

# Use and Usability

**F**orgive the title, you haven't strayed into a Jane Austen appreciation magazine by mistake. But the use of Linux, and its usability are themes which crop up again and again in the magazine for obvious reasons.

Perceived usability issues seem to be the cosh with which attempts by Linux apps to enter the desktop market are beaten back. But usability may cease to be an issue for followers of HP's grand vision of the future – a web-pervasive world where the real users are quite often other devices, and the people who employ the technology never have to break a fingernail. Find out how this grand scheme is supposed to work, and the part open source software has to play, in our feature starting on page 48.

Taking a completely different look at actually being able to use Linux, or computers at all, we have a special report on the recent accessibility conference hosted by Sun Microsystems, and the part Unix and Linux software like GNOME has to

play in making technology usable for the 750 million people worldwide with disabilities. The issues and the progress towards solving them are outlined in our report on page 12.

Ease of use was probably one of the factors you considered when you voted for the winners of the Linux Format awards 2001. Thousands of you took the time to register your preferences, and you can get a full rundown of the action and the worthy winners starting on page 54.

There may be some candidates for the next awards under scrutiny this issue – we have reviews of IcePack Linux, Heavy Metal: F.A.K.K.2, SuSE Firewall, Moneydance and plenty more this issue – turn the page for full contents.

And of course, there is another bumper crop of tutorials, covering everything from web servers to flash-card readers, CGI-scripting to Java serialization. There's more than enough here to keep you busy for another month, so I'll let you get to it...



**Nick Veitch** EDITOR

**Gaze into HP's crystal ball and find out why open standards are at the heart of their future visions p48**

**Linux and accessibility – we report from Sun's recent conference p12**

**Don't forget to check out our HotPicks – the best new open source software! p42**



12



42



48

## LINUX FORMAT

### 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...



**Richard Smedley**  
As well as compiling the Help Wanted and LUG pages, Rich is responsible for all the spelling mistakes.



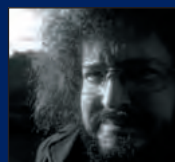
**David Coulson**  
Our Answers guy is a networking and security guru with plenty of sysadmin experience.



**Richard Drummond**  
As well as writing our Java series, Rich co-ordinates most of the reviews in the mag.



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



**Charlie Stross**  
Master of Perl, Charlie has been writing about Linux for more years than anyone can remember.

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

**Hoyt Duff**  
Fishing pier proprietor Hoyt spends his spare time installing Linux on anything that stays still long enough.

**Simon Goodwin**  
A hardware druid in more ways than one, Simon is currently researching every emulator known.

**Andrew Channelle**  
Now studying 'culture' or some such nonsense, Andy still finds plenty time to write the news!

**Maurice Kelly**  
A coder by day, this is the chap who spends his evenings scouring the 'Net to bring you HotPicks.

### Contact us

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

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

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

Disc problems:  
[lxf.support@futurenet.co.uk](mailto:lxf.support@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

# CONTENTS

# LINUX

FORMAT

LXF27 May 2002

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

## Accessible Linux for all

As Sun pile developers into GNOME, read about the practical consequences as we round up the developments unveiled in Linux accessibility.



12

## Security Roundup

We take a look at tools for keeping the script kiddies at bay and tightening up network security on Linux. Don't leave it until it's too late!



36

## Awards winners

You voted for them. Now see the results of your crosses (OK, clicks) as we count the votes and unveil all that was best in Linux in 2001



54

**COVER FEATURE****WELCOME TO**

# COOLTOWN

# 48

From joined-up thinking  
to joined-up lives – HP's  
Linux-networked future



Save money and subscribe to Linux Format. See page 96 or phone 0870 444 8645

## »» REVIEWS

### 22 HEAVY METAL FAKK 2

Find out if the gameplay is better than the plot as you pit Lara-like Julie against various foes.



### 24 ICE PACK LINUX

Friendly new graphical distro with a good choice of window managers and an easy install.

### 26 MONEYDANCE

Money makes the world go round. Now Linux has a slice of the pie in the form of this SOHO accounting package.

### 28 SUSE FIREWALL

We try out the easy-to-configure firewall-on-a-disk from those masters of user-friendly configuration, SuSE.

### 30 CRISP

Extensibility, source code analysis tools and a flexible approach to windowing from this commercial editor.

### 32 PRO VSD

If you're looking at providing hosting facilities with plenty of features, *ProVSD* offers an easy solution.

