatives that will target the needs of corporate, institutional and homes users, such as trade shows, conferences and participation in consortium-sponsored public relations activities and programs.

But noticeably absent from the group is Lindows.com, the company that prevailed in its fight with Microsoft to use the Lindows operating system brand name. Some participants confirmed the new consortium is, in part, a



reaction to the behaviour of Lindows, which had tried to control the agenda of the Desktop Linux Summit to be held later this month.

Before announcing the formation of the DLC, several of the desktop Linux companies withdrew from the Lindows-organised conference. They accused its chief executive Michael Robertson of using the forum too much to promote his company's wares when he stepped in to keynote the Desktop Linux Summit, after the DLC had set up an allegedly vendor-neutral advisory board.

"Based on this blatant disregard for the Advisory Board, which we felt was crucial to the vendor-neutrality of the conference, and the unilateral substitution of a completely new agenda, which is, of course, the heart of the conference," the DLC said in a statement, "DesktopLinux.com no longer can lend its good name and dynamic community to supporting the Desktop Linux Summit event."

The DLC also said Lindows.com is within their rights to host a conference on any subject whatsoever, but DesktopLinux.com is committed to the principle of vendor neutrality in its editorial content and initiatives.

"Our continued support of the conference as its major media sponsor would constitute a violation of the trust that the community places in our objectivity," the DLC said. Long-time Linux developer Bruce Perens, who withdrew from the conference and who is leading the DLC as interim executive director, said the consortium will assure that there is fairness in all Desktop-Linux related issues and events.

"All vendors will be fully represented and the Open Source ethos will be respected," Perens said.

## Uphill struggle

It would appear the consortium would still have an uphill struggle to make any impact on the desktop sector as market research firm IDC reported, while it has seen some increase in shipment of Linux desktop licenses, Linux remains a very small player on the desktop. "Most organisations do not focus on the operating system," said IDC analyst Dan Kusnetzky. "For the most part, their focus is on the applications and/or development tools that they need to be productive."

He maintained that if a given application is only available on Windows, "they'll choose Windows." According to Kusnetzky, the desktop consortium would have to persuade nearly every Windows developer to make his/her product available on Linux. "It is rather unlikely that the group will be able to find all of the Windows developers to chat with them," he said. And even if they could find all of them, Kusnetzky said these folks would point out that Windows holds about 94 per cent share of the client operating environments shipments worldwide (2001 data) while Linux held only about 1.7 per cent share in the same timeframe.

"This means that it would be very difficult for them to recover their engineering costs for the effort to port the Windows software over to Linux as a native product," he said. "It certainly would be easier if they would port over to the *Wine* environment," (an operating system

technology designed to imitate the Windows environment on a Linux platform.)

## Challenges

Microsoft, which did not want to comment directly about the formation of the consortium, said in its latest US Securities and Exchange Commission filing, "the open source movement poses a significant challenge to the company's business model."

"Since its inception, the company's business model has been based upon customers agreeing to pay a fee to license software developed and distributed by Microsoft," the filing states. "Under this commercial software development (CSD) model, software developers bear the costs of converting original ideas into software products through investments in research and development, offsetting these costs with the revenues received from the distribution of their products."

On the other hand, under the Open Source model, global communities of programmers produce software and that resulting software and intellectual property is licensed to end users at little or no cost. Microsoft also disclosed in its filing that to the extent the Open Source model gains increasing market acceptance, "sales of the company's products may decline, the company may have to reduce the prices it charges for its products and revenues and operating margins may consequently decline."

In spite of the challenges, the DCL will push itself towards the advancement and promotion of Linux, and expects by mid to late summer to put together some sort of vendor-neutral trade show. The non-profit trade association, funded by annual dues and donations, will be open to companies and to Open Source organisations throughout the world and additional members are expected to be announced soon, the group said. ■



**"Linux is firmly established in the server space and now desktop Linux is coming of age... and the ultimate beneficiary is the computing public."**

**JEREMY WHITE,  
INTERIM CHAIRMAN,  
DESKTOP LINUX  
CONSORTIUM**

## How to avoid being sued for

# \$1 Billion!

**B**y hacking code for a living or a hobby, you may unwittingly interfere with other peoples rights. It may not just be the technical results that are unexpected; there may be civil or even criminal responsibility in extreme cases. Common sense dictates that knowingly providing pure cracking tools puts one at risk of prosecution. As new copyright legislation is bought by content industries to fence off certain types of knowledge, this is a risk that increases in magnitude when considering or countering

**Are geeks in peril?  
DAVID HARRIS cuts  
through the IT  
world's increasingly  
litigious climate to  
stop you becoming  
another IBM v SCO.**

anti-circumvention devices. At a less severe level, hacked code may damage someone else's property – through bugs – and this may lead to civil liability. Even if you provide a code that functions well, in the event it compromises someone else's interests you may end up sued. As we have seen, the answer of many corporations to technical competition is to patent and sue, regardless of the strength of their case. And of course as with most things Linux, the free nature of code leads to an exciting extra degree of legal uncertainty.





# LIABILITY FOR BUGS – SOFTWARE LICENCES AND CONTRACTS

**What are the prospects of you getting sued if your hacked code is buggy?** If a corporation dumps *II/S* for *Apache* and then gets rooted because of a new bug, they may want to sue someone, particularly if the bug causes big losses. What would be the liability of the (presumably sweating) *Apache* team? The reasonable view is that since the user didn't pay a penny for the code, suing over a defect is unreasonable. This is the policy position of the GPL licence.

FUD (Fear, Uncertainty, Doubt) is a marketing technique used when a competitor launches a product that is both better than yours and costs less, rendering your item no longer competitive. Microsoft FUD used to hold FOSS (Free Open Source Software) as unworthy for the enterprise market – although one could sue hackers, they had no money, so it was pointless and one should deal therefore with Microsoft instead. However anyone reading a Microsoft EULA with its rigid and comprehensive disclaimers might equally ask what the prospects of successfully suing them would be. Can Microsoft or a GPL hacker rely on exemption clauses to protect themselves? Like any good lawyer, my answer is a definite “perhaps”.

Where software is supplied, three principal (but not mutually exclusive) mechanisms define the legal relationship between hacker and user: contract, licence and legislation. The liability will depend in part on which one of these applies, so a brief explanation of the differences is in order.

## Contract v licences v statutes

A contract is a legally binding agreement between two or more parties where mutual lawful obligations are exchanged for value. For example, the purchase of a car is a contract whereby a buyer promises to give a sum of money to the seller if he in turn transfers ownership of the car to the buyer. Alternatively, legislation may give a right use the property of another regardless of its owners' wishes. George Lucas may not appreciate you writing a critical review of *Star Wars* and he may not give his consent, but criticism or review is permitted by copyright law, so he cannot prevent it. Lastly, is the right given by a licence; Lord Diplock in his judgment in *Gist-Brocades* declared that “a licence passes no proprietary interest, it merely permits you to do something with the property of another that would otherwise be unlawful”; a key point in the GPL. There can also be a blurring between all three categories: a licence can be contractually based or a ‘bare’ licence and a licence can be granted by legislation, such as a compulsory patent licence.

## Exemption clauses

The desire of software authors to avoid liability for anything going wrong is the motive for the exemption clauses seen in software licences, an example of which is seen in the GPL (see the boxout on page 8). It is tempting to read

these clauses and give up all hope of suing or fear of being sued. This would be wrong because exemption clauses are often not as effective at disclaiming liability as desired. It is this fear that the hacker must confront.

In assessing contractual exemption clauses, courts in most parts of the World will consider a number of factors:

**“Even if you provide code that functions well, in the event that it compromises somebody else's interests, you may end up sued.”**

what is the bargaining power of both sides? How much was paid for the software? Was the code for mass consumption or customised? Should the user have known of the exemption clause? Was there an inducement to enter the contract? Could the user have got a solution elsewhere? For a non-contractual licence like the GPL, the question would be whether the conditions are reasonable.

## Contractual liability

**HACKERS** In the UK the *Unfair Contract Terms Act 1977* (UCTA) and the *Unfair Terms in Consumer Contracts Regulation 1994* embody these factors. These provisions usually apply when a business attempts to limit its liability to a consumer; they partially redress the balance of bargaining power between a user and a powerful corporation. It is most often applied when a company imposes a standard term contract on the other side. When that does happen, a statutory test of reasonableness applies. Does it apply to the GPL? It seems unlikely the GPL is a contract between end user and hacker for reasons given below, but in any event if code is produced as a hobby, then the hacker is not ‘acting in the course of business’ as UCTA requires and it would not apply: the main liability the hobby hacker faces are the non-contractual risks.

**COMMERCIAL GPL CODE** While large swathes of GPL code is not produced commercially, this need not be so; IBM and many others do, and there will probably be a contractual basis that underpins or incorporates the GPL licence. Where that is so, UCTA and the tests of the reasonableness of any liability exclusion clauses would probably be applied.

The fact that often no payment is made directly for the code is probably not important, as it can be argued that these corporations while not charging directly for the code, often still derive a commercial benefit from it. For example, they may tie value-added services to provision of the code. Microsoft distributes *Internet Explorer*, *Outlook* and more for free; but these were properly included in the anti-trust action since there was an underlying commercial motive. The UCTA approach to all exemption clauses is to ask the question whether or not they are reasonable in the light of statutory guidelines. These guidelines



## DISCLAIMER

**THIS ARTICLE IS BASED ON UK law except where otherwise indicated. Substantial differences exist between UK law and that elsewhere. The consequence of all law varies greatly with individual circumstances and thus nothing in this article is intended as or should be construed as advice or acted on without seeking your own lawyers' advice. The author regrets that he cannot give personal legal advice.**

COVER FEATURE **LIABILITY**

« include the factors we mentioned on exemption clauses and also the question of whether public liability insurance cover was available at a reasonable cost, or alternatively whether it was possible to absorb any liability without insurance.

## Is the GPL a contract?

The legal underpinnings of contracts include the existence of so called 'consideration', intent to form a legally binding agreement and acceptance of the contract.