### 34 BOOKS

Designing with JavaScript; Web Security, Privacy and Commerce; Python and XML; and XSL Essentials.

## »» TUTORIALS

### 68 APACHE

Configure *Apache* and start serving web pages, then move on to name-based virtual hosting.

### 72 PERL

More on *manserver* – a client/server protocol for man pages – and other network fun.

### 76 JAVA

Customize the way your classes are serialized as we delve deeper into Java's serialization mechanism.



### 78 KYLIX

Following last month's examination of error handling with a look at raising exceptions.

### 84 FLASHGO!

Plug three USB portable data storage interfaces into your USB ports? No need with *FlashGO!*



### 86 CGI

Introducing a short series on the Common Gateway Interface. Using Perl to provide dynamic web content.

## »» REGULARS

NEWSDESK .....	6
MAILSERVER .....	16
HOT PICKS .....	42
ANSWERS .....	90
HELP WANTED .....	108
USER GROUPS .....	110

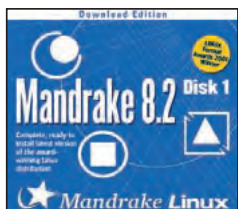
## »» FEATURES

HotPicks .....	42
Abiword, Bomber Instinct, Sylpheed	
Idaya Competition .....	57
Win a ten host VSD licence	
What On Earth .....	58
Is the Linux Standard Base?	
Emulators .....	62
Nintendo from 16 to 64-bit	

# Coverdiscs

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

Not one, but two CDs of **Mandrake 8.2**! Plus all the usual extras and code from the magazine. DVD users get the full 3 CD Mandrake as well as Smoothwall and Familiar – a distro for iPAQ h3600. Plus *Knoda* – the desktop front-end to *MySQL*; *Gspy* motion detection software for security cameras; *webmin*; *Wine*; latest *Galeon* and *Mozilla*; *OpenOffice* and much, much more...



# Newsdesk

Mandrake, Red Hat and SuSE have all been busy - but so has the GNU project. New office suites are imminent; and a new relational database. Meanwhile AOL get the Linux message.

## THREE MAJOR RELEASES

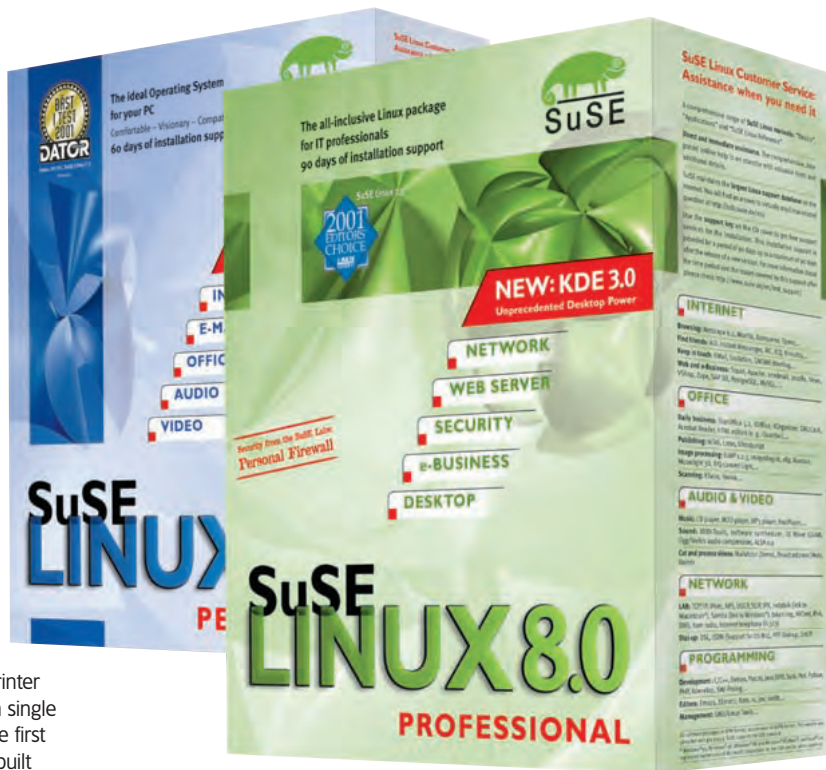
# Distro deluge from the

**SuSE 8.0.** SuSE's release schedule is hard to keep up with for even the most dedicated fan, but an x.0 release usually means more than mere tinkering. The theme for this version of the software is 'Linux easier than ever' and the biggest addition is KDE 3. SuSE are also trumpeting better security, a 'superfast' installation routine that needs little input from the user and improved multimedia facilities.