Consideration is the 'price' for which a contractual obligation is bought. It can be a right or benefit given, or a loss or disadvantage accepted by a party. Giving money is a classic form of consideration as it is an easily recognised benefit given; but accepting an onerous personal duty may also be consideration.

## "The fact that software might be free (as in beer) is unlikely to persuade a court that there's not a duty of care, and hence liability"

Similarly, the common law rule says that acceptance of the contract terms must be communicated to the other side. For software licence agreements, this can be very uncertain. Many lawyers doubt the validity of shrink-wrap licences, since the cases that have come to court in most parts of the World are rather contradictory: some US decisions have upheld them, others have not, and other common law countries including the UK have had similarly unclear decisions.

We often read terms saying "...by using this software you agree to the terms of the licence". Indeed the GPL says in clause 5: "...by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License..." Unfortunately there is no good legal basis to justify this. A hacker cannot compel acceptance of a licence and its obligations anymore than I can send you a contract (enclosing a five pound note) and claiming your car by saying "by opening this letter you have accepted the contract and the contract price. Please post the ownership documents and car keys by return post". For the GPL, a conventional legal analysis would suggest there is neither consideration nor an acceptance and hence no

contract. An inventive or desperate lawyer might try to fashion consideration by saying that the obligations on a GPL licensee are an onerous imposition: that to agree to reveal ones own future modifications rather keep them to oneself is precisely one of the burdens envisaged by the doctrine of consideration. The view is untested and most legal writers view the GPL as a bare license.

## Negligent code

An issue in software contracts is the exempting of loss caused by bugs or design defects although liability can also arise without any contract. Where there is negligence, so-called *tort* law (*ie* a non-contractual claims for a legal wrong) may impose a liability. Failing to read an RFC when coding a server, or to consider buffer overflows when writing software for secure applications can all give rise to issues of tortious loss. How well are hackers protected in UK law? In the most extreme possible case, UCTA makes it impossible to exclude, either by contract or a disclaimer notice, liability for injury or death caused by negligence. The fact that software is free (as in beer) would be unlikely to persuade a court in such a case to find that there is not a duty of care, and hence liability. Anyone thinking of hacking an Open Source heart pacemaker or an intensive care unit monitor would do well to take legal advice. In most cases however loss is likely to be just economic loss, which is harder to claim in tort. The courts approach the issue by asking if there is a so called 'duty of care' For this to arise there would need to be a sufficiently close legal relationship between hacker and user. Proprietary software released to the public for a price is thought by most commentators not to give rise to such a relationship since the number of users is potentially huge and courts are wary of opening the claim floodgates. Since GPL code is released even more widely, it is very hard on the same argument to imagine that a court would say there was such a relationship.

Another point to consider is whether the particular loss was foreseeable. Its not enough that a bug could cause a loss, it has to be foreseeable that it would cause that loss. A bug in a backup program may well cause the loss of data, but it would not be foreseeable that a company might be fined for filing late accounts because it had to rely on paper backups after a segfault. Finally, a court might well take the view even if the hacker were liable for negligence where the software is free (as in beer) and published for the

## GPL

### Liability exclusion clause

**11** Because the program is licensed free of charge, there is no warranty for the program, to the extent permitted by applicable law, except when otherwise stated in writing the copyright holders and/or other parties provide the program "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. the entire risk as to the quality and

performance of the program is with you. should the program prove defective, you assume the cost of all necessary servicing, repair or correction.

**12** In no event unless required by applicable law or agreed to in writing will any copyright holder, or any other party who may modify and/or redistribute the program as permitted above, be liable to you for damages, including

any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs), even if such holder or other party has been advised of the possibility of such damages.



benefit of all, that it would not be just and equitable to impose liability.

What would happen to a hacker with a generous judge? Can one be negligent and still avoid liability? Possibly; one often sees signs at the entrance to car parks saying "no liability is accepted for loss or theft of cars or any property in them". These are often binding on the legal theory that if one objects to such a disclaimer one can then drive to a car park without a disclaimer. The GPL attempts to provide a similar function. At common law, to exclude liability for negligence one has to say so explicitly, or it should be clear in the circumstances that no liability is accepted for negligence. Courts construe such clauses *contra proferentum* (against those who rely on them). The GPL doesn't explicitly mention negligence, so the question is whether its language can only refer to negligence. Let me illustrate the courts approach: in one case, where a car repair shop attempted to evade liability for a fire, the exemption clause was construed *contra proferentum* and said not to apply because several very different types of risk existed: negligence and also liability for fire, and the clause did not say to which risk it applied. It was therefore said not to be effective at all. However with code, negligent design or implementation seems the only realistic risk, and it might

have some chance of success. Disclaimers also have to allow you to evade the risk: a notice saying "Danger" posted on a field full of unmarked abandoned mine shafts would not be enough; one would need to say "Danger: unmarked mine shafts – keep out". It is therefore good practise to mention the buggy nature of code and the risks.

The free nature of GPL code opens up the possibility for software of some whackier defenses not normally available to the negligent hacker. One could argue contributory negligence if they failed to keep up with the latest version, or patch the faulty code for which they had the source and the rights to make a derivative work. I'd view that as a rather tender argument but perhaps worth trying. One could also argue more convincingly that there was consent to the risks of free software, since users constantly see the terms of the GPL and hear its exclusionary terms repeated on forums, see bug reports, and often see newer better or competing versions publicised. That would indicate that the flawed nature of all software applied to free software also.

There is also the view that you get what you pay for, which might someday lead to the ironic sight of a hacker being forced, by strategy, to argue before a judge that because his code was cost free it wasn't very good and the user was stupid to rely on it.

## LIABILITY FOR COPYRIGHT INFRINGEMENT

### Vicarious and contributory liability

**This was covered in issue 38 of LXF so go and dig it** out for a fuller discussion. Vicarious liability covers the situation where it is not practical to go after the direct infringer; say if he was a penniless student. Where someone rich is in a position of responsibility for that person and it is clearly better to go after them and it may be possible to force them to pay the damages whereas for a penniless student whilst one might win the case actually getting them to pay the awarded costs and damages might be impossible. In the famous McDonalds 'McLibel' case a few years ago millions of pounds were spent suing activists for defamation. McDonalds won but they never got a penny of their huge costs (though they knew that would happen). This is normally a considerable disincentive for suing in otherwise winnable cases. Vicarious liability is potentially a significant problem for employers; if the company sysadmin is not paying attention, and his users set up warez servers, it is quite possible that his employer will be held liable for vicarious copyright infringement. The accusation would probably be that through negligence they failed to exercise sufficient control which allowed the warez servers to operate. This is a growing problem for companies who fail to police their networks adequately; copyright holders are starting to go after companies where people run p2p networks. In that event, it is the sysadmin who likely to be the unfortunate scapegoat.

Contributory infringement is a related concept. While vicarious liability assumes that one person has the power to control or direct someone else, contributory infringement alleges that they assisted an infringer while either knowing what he was doing or that in the

circumstances they should have known. This is the basis on which \*AA are attacking p2p services. They argue that what these services do is knowingly assist copyright infringers. That they do so for a profit is an aggravating factor but not essential to the accusation: it contributes to

**"Litigation is not always pursued to obtain a legal remedy: crushing a financially weaker competitor may be the true goal."**

the moral indignation of the court. In the case of Napster the argument was probably well made, at least in legal terms. Other p2p systems have learnt the lessons of Napster by changing both their design and basis of operation. Systems such as Kazaa now attempt to avoid liability for contributory infringement. They do this by distancing themselves both from directly infringing acts such as storing indexes of infringing material, and also by ensuring that they are content agnostic systems, i.e. they do not focus on music or video alone but also chat text and images. Nonetheless as Kazaa are finding out, that that may not be enough; and Kazaa is trying also to move itself away from the reach of the US courts by operating from Australia and Vanuatu. Nonetheless, litigation is not always pursued to obtain a legal remedy but rather a business one: crushing a weaker competitor by forcing them to spend more than they can afford on expensive lawyers may be the true goal.

While the \*AA attack the higher profile p2p providers it should not be thought that individual



## COVER FEATURE **LIABILITY**

hackers, p2p or otherwise, remain impervious to threat. Anyone who hacks code that threatens the interests or rights of others is at risk. Jon Johansen, who created *deCSS* to enable people to view and decrypt legally owned DVDs, trod on the toes of Hollywood and faces an unprecedented attack as a result. Though initially acquitted of criminal charges, he now faces retrial; a Hollywood sequel: *DeCSS2 – the Empire Strikes Back*. The impending EUCD will further strengthen their hand. Likewise, while at the moment the music industry is pursuing Kazaa, it is unclear what approach they will take with Open Source development teams. The fear they have is that these small smart fast-moving hacker teams will produce the next generation of file sharing apps that they will find impossible to stop under current legal and technical regimes. There is a clear attraction in attacking such projects, regardless of the legal merits of any case. The hope would be to suppress threatening innovation and intimidate the like-minded. We have seen Blizzard games shut down the Open Source *bneta* project, notwithstanding a miserably poor case under both the DMCA and trade secrets law. Although I believe no accusation of contributory infringement was among the allegations made, it could have been. Similar projects could be susceptible to the same litigation.

It is hard to give a general view on what would defeat such legal attacks, given that the legal analysis depends on particular facts that vary greatly. With p2p applications, it is clear that liability has arisen from several causes: non-agnostic content such as audio or video files; unencrypted transmissions allowing powerful opponents to map network content and thus to make a good case that much of the network use is for copyright violation with insignificant non-infringing use; Kazaa was dumb enough to advertise its application with pictures showing copyright violations, a gift to opponents since it showed constructive knowledge of infringement.

### Reverse engineering

It is often stated rather baldly that reverse engineering is lawful. This is too wide an assertion. Indeed often it is lawful; reverse engineering an electronic or mechanical device is usually lawful, but reverse engineering software is only lawful when permitted by an agreement or the exceptions given in Section 50B of the Copyright Designs and Patents Act 1988. This provides that one can reverse-

engineer a program (defined as converting from a low level to high level language) where it is necessary to do so to create another program to interoperate with the decompiled one or any other program. Several conditions qualify this right:

1. It must not be used to create a rival program to the one decompiled;
2. The information necessary to achieve this isn't otherwise readily available;
3. The information can't be passed on to anyone not involved in the process of achieving interoperability;
4. No more reverse engineering than is necessary to achieve interoperability can be done.