One area that SuSE's distro has always been strong in is partitioning and hardware detection, and 8.0 looks like carrying on the tradition, needing only a few details before configuring everything from printer and network to TV card with a single click. The company claim to be first out of the block with a distro built around KDE 3 and is promoting the open source desktop as a genuine alternative to Windows in both the home and professional markets. Another first is the inclusion of version 0.9 of Advanced Linux Sound Architecture (ALSA) for better multimedia performance. A spokesman said this was 'the most advanced Linux distribution for professional and private desktop users'.

Corporate users are well catered for with the usual array of office applications such as *KOffice* and *Samba*, while home users get a massive selection of games including *Flightgear* and *Nethack Falconseye*.

One thing that will be of benefit as broadband connections become



SuSE jumps up a version number with "easier than ever" Linux.

increasingly common is the inclusion of security services such *SuSE Personal Firewall 2*, GPG email encryption and the crypto file system.

**Mandrake 8.2** French outfit Mandrake are hoping to build on the success of their 8.x series with this latest release, which has seen some changes in the pricing structure which are causing consternation for Mandrake's loyal users. Mandrake has also had to make a few difficult requests - asking users to join a special 'club' at a cost of \$5+ per month - in the hope of staving off a financial crisis.

Mandrake 8.2 is 'the most feature-

rich, multi-purpose Linux operating system ever' and features the excellent *DiskDrake* configuration tool as well as an innovative tool called *MandrakeFirstTime* which is designed to hold the hand of new users as they venture online for the first time.

*FirstTime* configures email and registers the user with MandrakeOnline for automatic security and software updates. 8.2 is also the first Linux distribution to include the newly minted (and priced) *StarOffice 6* suite from Sun Microsystems - and this has caused a few problems.

Just days after announcing the club, and getting response from users anxious to support the company, Mandrake announced that only those paying the higher club tariff would be allowed to download the full version of the distribution, which includes *StarOffice* - leaving two-thirds of members with a limited product.

The club's plans initially said that all members would enjoy the same



Cut down on Internet traffic by installing from the LXF coverdisc.



## SCIENTIFIC TEXT EDITOR

## TeXmacs release

Somewhat overshadowed by big announcements from Mozilla and OpenOffice, TeXmacs, the WYSIWYG scientific document editor, has just achieved a 1.0 release.

Inspired by TeX and emacs, the program has a user-friendly interface, can use TeX fonts, and also supports the Guile/Scheme extension language

– so that you may customize the interface and write your own extensions to the editor. It is available for x86, PPC and Solaris. Converters currently exist for TeX/LaTeX and are under development for HTML/MathML/XML. The intention is to create a complete scientific office suite. See [www.texmacs.org/](http://www.texmacs.org/) for more.

## big three

benefits, but were later changed to save money. The company said the compromise was forced on them by Sun's pricing policy. Denis Havlik told Forum members that it was a choice between cutting revenues or

limiting the download to 'higher levels of membership'.

"After counting all the 'pros' and 'contras', and doing some polls among the users, we thought that the latter is the best solution," Havlik wrote.

## Red Hat SkipJack

Advanced Server for the enterprise sector

SkipJack is the codename for Red Hat's new beta built around the combined attractions on kernel 2.4.18, KDE 3, GNOME 1.4, Evolution and Mozilla 0.9.9.

As ever, the Red Hat hackers suggest you avoid this for mission critical systems, but if you want to download it and offer the benefit of your experiences (via the medium of bug reports) to the team, you'll find a list of mirrors at

<http://www.redhat.com/download/mirror.html>.

Red Hat have also announced the company's 'biggest step yet' into the enterprise world with the release of Advanced Server, their first product aimed specifically at the corporate sector. Based on Red Hat 7.2, Advanced Server has a kernel especially compiled for midrange deployments including failover, load balancing, clustering features and greater stability. Red Hat's

Mark de Visser said the company would be extending the development cycle of the product in order to provide a stable environment for users to test and run their applications.

The system will initially support Intel's 32-bit chips, but hope are high of bringing an IA-64 version to market in the near future, and the company have lined up an impressive range of partners, including SAP, Tivoli and IBM, who have pledged to support it.