For FOSS that creates a number of issues. It may be possible for Microsoft to prevent GPLd software like *Samba* reverse engineering its APIs and code by simply by licensing the API information on a non-free basis. The Act does not mention obtaining the information on fair terms or open terms. A court would be likely to regard something like the Microsoft 'shared source' licence as acceptable since public policy does not (yet) recognise FOSS as a public policy consideration. There is also a distinction between using the information to create interoperable programs and competing programs. *Samba* would be acceptable on this basis since it does no more than allow Unix and Windows file systems to interact. Consider instead however *OpenOffice MS Office* import filters. Are we using decompilation information to allow it and *MS Office* to interact, or are we using the information to create a program competing with *MS Office*? I think the former is right: what the Act is trying to prevent is people directly copying programs and using decompilation to get the knowledge to allow them to do so. Here we have a separate program and plugin filters (which are not therefore an integral part of that program) which merely enhance its utility. This view is bolstered by the Act which says "[decompilation is not lawful if he]... uses the information to create a program which is substantially similar in its expression to the program decompiled..." together with a recognition that the entire purpose of the Act is to increase competition in the software market and prevent technological lockout by the use of copyright law. Note that this means that it would be unlawful to decompile a proprietary program to create an FOSS competitor even if decompiling it just to allow proprietary files or protocols to be used would have been lawful.

## CRIMINAL LIABILITY

### Unlawful access

**It is not just Linux hackers who face risk; system**

administrators also do things to put themselves at risk. We have already discussed the risk of negligent coding, however there are also things administrators do that would give rise to potential criminal liability. It may be tempting to set up an unauthorised shell account on an employers system using the root privileges that come with the job. Some do this for the sakes of convenience: its a whole lot easier to unmangle *Postfix* from home than drive 30 miles through sleet and



snow. Equally, hackers sometimes put a backdoor in their code for similar reasons. Nonetheless the use of shell accounts or backdoors may be an unauthorised access within the meaning provided in Section 1 of the *Computer Misuse Act 1990*. Many people have the misconception that this Act is targetted only at crackers. That is wrong, employees can also be guilty of an offence where they knowingly exceed the clearly defined scope of their employment. It would seem however that there does need to be a clear definition of these limits, and it seems fairly rare in



practise for employment contracts to specify these. In the absence of a clear definition it would be hard to demonstrate criminal intent. Where there is a term saying 'no remote administration access' it would seem wise to obey that.

## Vulnerability scanning

A question often seen is about the legality of security scanning. Crackers will typically undertake electronic reconnaissance of a potential victim by using tools such as

**“Employees can be guilty of an offence where they knowingly exceed the clearly defined scope of their employment.”**

*nmap* to determine operating system type, open ports and other useful metrics. However since a scanning attempt will not by itself result in a system compromise, it is what is

known in criminal law as a 'mere preparatory step.' The act of buying a ski mask prior to robbing a bank is not in law an attempted bank robbery, it is just preparation – the attempt only occurs at the point the bank is stormed. After the scan it will be necessary to use another mechanism to connect to the port and attempt a compromise. In the UK this is therefore most likely to be regarded as a 'mere preparatory' act not an attempt. Note however that by the *Criminal Attempts Act 1981* the fact that it is technically impossible to use an exploit is immaterial to the criminality of an attempt. The best view would therefore be that port scanning *per se* is not criminal. Other forms of automated attack such as war dialling/driving or an exploit scanning where the connection mechanism is combined with the exploit code may well be criminal, and it is irrelevant that the target system is patched to prevent it succeeding. While an employee acting properly is probably safe from this type of risk, contract sysadmins who fail to obtain explicit authorisation may be at risk from any resulting ambiguity.

# PATENT INFRINGEMENT

**Patent infringement is probably the biggest potential threat to the FOSS hacker.** The danger comes less from the European and UK patent laws than American law. That said, recent changes to European patent office guidance has arguably lowered the thresholds needed for software patents. The clarification restated the law but in a way that shifted emphasis and definitions to more easily allow what previously was unpatentable. Patents are defined by their claims which are the fence-posts that stake out the area of monopoly into which others may not stray. When drafting claims the strategic objective is to make them as wide as possible in their application but not so wide as to risk being struck down by prior art. It is this very breadth that can make it rather hard for the layman to figure out if what they are doing infringes. Were I to draft a patent for a spade I really would call it “a portable blade for translocating loose material”. This is done not for the sakes of being obscure but to try and stop someone circumventing the language of my claim. If I called it a flat bladed metal tool for moving earth someone might merely make a non-flat kevlar blade for moving rubble and earth.

This obscure language means therefore that no hacker can sensibly wade through the myriad patent applications and patents just to see if he risks infringing; he just would not know and the only sensible thing to do is plough on in hope. In reality no FOSS projects seem to have fallen foul of this issue so far as I can tell. The risk lies mainly in the larger strategic field. Microsoft, for example, has patents related to .Net and also CIFS which is used by *Samba* to allow non heterogenous networks of Linux and Microsoft systems. The recent US DOJ (Department of Justice) / Microsoft anti-trust settlement has given rise to fears of a patent rampage with them either suing for infringement, demanding fees or discriminating against FOSS projects when granting licences. Whether their patents can be successfully circumvented is uncertain but in any event no project like *Samba* or *Apache* has the resources to fight companies like Microsoft or IBM. A brief and not very expensive patent action can cost the loser £50,000, more

extensive cases begin to cost real money – several million pounds is not uncommon. Microsoft has thus far played it fairly cannily by not litigating, but the future may be different – the DOJ is out of the way and the EU commission will make a ruling soon. After that Microsoft may be in a position to use its rights management systems embedded in *MS Word* to prevent interoperability with *OpenOffice*, or its SQLserver-based Windows file systems to break *Samba*. Patents and copyright law will play a large role in this. Nor is it just Microsoft that is a potential problem, the threat can come from even from UNIX brethren as SCO have recently demonstrated.

## Home users

We've discussed the threat of patents to hackers but what of home users? One defence to a claim of patent infringement under the *Patent Act 1977* is that the use is private and non-commercial use. A hacker who implements software encumbered by a patent can nonetheless use it privately without a licence. This is however a rather limited

**“Obscure language means no hacker can sensibly wade through the myriad patent applications to see if he risks infringing.”**

defence in practise. It probably would not apply to charities, schools or community projects, since this is unlikely to be a private use even if it is non-commercial. Commercial distros such as Red Hat or SuSE would be susceptible since they would provide the patented code for profit. Debian, Gentoo, LFS and other social distros may well be vulnerable too. Despite being non-commercial, the issue is whether using or providing the code is a private act, and a court is likely to define private in a narrow way that would permit it only for oneself, friends and immediate family. ➤

## « LEGAL REMEDIES AND DAMAGES

**If a court holds one liable of infringing the rights of** anyone in the ways we've just discussed we will come to the painful part of litigation; damages. Damages are designed to put the wronged party back in the position they were before the wrong. This is of course a rather contrived exercise in the case of personal injury. After a car crash it is not all that convincing to ask "How much money would compensate you for the loss of your leg?" In the case of intellectual property rights however, the task is much easier since their value is typically measured by a licence fee. Thus for a patent infringement, the damages would often be based on the value of a licence had the patent owner willingly granted one to the infringer and a similar approach would often apply to other intellectual property.

Damages come in several different flavours:

### Nominal

Nominal damages reflect the fact that although a claimant wins his case he has not really suffered a loss; an award of one pound would reflect this. If a claimant wins and the court regards him as a particularly vile person, it may award contemptuous damages which would be the lowest legal tender: one penny. The danger for the claimant in these cases is paying the costs of the action. During litigation, if the defendant offers a settlement and the court later awards damages of less than those the defendant offered, the 'winner' would have pay the costs of both sides. He might end up winning a penny and paying £100,000 in costs. This should be contrasted with the US where each side normally pays their own costs whatever the outcome. This is one of the underlying causes for a greater willingness in the US to litigate: the costs are much, much smaller.

### General and special

General damages are the essence of civil litigation and they have the restorative function we talked of earlier. Special damages by contrast compensate for specific items of loss: if you can put a figure on it, it probably falls in this category. Lost salary or profits or extra expenses would be typically be a special damage.

Damages will be available for a breach of the GPL. If a proprietary software company were to unlawfully embed free software within a proprietary product, the damages may be more than one might think. Since GPL code is often cost-free it might be thought that the damages for such a free package would be nominal only. This is not necessarily so; code released under the GPL can be released simultaneously under a proprietary licence for which a fee might be charged on a conventional model. The GPL seeks to encourage source code disclosure and the making of derivative works on similar licensing terms, and many thus release their code for no cost under the GPL simply to encourage use of their code and the GPL. The hackers concerned might then say "if you do not want to play by the terms of the GPL, fine, have it on a

### ABOUT THE AUTHOR

DAVID HARRIS IS AN IP barrister at [www.ukitlaw.com](http://www.ukitlaw.com) When not suing people, he writes 'depressingly poor code'.

non-GPL licence and pay me." A court would be invited to say that it does not lie in the mouth of a wrongdoer to say that his unlawful acts should be rewarded by damages at the lesser of these two licence costs. On this analysis a court would be able to ignore any zero cost to the code and measure damages by the cost of a licence for an equivalent proprietary closed product.

### Aggravated, exemplary or punitive

Aggravated damages arise when the defendants behaviour causes more loss than normal but which is not so bad as to require punitive or exemplary damages. Exemplary or punitive damages are the ultimate in

**"If a proprietary software company were to unlawfully embed free software within a proprietary product, the damages could amount to more than you might think."**

severity and the purpose is less to reimburse than punish and deter. English courts are much less inclined to grant these than American ones where damages can triple under this category: SCO are for example claiming triple damages from IBM for trade secret infringement. The award is seen in two principal types of case: in the first where an agent of the state (for example, a policeman or civil servant) behaves in an oppressive arbitrary or an unconstitutional manner; or in the second when a defendant behaves in a way calculated make more profit than would be awarded in damages. The latter is often seen where newspapers defame someone in the belief that extra sales will outweigh defamation damages. Could a geek or hacker fall into this latter category? If one talked not of a net profit but rather of inflicting a net loss, then possibly they could. For example, suppose someone deliberately published a copyright circumvention tool calculating that it would cause losses of millions of pounds to RIAA member companies, while damages awarded against him might only amount to a few tens of thousands of pounds. If this could be proven, an award of exemplary damages might well be made. ■



### ACRONYMS USED IN THIS ARTICLE

**EUCD** – European Union Copyright Directive  
**EULA** – End User Licence Agreement  
**GPL** – General Public Licence / GNU Public Licence / Guaranteed Public for Life. See the FAQ at [www.gnu.org/licenses/gpl-faq.html](http://www.gnu.org/licenses/gpl-faq.html) for more info  
**RIAA** – Recording Industry Association of America  
**UCTA** – Unfair Contract Terms Act 1977  
**UTCC** – Unfair Terms in Consumer Contracts Regulation 1994



# Professional LVM

**M**ost sysadmins will have been in the situation where their hard disk partitioning just isn't working, and a re-think is required. Whereas most Windows users create one large 'C' partition for all their programs and data, most Linux users realise it's much better to separate disks into partitions for various parts of your system. But what happens when you find one partition is full and another is empty?

The *Logical Volume Manager (LVM)* is the solution to your problem – it allows you to create one large partition handled by *LVM*, and sub-partition that into filesystems. This group of filesystems can then be dynamically resized to shift space from one drive to another, even while the filesystems are in use.

## Anatomy of LVM

*LVM* fulfils the need for a higher-level view of disk storage, in that space allocated to filesystems is allocated *through LVM* rather than directly on disks. A key problem facing users installing Linux, or indeed any other OS, is that one needs to choose a partition layout before installation. Linux users, knowing the advantage of partitioning their filesystem so that `/`, `/home`, `/usr` and the like are on separate partitions, split their disk up into parts for easier maintenance. However, unless a computer has a very specific task (eg web/ftp server), it's hard to predict how much space will be required in each partition.

For me, my home partition continues to bloat ever-larger with all the programs I write and test, and soon I'm likely to run out of space there altogether. When that happens, I either start trying to delete content or back stuff up to CD, buy a new hard disk, or reformat everything and allocate much less space to `/usr` this time around – I have a couple of gigabytes free there that I could really use in `/home`!

So, even though I have more than enough room on the drive as a whole, my faulty partition layout means that I need to shuffle data around to make space. With *LVM*, the whole of my `hda` drive would be one *volume group*, which could contain multiple *logical volumes*. One logical volume is the equivalent of a non-*LVM* disk partition, and would contain a filesystem as usual, eg `/usr`. One *volume group* can contain multiple *physical volumes*, which are actual devices capable of storing data – normally this is a hard disk.

To reiterate, one volume group contains multiple physical volumes, and each volume group can hold multiple logical volumes. This "group then divide" philosophy means that a system with three 30GB drives can combine all three into one 90GB volume group, then share out space from that into, eg, 50GBs for a `/home` logical volume, 20GBs for a `/usr` logical volume, 10GBs for a `/var` logical volume, and so on.

**Ever found yourself juggling data around to maximise your hard disk space? PAUL HUDSON says make your life easier with *LVM*...**

*LVM* makes a great choice for both small and large systems, but for different reasons. Small systems benefit as discussed above, as in my own solution. Home PCs have to take quite a beating with regards to what is installed and used on there, and the inherent unpredictability means that having the ability to re-arrange drive partitions hassle-free is a great boon.

## Why enterprise should be using it

However, enterprise users certainly have the most to gain by switching to *LVM*. Many of you reading this will be more than familiar with the pains of drive-juggling on large systems. When you have to deal with hundreds or even thousands of users, making sure there's enough space to go round can become a full-time job in itself. Adding new drives becomes a monthly task, and data regularly needs to be moved from old drives to new drives in order to free up space and keep things going.

With *LVM*, this can become much more efficient: all drives in one system combine together to form a single volume group. Adding new drives to the system is as simple as slotting the drive in, adding it to the volume group, and resizing logical volumes, distributing the space as appropriate. Furthermore, if you want to take out an old drive for replacement, the process is simply the reverse of adding a drive: resize the logical volumes so the drive is free, remove it from the volume group, then take the drive out.



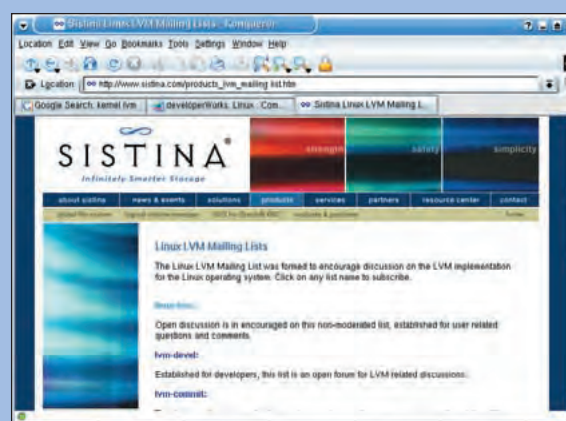
## PROBLEMS USING LVM?

Relax, you're in safe hands

The company who developed *LVM* for Linux – Sistina – maintain a mailing list for people using *LVM*. These mailing lists are archived on the Sistina homepage, and make for very informative reading.

Point your web browser to the archive URL at

[www.sistina.com/products\\_lvm\\_mailing\\_list.htm](http://www.sistina.com/products_lvm_mailing_list.htm)



The mailing list archive is packed with helpful question and answer posts from users taking advantage of *LVM*, and is a sure way to find solutions to any problems you have.

## STORAGE



If you're thinking *LVM* sounds like a RAID array, you're almost on the right track. While *LVM* is somewhat like software RAID, there are some key differences. Particularly, RAID can be configured to be fault-tolerant using both RAID 1 and RAID 5 formats, whereas *LVM* is not fault-tolerant at all. Furthermore, RAID can be configured to stripe data writes using both RAID 0 and RAID 5 for maximum performance, whereas, although *LVM* can be configured to stripe write, it makes no attempt to maximise performance, particularly considering that drives need to be connected to different disk controllers.

However, a key point is that *LVM* is a *logical volume manager* (hence the name) and NOT a filesystem like *reiserfs* or a disk sub-system like RAID. *Reiser*, or indeed *ext2* or *ext3*, sits on top of *LVM*, and *LVM* itself can sit on top of a RAID system. Confused yet?

Put simply, while *LVM* can be used to stripe write, it's often easier (and more performance-conscious) to use a separate



raid controller to create a RAID 5 array, and add that as a physical volume in a volume group. That way, fault-tolerance is provided for, as well as the usual performance boost associated with disk striping. Furthermore, extra fault-tolerance can be added by using a journalling filesystem on top of *LVM*, such as *reiserfs* or *ext3*.

Finally, one last cool feature of *LVM* is that it allows you to take a *snapshot* of a drive at a given time. This is an exact, byte-for-byte copy of a logical volume that can be kept safe as a backup. This backup can then be restored at a later date, if needed.

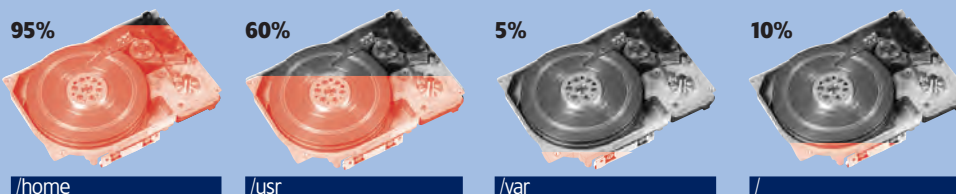
## Implementing LVM

Since Linux 2.3.49, *LVM* has actually been a part of the Linux source code. However, the version you have is unlikely be the latest, so you may wish to upgrade to the latest version. Sistina, the makers of *LVM*, have the latest version on their site, [www.sistina.com](http://www.sistina.com). At the time of writing, the latest

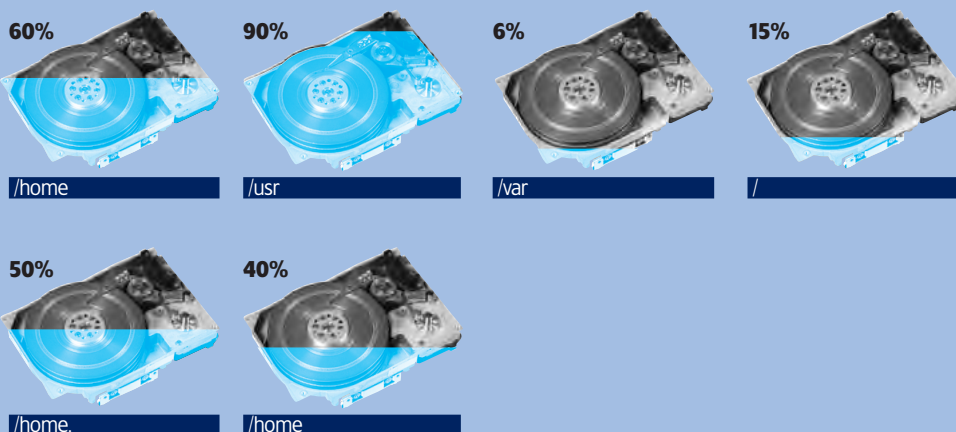
## LVM IN PICTURES

### OLD/TRADITIONAL

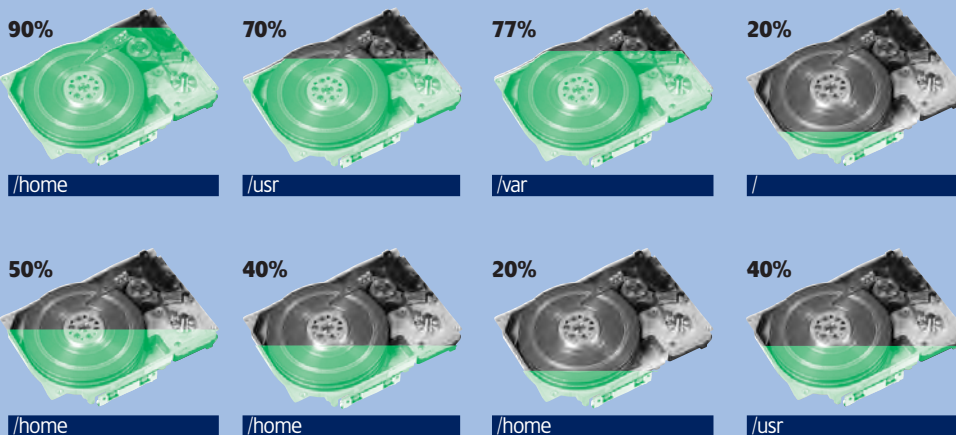
**1** In an example server, there are four hard disks. Each hard disk operates independently, and have been assigned their own mount points on the server – disk 1 is `/home`, disk 2 is `/usr`, disk 3 is `/var`, and disk 4 is `/`. There's also a percentage on each drive which show how full it is. As you can see, disk 1 is almost full, and disks 3 and 4 are almost empty.