## NEWSBYTES

■ **nVIDIA** have just published the latest Linux drivers for their full range of graphics cards, including Geforce 4 and Quadro4. Boasting support for OpenGL 1.3 and Anisotropic filtering and MPEG acceleration, get the drivers from [www.nvidia.com](http://www.nvidia.com)

■ **Linux Motor** has announced a range of support options for users of any distribution, including Red Hat, Debian and FreeBSD. 60 days of email support will set you back \$19.95, while phone support is an additional \$10. The company also offer support for an array of software solutions. [www.linuxmotor.com](http://www.linuxmotor.com)

■ Wall Street players **Merrill Lynch** are following a trend and implementing Linux in their IT infrastructure in order to save licensing and support costs. The difference here is that ML are going the whole hog and opting for a company- rather than departmental-wide strategy. Red Hat will provide support. Adopters are finding that, in addition to cost savings, Linux is providing more flexibility when it comes to expansion.

■ **Loki** staffers who had shown resilience over, above and beyond the call of duty as the troubled games company lurched from crisis to crisis, are facing more difficulties as the collapse unfolds. Bankruptcy procedures have revealed that staff stopped receiving their salaries back in 2000 and, in one case, one hapless employee was owed \$350,000 in unpaid salary and expenses incurred on a personal credit card. Among Loki's creditors were SuSE, Activision, TheCompany and VA Linux. Scott Draeker failed to turn up to the bankruptcy hearing, which triggers an automatic four-week continuance.

■ **Microsoft** is suggesting their next big OS release will be based on a completely new database-like file system, causing trouble for developers trying to compete with the Redmond behemoth on the application front. The prospect of Office file formats being tied to the Windows operating system will have antitrust lawyers rubbing their hands with glee. Still, they've got to get through the first trial first, and Microsoft's antitrust odyssey appears to be never ending. The latest revelations suggest that Dell was pressurised into dropping support for Linux on the desktop.

■ **Shine Micro** have release the SDK for the SM2496 Handspring Visor DSP module – including full source code – under the GPL. The modem code for the device is based on Thomas Sailer's Linux kernel soundmodem. "With a sufficiently motivated programmer the SM2496-16 (160MHz DSP with a MMC card slot) could be made to play and possibly record using the Ogg Vorbis format," said developer Brian Lane.

## 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

## Hooligans – time to stop

“Ever try to get help when you're new to Linux? You're most likely to be told to RTFM, or “Read The F\*\*\*ing Manual” – ignoring, of course, the abysmal state of Linux documentation.

Case in point: I was curious as to why a series of two dashes appeared before my sig when I used KMail and, sure enough, an answer was in the KMail FAQ accessible from the Help menu: 6.11 My signature has two dashes above it. This makes me very sad.

Separating the signature from the message body with two dashes and a space is a common and useful kind of standard. If you have strong feelings about this I suggest you modify the KMail source to remove that line.

And this answer made me very sad. The phrasing of the question ridicules the questioner. And it is a convention, not a standard; being found in no RFC, just a comment on changes to the RFC. The last part of the answer given to a newbie tells them they are out of luck since it's likely that they have not yet acquired the skills to modify any source code.

The attitude behind this FAQ response is a part of Linux culture whose time has past. Codified by a Richard Stallman essay on how to ask questions, it is an arrogance on the part of those insiders who have fought the gritty battle to learn all things Linux; it's an artifact of the hacker culture – bullying and ridicule are equated to “tough love,” and being a smartass to “clueless newbies” is a sport whose pursuit is highly prized.

I have a message for Dick and his buddies: Times have changed. Grow up. Lose the hooligan attitude. I have submitted a change to the KMail FAQ that provides a respectful, accurate and informative answer. I have vowed never again to bully eager new learners with RTFM.