**2** Here the system administrator has found that the `/home` directory has filled up with files as users save more data to the server. So, the sysadmin took the decision to add two more hard drives, then move roughly a third of users to each of them. There's still lots of space available in `/var` and `/`, but these are on separate hard drives. While it's possible that this space could have been used by symlinking some parts of `/home` to `/`, this would have been bad practice, and could have left a spidery tangle of links across the system as time goes by.



**3** Here the first drive has continued to grow, whereas drives five and six have not – presumably because the users that were moved to five and six were not the 'data culprits'. As a result, drive one has overflowed again, and so drive seven has had to be added, even though drives five and six have space available. Furthermore, drive two with `/usr` has overflow, requiring drive eight to be installed. Drives three and four still have heaps of space available. At this point, the administrator now has to try to juggle `/home` between four drives and juggle `/usr` over two drives. See the problem?





version is LVM 2.1.95.15. Here's how to install the stock *LVM*:

- 1) Download the Linux kernel source. You will need to rebuild your kernel to use LVM.
- 2) Download, compile and install libdevmapper from [ftp://ftp.sistina.com/pub/LVM2/device-mapper/](http://ftp.sistina.com/pub/LVM2/device-mapper/). Debian users can use

```
apt-get install libdevmapper-dev
```

```
3) tar xzvf ~/LVM2.1.95.15.tgz
```

```
4) cd LVM2.1.95.15
```

```
5) ./configure --prefix=/usr --mandir=/usr/share/man --with-kernel-dir=/linux-2.5.65
```

```
6) make
```

```
7) make install
```

You should now have the *LVM* tools installed. Next, go to your Linux kernel source directory, and enter make menuconfig. If you see errors along the lines of:

```
scripts/xdialog/dialog.h:29: curses.h: No such file or directory
```

Then you need to install libncurses-dev. Debian users can:

## MORE INFO

Be sure to read the Sistina product information at [www.sistina.com/products/lvm.htm](http://www.sistina.com/products/lvm.htm) to see how *LVM* can help you; updates are being released all the time, and it's definitely here-to-stay in the Linux kernel.

## apt-get install libncurses5-dev"

Once you're into the menuconfig kernel configuration screens, look under 'Multiple devices driver support' and select 'Device mapper support'. Finally, select any other configuration options you use, compile, and build the kernel.

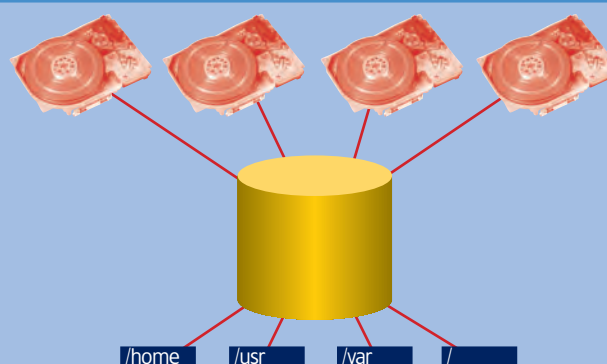
*LVM* comes with good documentation in man format, which will walk you through the process of setting up *LVM* to your customised layout – after all, no two people's setup is likely to be the same. You may find it helpful to look over the example *LVM* configuration file while you read the documentation – it's in doc/example.conf.

## In conclusion

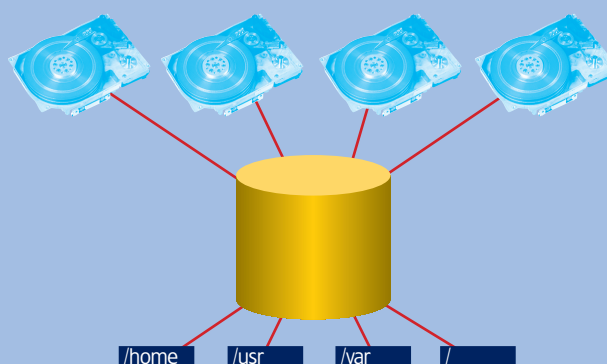
*LVM* is the way forward for both small- and large-scale users, of that there is little doubt. It will take a little time for many to accept the new partitioning paradigm, however, as the advantages are many, hopefully this process shan't take too long! ■

## NEW/LVM

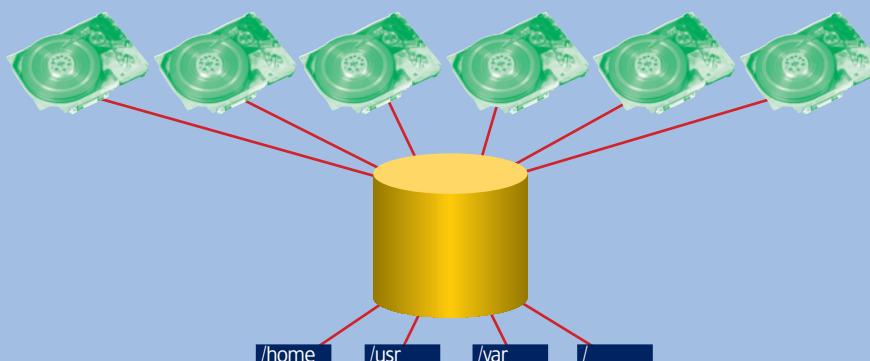
1 Here is the same system, this time with *LVM* in use. The four drives are placed into one volume group (illustrated by the gold cylinder in these examples), then logical volumes are created for each mount point: `/home`, `/usr`, `/var`, and `/`. All the hard drives have their space pooled into the one volume group, from which each of the logical volumes draw. Each logical volume doesn't know about the actual physical volumes (the hard drives) it's on, and doesn't need to.



2 Look the same? That's because it is! Whereas in the traditional partition layout extra disks were required, when using *LVM* the same number of drives are more than enough because some of the space allocated to `/var` and `/` can be taken away and reallocated to `/home`, so only four drives are needed.



3 Eventually all the space assigned to the volume group will be used up, and so the system administrator makes the decision to add new disks. As you can see in the diagram, two new drives have been added to the volume group, although this is transparent to the logical volumes. The new space can then be distributed across the logical volumes as required simply by resizing the volumes as appropriate.



## CASE STUDY



# Lineup for Linux

**V**arious UK police forces have experimented with Linux before, and we have included reports both in *Linux Format* and *Linux Pro* on their activities. The VIPER initiative could help all constabularies across the UK leverage Linux technology to their benefit.

VIPER stands for Video Identity Parade Electronically Recorded, and should give you a good idea of what is going on here. The system is designed to cut the cost and increase the speed at which ID parades can be performed – ever more important due to recent street crime initiatives from the government.

The traditional police identity parade involves gathering witnesses, suspects and 'lookalike' volunteers together. It sounds simple, but in practise it can take many weeks to organise, the major problems being finding a time that suits everyone involved, and finding volunteers who both the police and the suspect are satisfied with. Contrary to how it so often appears in TV cop dramas, in practice, the time taken to enact an identity parade is usually between six and eight weeks, at a cost of over £1000.

The problem with this system is not only in the cost and delay to the legal process, but also that witnesses' memories become understandably less reliable as time goes by. It is therefore in everyone's interest (except perhaps the guilty suspects' interests!) that the ID parade be carried out as quickly as possible.

## The solution

A combination of technology and cunning have found a way to address this problem. The West Yorkshire police, with the help of IBM and Sagitta, the storage services company, have designed a system which allows identity parades to be organised within a matter of hours rather than weeks. The concept itself is simple, though the actual system that makes it work may not be. Volunteers across the country are videotaped in the standard parade poses. The footage is then indexed and stored digitally on the main VIPER system. Operators at the VIPER bureau can then digitally combine the footage to create ID parades whenever required.

## NICK VEITCH investigates how VIPER is introducing Linux to the traditional police lineup.

Tony King, Project Manager at the VIPER National Bureau explains.

"We currently have a database of around 9,000 clips of volunteers. The suspect is filmed, usually a thirty second clip, which we edit down to 15. While that is happening, the suspect and solicitor, along with the police will select thumbnails of lookalikes from our system. They send us these serial numbers, and we can then edit together a parade. The standard service is two hours, but we can do more urgent requests – the record is actually fifteen minutes."

To cope with the new requests generated, the VIPER Bureau aim to increase the number of clips stored on the system. Tony King explains:

"At the moment we have about 9,000 volunteers on the system. We think that probably 12,000 would be a good average to have, otherwise you can just keep adding too many of one type of person. What we do need to do is get feedback from the stations on parades that they can't run with our system, and we have the inevitable form for that. They might tell us that they can't run a parade because there are no white males, 18-stone weightlifters or something. Then we can go down the town and look for this type – in fact we are going to Blackpool soon where they have a weightlifting competition.

"We keep adding to the database. We started with about 5000 people and we have been building it up from there."

## The Linux factor

The power behind this system is a well-configured Linux solution. It has to have the capacity to cope with changing workloads, scalability and still give good value for money. Linux was a natural choice, as Dave Leyland of Sagitta explained to us:

"The VIPER bureau system is basically a five node Linux cluster based on IBM Netfinity server technology. The servers feed 32 Workstations running professional video editing software. We discovered that the system copes well with this configuration and will behave predictably when there are hikes in workload, which is perfect for this case where demand varies across the day.





"Behind the servers we have deployed a SAN-based fibre-channelled storage solution. On the front end we feed out to dual redundant gigabit ethernet located in a DMZ. The suspect videos are taken wherever, and transmitted back across a VPN to the West Yorkshire server. The finished parades go back through this DMZ.

"The reasons we decided on a Linux platform were several-fold. The characteristics that Tony described in his requirements were for a system that had specific performance and scalability characteristics, and an availability characteristic, where we could actually prove the way it failed.

"Rather than taking a view where it's a kind of non-stop setup, we understood what items would fail and what the resultant capacity of the bureau would be at that point Linux fitted very neatly into that slot and we discovered during testing that it scales beautifully in this kind of environment – there's a very linear growth path."

"The other really important thing was using the GPFS from IBM, the parallel filesystem. When moving images around this parallel file having a single filesystem across the whole system is great, and we also found GPFS is great for load balancing."

The database of clips is far from static. Between 20–25% of all the video stored is being refreshed once a year; if only to take into account changing fashions and hairstyles. Sometimes updates have to be made more rapidly. Tony King explained:

"One incident happened last year where many people in the Chinese community changed their hairstyles, and there was a big panic. We went out and filmed about 150 Chinese volunteers with the new style.

"Of course we can amend the footage to put glasses on people and things, so you can make two lineups from one volunteer. We had a situation recently where police forces wanted us to put bobble-hats on people, because there was a spate of thefts with people wearing them."

And does the system work? "Yes," continues Tony:

"We've only had notice of one which failed in court, and we've done 15,000. We did investigate this case, and it did seem a bit suspect on both sides, and the judge decided that it wasn't a suitable identification. In the end it seems it wasn't a problem with the system, but with the way in which they had dealt with the prisoner.

"There are a range of people that this won't work for [for example, people with physical disabilities are not currently covered], but that's a very low percentage of the total."

## Expansion

However, the VIPER system isn't just local to the West Yorkshire police – it's a service available to other forces across the country. Obviously, for this they need to make some capital investment in systems to video capture their suspects and playback systems, but the system will usually pay for itself in a short time. The cost to each force per lineup is reduced to as little as £150, a substantial saving on the traditional model. Added to that that in some parts of the country, larger stations (eg Bristol) often undertake to create parades for local forces, they can be running hundreds of parades per week. The 50 or so police forces currently signed up expect to make a saving of £7million a year.

Tony King adds: "We're still expanding. We're currently taking orders from other police forces in England and Wales and they are paying to join. At the moment we are also putting a trial site into Edinburgh, and the Isle of Man want a suite, and Jersey are also interested. They might be more difficult.

"At the moment though we are averaging at 78 parades generated a day, seven days a week. I reckon we'll have half of the police forces signed up within a few months."

The VIPER system is also expected to form a large part of the National Video Identification System. Last year changes to the *Police And Criminal Evidence Act* made video parades the preferred option, and the NVIS initiative

**"One of the great things about using a Linux cluster... is that we can accurately match workload with the assets required."**

is an attempt to create a truly nationwide database facility for checking identity.

Is the Linux system going to be able to cope with this rate of rapid expansion? Of course it is, as Dave Leyland explains below:

"One of the great things about using a Linux cluster for this is that, historically this sort of integration would have involved predicting what kind of system you would need in two years time. Using the cluster approach, we can accurately match workload with the assets required. As the VIPER unit grows, it is simple to add more resources, whether that's more nodes, more storage or whatever, in a non-disruptive way. It's modular, but the Linux architecture lends itself to this very linear predictability. At the moment there is some headroom in performance."

"This sort of solution has come of age rather recently, and the catalytic component here is GPFS. This enables you, particularly in a media environment, to have these scalable solutions. Linux is a perfect OS for building these systems. The ability to scale a single filesystem over multiple platforms brings a whole new dimension to it, which is brought into sharp focus when you start trying to move large images around." ■

### Acknowledgments

With thanks to Tony King of the VIPER Bureau, Dave Leyland of Sagitta and IBM.



Tony King

**Since VIPER is much cheaper, it allows forces to spend more in other areas of policing.**





## Cross-platform GUI development

# Dream or reality?

**W**hen you read the extra-large PHP tutorial in *Linux Format* this month, you'll see that, by using the GIMP graphical user interface (GUI) toolkit GTK+, applications can be developed in various languages and then run on several platforms. For some, this probably sounds like quite an easy task to accomplish – after all, why should applications that run on Windows not run on Linux?!

The act of developing multi-platform user interfaces has a long (and very rocky) history, and only really now is it finally starting to come together.

### The story so far

In the beginning, God created machine code, and it was good, and he was happy with his creation. Now machine code was platform specific and terribly hard to understand. So God said, "Let there be an assembler" and assembly language came to pass, allowing programmers to use mnemonics in place of binary digits. But still, assembly language was platform-specific, and really not that fun to use.

Finally, in 1973, God created C, a language of such inordinate power and elegance that God needed to take quite a long break after creating it.

C (and its apparent successor C++) has been predominant for many years now – partly because it allows a great deal of control over hardware, but also because a program written in ANSI standard C can be ported to any other system which has a compatible C compiler. Compatible platforms include Linux, BSD, Solaris, Windows, HP-UX, and many others – C has been around for such a long time that a compiler exists for nearly every platform.

However, one thing C doesn't cover, due to its core philosophy of providing the *method* of implementation as

**What have Opera, Photoshop Album, and KDE all got in common? Find the answer below, as PAUL HUDSON investigates how cross-platform GUI development is now ready for prime-time...**

opposed to *actual* implementations, is any form of graphical library. C programs that are portable are command-line based and don't interact with the operating system much beyond basic POSIX-compatible calls.

Sun thought they had the solution with Java. It's cross-platform, has its own graphical toolkits in the forms of Swing and AWT, and also is powerful enough to tempt away quite a few C++ programmers. However, Java is compiled to bytecode and then interpreted at runtime, which slows execution time down substantially. Furthermore, most C++ programmers refused to make the jump from their language of choice. So, there was still a vacancy available for a toolkit that would allow developers to program in C and create fast, compiled, and, most importantly, *cross-platform* GUIs.

### 1 The Qt solution

Founded in 1994, Trolltech was founded with the goal of developing a cross-platform development framework for C++. Qt, one of its flagship products, is the embodiment of that vision – there's a native Qt library available for Linux, UNIX, Windows, Mac OS X, and also embedded Linux.

The largest proponent of Qt in the Linux environment is KDE, and indeed KDE was the driving force behind acceptance of Qt on Linux. Early versions of KDE were, however, powered by equally early versions of Qt, and before version 2.2, Qt was not available under the GPL. Naturally this state of affairs caused a lot of arguments amongst developers, and the end result was the Free Software Foundation started the GNOME project, which would use its own windowing toolkit, GTK – more on GTK later on in this article. As of version 2.2, Qt has been available under a GPL licence for Linux, so programming Qt applications is now much more acceptable.



## Skins and learning curves – Qt

With Qt, the default theme used to draw GUI objects ('widgets') matches the current environment, which means that Windows users get Windows look and feel, Mac OS X users get OS X look and feel, GNOME users get GNOME look and feel, etc. Whereas Qt *can* be skinned if desired, this works in marked contrast to Java, which defaults to its own Metal theme, leaving Windows users, GNOME users, and Mac users alike feeling alienated!

One of the biggest advantage of programming with Qt is that it is wholly written in C++, allowing developers to make the jump to GUI development while still retaining all their existing skills. Also, the learning curve is much lower, as user interface items are each implemented as C++ objects, and so are easy to pick up and use. One minor down-side is that Qt relies on several macros to get the job done, so there's a lot of "magic" involved.

Even though the fact that Qt is written in C++ is a particularly big advantage, for corporate users there is one even bigger advantage: you can buy an enterprise licence to Qt and thereby avoid the GPL requirement of distributing your source code.

## Corporate acceptance – Qt

It is this route that enticed big-names such as Adobe and Opera to choose Qt for their software development purposes. Adobe has recently released *Photoshop Album*, an image management product for Windows, which was programmed using Qt as opposed to Adobe's in-house toolkit. Mike DePaoli, the Engineering Manager of *Photoshop Album*, said "Qt simplified our task of developing *Photoshop Album* by providing high-level tools that we could customise to meet our needs". High praise indeed!

Opera Software also made the decision to use Qt for the Linux port of their Opera web browser, which means that two out of three popular browsers for Linux (out of *Opera*, *Konqueror*, and *Mozilla*) are built using Qt – not a bad achievement! Again, Opera sings the praises of Qt: "When we started to work with Trolltech three weeks ago, we did not expect to see a working browser on Linux so quickly. We are

A cross-platform app that's tired of living? See overpage...



impressed by the performance of the Trolltech team and the quality of their toolkit"

## Scripting and embedding – Qt

So, on the applications front, Qt certainly has a great deal going for it. However, Trolltech is continually upgrading and enhancing the library – the latest development is known as *Qt Script for Applications*, scheduled for final release in Q2 2003. *Qt Script for Applications*, or simply *QSA*, is designed to allow developers to add cross-platform scripting capabilities to their Qt applications using the *Qt Script language*. *Qt Script* itself, based on *ECMAScript*, is implemented as library that can be distributed with the application, along with the *Qt Scripter IDE* in order to allow end users to further customise scripts as needed.

**"A particular sticking point is performance – a native application will usually beat a Java app by quite a substantial margin."**

As with the main Qt library, QSA will be distributed under the GPL for Linux/UNIX, and it's very likely that future versions of KDE will take advantage of Qt Script.