## MIGRATION MADE PAINLESS

# IBM guide to porting

**N**ever one to miss an opportunity to promote its own wares while, of course, deriding others, IBM have created a special site for developers who want to port their applications to Linux from Sun's Solaris OS

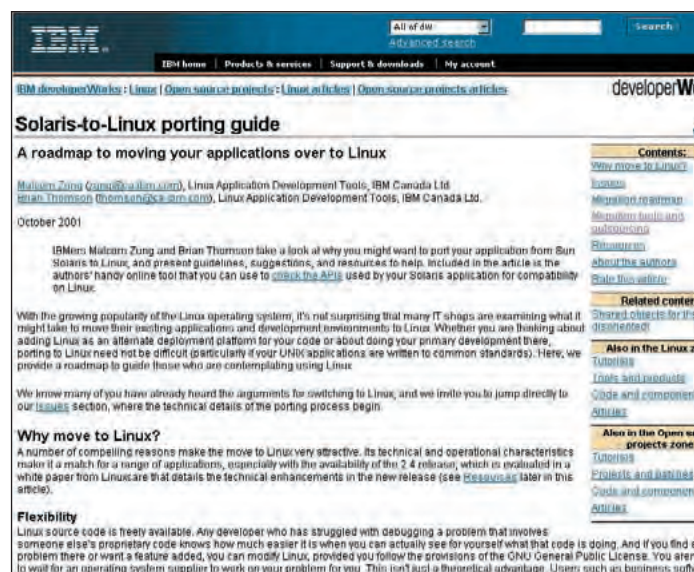
The guide, written by Malcolm Zung and Brian Thompson of IBM's Linux Application Development Tools team, contains a wealth of tips, guidelines and resources. "With the growing popularity of the Linux operating system," the pair write, "it's not surprising that many IT shops are examining what it might take to move their existing applications and development environments to Linux."

As well as providing practical advice on the subject of porting applications, the site also has a primer on Linux (including details of some of the OS's corporate successes) and a series of

well-rehearsed arguments about why you may want to upgrade to Linux. However, the real nitty-gritty is in the 'issues' section that poses the questions developers should ask before beginning their ports, and then provides some of the answers. Here you'll also find a very useful migration roadmap providing a little handholding through every stage of the process.

Though the site may (or may not) have been created for political purposes, it does contain a lot of useful information on moving from proprietary operating systems (which could include AIX) to open source. It also leads nicely onto Red Hat's more fully featured guide.

<http://www-106.ibm.com/developerworks/library/l-solar/index.html>  
[redhat.com/devnet/whitepapers/solaris\\_port/book1.html](http://redhat.com/devnet/whitepapers/solaris_port/book1.html)



## IBM urge users to ditch Solaris for Linux.

# LinuxWebWatch



[www.kevinwarwick.org](http://www.kevinwarwick.org)



[www.stelarc.va.com.au](http://www.stelarc.va.com.au)



[www.robodex.org/e/index.html](http://www.robodex.org/e/index.html)



[www.eecg.toronto.edu/~mann/](http://www.eecg.toronto.edu/~mann/)

# Hi, Robot

Hey, are you tired of your ordinary body? Boob augmentation too 20<sup>th</sup> Century for you? Why not join the ranks of 'forward thinkers' using their bodies as cyber laboratories.

The premier exponent of British cyber augmentation is Kevin Warwick, Professor of Cybernetics at Reading University. Warwick has been derided by some sections of press for being a bit, well, mad, but he claims his experiments point to the future of human/computer interaction. Being such a digital legend, Warwick has an official presence ([www.kevinwarwick.org](http://www.kevinwarwick.org)) and also a number of unofficial sites devoted to him. The best of the latter is at

<http://www.kevinwarwick.org.uk/> which seems to be linked to [www.ntk.net](http://www.ntk.net). The official site has some quite gruesome pictures of the cyborg's adventures in surgery-land.

Mr Warwick is a rank amateur next to the genius that is STELARC. This Australian performance artist has a *penchant* for radical techno-nutterdom (and a shocking laugh). Check out <http://www.stelarc.va.com.au/> for detail of his latest wheeze: ping body. This 'performance' allowed operators in

Paris, Helsinki and Amsterdam to stimulate his nerves via the Internet. He has a thing for meat hooks too!

A more traditional robot is on offer from Sony in the form of Aibo's successor the SDR-4X. This biped 'entertainment robot' was unveiled at this year's Robodex conference in Yokohama (<http://www.robodex.org/e/index.html>). The SDR-4X features a pair of 64bit processors, can sing a wide range of songs and can even recognise faces. Start saving though,

**Sony says the device will cost about the same as a luxury car.**

Professor Steve Mann (<http://www.eecg.toronto.edu/~mann/>) is Canada's answer to Kevin Warwick and has been 'living as a cyborg' for the past 20 years. His WearComp project is exhaustively documented at <http://wearcam.org/>. Mann was recently strip searched at Newfoundland airport, a process that, he says, caused \$56,000 worth of damage to his augmentations.