With regards to porting to embedded Linux appliances such as Sharp's Zaurus PDA, Trolltech has even created a version of Qt for that, too: *Qt/Embedded*. Operating similarly to Qt for desktop systems from the user's perspective, *Qt/Embedded* is designed to be much more resource-efficient, whilst still only requiring a recompile. *Qt/Embedded* comes as part of Trolltech's *Qtopia* desktop suite, which provides a flexible application platform on top of *Qt/Embedded* to provide end users with a common operating environment on their PDAs (and is covered next issue).

## WHAT ABOUT JAVA?

### The need for speed...

**JAVA CONVERTS READING THIS FEATURE** will likely almost have been foaming at the mouth that I dismissed Java in just one paragraph, and perhaps quite rightly – Java has much to offer cross-platform developers in so far as ease of development. The Java language is marvellously easy to learn, sweeps away much of C's complexity, while also strapping on toolkits to process XML, display user interfaces, and more.

Although many programmers actively disparage languages that insist on including default methods of implementation for problems, Java has

gained a fairly solid level of respect in the community. However, a particular sticking point that still remains is performance – a native application will usually beat a Java application by quite a substantial margin.

In the modern world of graphics-heavy user interfaces, performance has never been more crucial, and as such Java doesn't generally cut it. Luckily, the Java team recognise this fault, and a variety of performance-improving enhancements are aimed at both J2SE 1.4.2 and 1.5. Also, ironically enough, 1.4.2 will debut a new pluggable look and feel – GTK!

## The big downside for Qt

By now you're probably quite impressed with Qt's capabilities, and quite rightly so – it's a very flexible and well thought-out toolkit that has a lot to offer developers at all levels.

However, at the end of the day Qt is still owned by a company, and the general nature of companies is that they want to make profit from what they do. For Trolltech, income is generated by way of selling the commercial development licence for Qt, and also selling access to the *Qtopia SDK*. Qt licensing is handled on a per-developer/per-platform basis, for example to allow one developer to write applications for Mac, Windows, and X11, the licensing fee is \$4660 – a hefty chunk out of *anyone's* chequebook!

As mentioned already, though, Qt for X11 (Linux/UNIX) and also *Qt/Embedded* are both available under the GPL, so open-source Linux Qt development doesn't come with any licence fees. However, the GPL stipulates that you must make publicly available your application's



## CROSS-PLATFORM

« source code if you make use of GPL code, so the GPL Qt licence isn't particularly popular with commercial users. Furthermore, for Windows and Mac, there is no GPL option available: to port your applications to Windows and Mac, you *must* purchase a licence.

Unsurprisingly, quite a few people are unhappy with this state of affairs, and there have been many attempts to create an open-source clone of Qt that would be licensed under the GPL or even the LGPL. If you're unclear of the difference between the two, the GPL stipulates that any applications distributed that make use of GPL code must have the source code made available for users to read, whereas the LGPL does not have such a stipulation, and can be used inside proprietary programs.

## "If you're willing to lay out the cash for the necessary licences, Qt is the perfect choice for proprietary development."

The current favourite Qt clone for Windows is part of the KDE/Cygwin project, and aims to port Qt to Win32 by taking the Open Source Qt available for Linux and replacing all the X11 function calls with their Windows equivalent. Since Qt is over 80,000 lines of code, this is no small task. However, last time a project managed to get any headway with creating a clone of Qt, Trolltech took notice, and it was probably one of the primary reasons that Qt for Linux ended up being GPL.

### The Qt verdict

At the end of the day, if you're willing to accept non-free licences for non-UNIX operating systems, Qt really is the powerhouse you're looking for. It's wholly C++, infinitely



skinnable, database enabled, embeddable, and, soon, it will be scriptable too.

If you're willing to lay out the cash for licences, Qt is the perfect choice for proprietary development: a simple re-compile allows Qt apps to work – with native binary and native look and feel – on UNIX, Mac, Windows, and PDAs.

## 2 The GTK+ solution

The GIMP Toolkit (GTK) started its life as a result of Qt originally not having a free licence. When Qt switched to a free licence for UNIX platforms, GTK lost a little of its support, but Qt is still not free for Windows and Mac development, so there is still a good argument to have another GUI toolkit in development. Furthermore, GTK has managed to distinguish itself as a great toolkit in its own right, and shares much of the functionality of Qt. GTK is now available entirely under the LGPL licence, which means it may be used in both Open Source projects and proprietary projects alike without the need to keep source code open.

### Skins and learning curves – GTK

For many users, the look and feel of the front-end of an application is almost as important as the functionality. GTK is as skinnable as Qt in that the styles, colours, and 'feel' of GUI objects is customisable using themes. However, GTK is a little less intelligent than Qt because it defaults to its own internal styling when not informed otherwise – and the default styling is far from attractive! Qt on the other hand uses the default look and feel for the environment on which it is running.

A further problem is that GTK is skinnable on UNIX only – GTK applications running on Windows use the default GTK theme and nothing else, which is a shame,

## WRITE YOUR OWN QT PROGRAM

It's easier than you think!

QT IS THE MOST POPULAR TOOLKIT OUT there, mainly thanks to KDE. So, to give you a quick taste of cross-platform programming and how easy it can be, here's a short guide to help you get started using Qt and C++.

To compile this program, you'll need the libqt-devel headers and libraries installed. This was tested on Qt 3.1 using GCC 3.2. First, create a file **qtgoodbye.cpp**, and bring it up in your editor. Next, type in the code below, and save the file:

```
#include <qapplication.h>
#include <qpushbutton.h>
#include <qfont.h>

int main( int argc, char **argv )
{
    QApplication a( argc, argv );

    QPushButton goodbye( "Goodbye, World!", 0 );
    goodbye.resize( 250, 40 );
    goodbye.setFont( QFont( "Helvetica", 18,
```

```
QFont::Bold ) );
    QObject::connect( &goodbye, SIGNAL(clicked()),
&a, SLOT(quit()) );
    a.setMainWidget( &goodbye );
    goodbye.show();
    return a.exec();
}
```

To compile that, simply enter:

```
gcc -O2 qtgoodbye.cpp -o goodbye -lqt
```

Once the program has compiled, run **./goodbye** to execute your program. Clicking the button will terminate the application. There's not enough room here to explain exactly how the code works, but the C++ nature of Qt should fairly obvious, and also how easy it makes writing a GUI.

If for some reason you get errors like **application.h: No such file or directory**, you probably don't have your Qt3 headers installed and configured correctly. Try running **locate application.h**. If you don't see **application.h**



listed somewhere, you don't have the correct files installed. If you see it installed, but its not in a place GCC usually expects it to be, try this: 

```
gcc -O2 -I/usr/include/qt3 qtgoodbye.cpp -o goodbye -lqt
```

The use of the **I** flag (that's a capital i, not a lower-case L) specifies a particular include directory for header files, and allows you to specify where your **application.h** file is. In my case, it's in **/usr/include/qt3** – you'll need to customise it yourself.

If you get seemingly incomprehensible errors like **/tmp/cc7pETfk.o(.text+0x23): In function 'main':**, then there's a problem linking with the Qt library. Make sure you have a **libqt\*.so\*** file in **/usr/lib**.



considering how many other programs allow this as a matter of course.

GTK is written in C, which isn't terribly easy to use for GUI development – widgets are created using a wide variety of functions that basically you just need to memorise. This is mitigated somewhat by the fact that GTK has fairly extensive documentation available to help explain how things work.

One major advantage to GTK is its GUI designer (no, that's not a misprint!). *Glade* is a GUI tool designed to help you develop GUIs, and generates source code for C, C++, Ada95, Perl, and Eiffel. Qt has its own version, *Qt Designer*, which has much the same functionality except without the cross-code functionality. Using *Glade* cuts design time in half or better, and, through the use of the *libglade* library, one can even load a *Glade* user interface directly – no UI code required.

So, while the base of GTK is written in C, there are language bindings for many other languages, including C++. In fact, the C++ port of GTK (*gtkmm*) is so feature-complete as to be technically superior to Qt in language terms, even though Qt was written natively for C++! STL programmers will be happy to hear that *gtkmm* uses `std::string`, `std::vector` and others, whereas Qt uses its own custom versions.

## Corporate acceptance – GTK

For one reason or another, GTK lacks acceptance in the corporate world in so far as there are as yet no major proprietary applications developed using GTK. This is quite an odd state of affairs, given that there are over 350 Open Source applications written using GTK, that GTK is completely free, and that GTK can be used from a variety of languages.

It's possible that GTK's licence works against it in some ways, particularly because corporate developers are often suspicious of the word "free". However, any worry over GTK's licence is spurious at best, because its licenced entirely under the LGPL and so can be used in proprietary applications without the requirement to distribute the source code.

One potential hiccup is that GTK's support for other platforms is known to be less than stellar. The Windows port is stable and complete, as is the original UNIX version. However, the Mac version is still under development, and isn't likely to be finished for the next few months.

The current state of the Mac port, at least for OS X, is impressive nonetheless, though – there are actually two separate efforts going on, and both allow native operation of GTK applications for OS X. Furthermore, both versions share the attractive native look and feel of Mac OS X by default, and, in time, the OS X port should join the others, which in turn should make it acceptable to a wider range of developers.

One advantage in GTK's favour is that it is entirely Open Source, and so has the open source philosophy driving it. In practical terms, this means that porting to other platforms is often done by enthusiasts looking to help out. In the case of GTK, this has resulted in a port to the ill-fated BeOS, amongst other things!

**The OS X port of GTK is coming along very well, and defaults to the visually pleasant OS X Aqua look.**



## Scripting and embedding – GTK

At the time of writing, GTK has no method to allow its interfaces to be scripted at runtime, which puts Qt in the lead in this oft-ignored area. However, scriptable interface components are something that many people can live without, so the lack of any script engine in GTK can hardly be held against GTK. Furthermore, *Qt Script* is still under development, so it's new to *everyone* at this time.

With regards to embedding, things get a little hazy. Whereas Trolltech actively produces and markets *Qtopia*, GTK's effort is more fragmented. The one project that is leading the way is the GPE Palmtop Environment, which uses X Windows and GTK to draw its widgets. As can be seen at <http://gpe.handhelds.org/screenshots.shtml>, the GPE project is remarkably far advanced, and is capable of running a wide variety of GTK programs on handhelds. However, again it suffers from lack of corporate acceptance, which is hopefully something that will improve over time.

## The big downside for GTK?

So GTK is completely free, cross-platform, multi-language, and embeddable. Are there any down-sides to using it? Well, not really, no – GTK is a legit WYSIWYG development toolkit with no surprises lurking. The licensing means you can start developing right now with no fear of legal battles later, and the fact that GTK is accessible through so many languages (even PHP!) should put a smile on the face of every developer.

## The GTK verdict

Free as in beer and cool as in wow, GTK is all set to take the world by storm. However, it still lags quite a long way behind Qt in the cross-platform stakes, probably more due to the lack of refinement than anything else. With the release of GTK2, a lot of work was done behind the scenes to clean and improve the API that should bolster the overall effort.

With regards to whether it will be enough to allow GTK to compete with Qt, it's anyone's guess – Qt have a big lead, both technologically and also in market share, and certainly aren't winning to give that up.

Whether you're willing to spend money or not, GTK is an all-round good choice. It's not yet ideal, because some of the ports aren't 100% there yet, but things are improving at quite a drastic rate. GTK has been around less time than Qt, and is still stabilising – with time, unless Trolltech changes its licensing policy, GTK may well prevail. ■

## SECURITY



# Is Linux Secure?

**T**here continues to be fierce debate within the IT world, much expressed in online security forums, about how secure Linux really is. At the end of the day, Linux is an operating system, and no operating system is 100% secure. However, the question should not be "Is Linux secure?" but "Can Linux be made to meet the security requirements of its users?" I believe the answer to the second question is, undoubtedly, yes.

During any of the following discussion, sight should not be lost of the fact that securing of Linux should take place within the auspices of an overall security framework. Users should be aware of what they need to protect, and what they are protecting against. A risk management approach should be adopted, and acknowledgement made that procedural and physical measures have their role to play in an overall security policy and strategy.

Your "straight from the box" Linux distribution (or download) is a very powerful, fully featured operating system, executing many services. Due to its Open Source nature, it has undergone a process of broad continuous peer review. It is therefore less likely to suffer the flaws suffered by other vendor software, and will keep on being reviewed and improved, with frequent fixes and updates.

## Functionality == security risk?

It is inherent in the 'fully featured' nature of Linux, that it has a huge level of functionality, and therefore security risk. High levels of functionality are often the bane of the security world, as they give the hackers too many tools and exploits to work with. The aim of the home user or system administrator should therefore be to minimise the opportunities presented to these groups.

Linux can, within reason, be made as secure as required by the users needs, by a process of installation, configuration and ongoing maintenance. The first two items may be obvious, but the third one is at least as important. Any graph showing attacks against computer systems will show an increasingly upward trend. The attackers are growing in numbers, and the attacks increasingly sophisticated. Sight should never be lost of the fact that security is a continuous process.

Installation is the first step in the process of securing your Linux system. Many of the services that come as standard with Linux are unneeded, and/or pose a potential security risk. So the components you want to install should be chosen carefully, to give the functionality you need, without providing an unacceptable security risk. For

**PETER MACKLEY  
of Marconi  
analyses whether  
Linux is really  
more secure than  
any other OS.**

instance, do you really need to install *fw*, *whois*, *portmap* or *rusers* on your system? Keep the services you install to a practical minimum, understand the vulnerabilities associated with them, and continuously review the system's requirement for them.

## Securing the system

Next come the configuration aspects, which means securing the services and software we have decided to run on our system. This includes configuring such items as password policy, the root account, setting console access and kernel parameters. It is vitally important that the kernel is secure, as it controls networking functions, so compile it minimising the level of functionality to that required. Hardened Linux kernels are popular among Firewall vendors, because of the added security it brings; and hardening utilities such as Bastille Linux are freely available. Again, review your configuration on an ongoing basis, so it continues to effectively meet the demands of your policy and any new requirements.

Last, but certainly not least (as indicated earlier), there is the ongoing process of keeping your system up-to-date, in order to address new exploits or vulnerabilities. Monitor your log files, keep up to date with the latest security fixes from your Linux distributor, and subscribe to security alert sites, such as [www.securityfocus.com](http://www.securityfocus.com). With Linux, you can be assured that when security flaws are discovered, they are patched much more quickly with Open Source than with proprietary software. You can also be safe in the knowledge that the problem has been fixed at source level, as source for fixes is freely available to all, and has been extensively reviewed.

Finally, a word on viruses. According to the Symantec Internet Security Threat Report, in 2002, 80 percent of infections came from *just three* Windows 32 viruses. Although it warns that Linux is becoming more popular with virus writers, it has a long way to go before reaching the level of popularity that some operating systems have within the malware community.

If you have, or are intending to run a Linux system, take from this that Linux can be a secure platform. It is certainly more configurable and controllable in this respect than other operating systems from well-known vendors. You can implement Linux the way that suits you and your requirements, not just the way the operating system allows you to. In this respect, it is the ideal choice of operating system where security is a significant requirement. ■

## INFOSEC EXHIBITION

**Marconi Selenia Secure Systems Ltd are exhibiting at Infosecurity Europe, the largest and most important information security event in Europe. Now in its 8th year, the show features Europe's most comprehensive FREE education programme, and over 200 exhibitors at the Grand Hall at Olympia from 29th April - 1st May 2003.**  
[www.infosec.co.uk](http://www.infosec.co.uk)



# The Pure and Simple Truth

**T**here is a special T-shirt that I wear to every Disaster Recovery test that I am involved in. It has a quote on it from Oscar Wilde: "The pure and simple truth is rarely pure and never simple."

As a rule, when developing articles on the security of operating systems, the usual order of the day is to focus on topics like Footprinting, Scanning, Enumeration, Penetration, Privilege Escalation, Backdoors, Session Hijacking and Social Engineering. But not today.

Most technical security professionals already have a grasp of each of these subjects. This is how to view the security issues on your systems from a hacker's viewpoint. This article will view the issue from the other side of the fence.

## What layer is Security?

Linux Security is no longer just the tightening of security on a single device within the organisation. Devices in the Corporate infrastructure are now all interconnected from

OSI Layer	IP Protocol Suite	Network Devices	Attacks	Counteractive Measures
7. Application	Application	Gateway	Web Spoofing	DNSSEC
8. Presentation			DNS Cache Poisoning	
5. Session			DNS Spoofing	
4. Transport	TCP/UDP	L4 Switch	Message Flooding HTTP Tunneling RPC (Information Gathering)	Secure Portmap CHAP
3. Network	IP	Router & L3 Switch	TCP Hijacking Denial Of Service (DOS)	Firewall configuration
2. Data Link	MAC	L2 Switch Bridge	IP source routing Root Spoofing IP source address spoofing	Router Configuration Routing Protocol Security Firewall configuration
1. Physical	10-Base?	HUB (Repeater)	ARP spoofing MAC Address Table Overload	Static ARP tables Switch Configuration
			Wire Tapping	Physical Security

layer 1 (the cabling infrastructure) to layer 7 (the applications). If you are not getting to grips with the whole design, you're not seeing the complete picture.

## Microsoft?

*Samba* is enabling MS Windows PCs to connect seamlessly to Linux hosts. Linux is making inroads into the corporate world. Mail servers are running on Linux systems that allow an *MS Outlook* client to function as if it were connecting to a MS mail server. Does this give you a taste of the problem?

## Getting Physical?

We all know that if you lose the administrator or root password to a system, you can still get into the system if you have physical access to the device. Think about that for a minute: if you can touch it, you can get *complete access* to the device, given time (and you won't need a lot of it!). What is the physical security surrounding your devices?

## The dreaded Ps

I used to find Policies and Procedures a real downer; paperwork that got in the way of a dynamic working environment. Why would I want to do all that stuff? I know

## RIE DODSWORTH of IT Security Audit takes an alternative look at security policies.

how to do my job don't I? Well, now I am a convert. Policies and Procedures are a great way to focus on the big picture of security and then work the way down to the fiddly bits.

Policies & Procedures are a bit like developing a training course; you have to focus on the big picture when defining the Policies and get the Procedure microscope out for each of the little fiddly bits.

You can't get a good grasp of a problem until you can feel all the lumps and bumps.

## Who wrote this OS anyway?

There has been a lot of discussion in the past about how open systems are better than proprietary systems, from a security standpoint, because the bugs come out quicker and are fixed faster. Let me just say that I don't believe that obscurity is a valid security methodology.

Microsoft has been given a hard time over its security record. True, it may have made some mistakes in the past (and present), but there have been a number of real corks in the Linux world too. Do you remember when SSH had a nice little hole placed in it and almost everybody had that hole on their systems?

What about the RPC buffer overflow hole leading to the execution of arbitrary code on practically every system? Each day, there are many new vulnerabilities made public.

How do you keep track of these new pieces of info?

## Who wrote the Application?

This is not the same as dealing with the operating system. Applications require access to resources within the system and possibly other systems. If the application requires high level rights and can be compromised, then your system(s) can be compromised.

I have seen, on many occasions, where in-house developers are also the live system maintainers. This may not at first glance appear to be a problem, but let me ask you a couple of questions:

Do your developers have access to the live systems from their development systems or access both the development and live systems from the same workstations?

Do the development systems have users configured with names like 'test' and with passwords like 'test'?

The answers to each of these questions may feel safe enough on their own, but join these two thoughts together and you may start to feel an unpleasant sensation. Add to this the thought that bugs may be found in both off-the-shelf packages and libraries used by many applications.

These types of bugs are posted every day on the Internet. How do you keep track of these new pieces of information?

IT Security should be an integral part of corporate thinking. Not just a technical issue. ■

## ABOUT THE AUTHOR

**Rie Dodsworth is one of the founders and Security Operations Director for IT Security Audit Ltd, and has been in the Networking and Security industry for more than 17 years. IT Security Audit are exhibiting at Infosecurity Europe, Grand Hall at Olympia from 29th April – 1st May 2003.**  
[www.infosec.co.uk](http://www.infosec.co.uk)