## NEWSBYTES

■ **Lycoris** (formerly Redmond Linux) have just started shipping a selection of Hewlett Packard desktop and laptop PCs preinstalled with their Lycoris/LX distribution. Marketing Director Jason Spisak said there had been quite a bit of interest in the systems.

■ **Psi-Domain** have launched a new website to bring together a range of high performance, open source clustering solutions. Psi-Domain says their systems represent 'a synergy of rock-solid hardware and leading software technologies'. See [www.openclustering.com](http://www.openclustering.com)

■ Seeking a new challenge, **Thomas Capricelli** set about designing a virtual architecture simply to port Linux to it. Capricelli says his project Zeta (<http://orzel.freehackers.org/zeta/index.html>) is the culmination of his fascination with low-level programming. "Porting Linux to a new platform has always been one of my dreams," he says. His current project is getting *Konqueror* and *TrollTech's Qtopia* working.

■ **Massive asteroids** passing close to Earth may not be an altogether rare occurrence, but the last one, on March 8, was only noticed three days after it had passed us by.

■ Three former **Linuxcare** founders are taking their entrepreneurial flair into the realm of public wireless networking with their latest venture, Sputnik. David L. Sifry, Arthur Tyde and David LaDuke have been sitting on their pet project for the last 12 months, but only announced their plans in March. The aim of the project is to wire(less) up public spaces so that anyone with a 802.11b equipped PC or phone could access the Internet through the ether. [www.sputnik.com](http://www.sputnik.com)



## AMERICA ONLINE-UX

# AOL makes Linux overtures

The Mozilla project has received a vote of confidence from Netscape's parent company AOL. The media giant has begun beta testing a version of its flagship client software based on the Gecko rendering engine, rather than Internet Explorer.

AOL first dipped its toe in the Gecko waters with the last beta release of the CompuServe software, which was

previously based on MS's browser core, and user responses to the change have been 'very positive'.

One thing that doesn't seem imminent is an AOL client for Linux. A source said this was unlikely due to support issues, but a switch by the world's biggest ISP to a non-MS browser will have an impact on all users of other software as webmaster's

rush to correct their code to comply with W3C, rather than proprietary standards. 30 million Mozilla users would be a powerful lobby group!

AOL is also following a number of large enterprises by switching the company's internal computing systems over to Linux. "Don't tell our competitors, let them keep buying expensive crap," a spokesman said.



Open Forum Europe will press the benefits of OSS to British business.

## BUSINESS INITIATIVE

# Europe gets new OSS advocate

Open Forum Europe is a new organisation set up to promote the adoption of open source software in the European business sector. Launching the initiative, the group said it 'aims to strengthen the perception and credibility of Open Source, particularly within Government and commercial organisations.'

Geoff Morris, Chairman of IT Forum Foundation, who are backing the initiative said: "Open Source software is increasingly featuring on the agenda of board level directors, in particular CIOs. Our research has highlighted the fact that the commercial aspects of OSS are of at least equal importance to technical issues, but that they are currently not being addressed."

The problems were highlighted during a three month consultation project undertaken with the Dept of Trade and Industry. It concluded that there is a need for 'credible, comparative business models' for open source software and 'commercial and technical transparency' and the fact that 'topical benefits such as improved security and lower cost of ownership are not well articulated or proven.'

The consultation also revealed the differences in governmental support for open source software, with Britain 'lagging behind' other parts of Europe.

Bruce Forbes of the Financial Times Group said his company was very pleased to be involved in the project. "The launch of OpenForum Europe is an invaluable opportunity to help UK business grasp some of the real business opportunities surrounding Open Source Software," he said.

Open Forum Europe also announced a partnership with the National Computing Centre (NCC) to help UK businesses adopt and exploit opportunities arising from Open technologies and OSS

Michael Gough, of the NCC, said businesses needed more information on the risks and benefits of open source software. "There is currently a degree of uncertainty in the market over what Open Source actually is," he said. "The NCC believes that the UK would benefit greatly from the provision of independent support and guidance advice on the availability and suitability of open alternatives."

[www.openforumeurope.org](http://www.openforumeurope.org)

## Jono Bacon

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



## COMMENT

# Freedom and equality

“ Although I have a technical and social interest in Linux, it never ceases to amaze me on even the most primal of levels. This month I have been doing that pondering thing again, and my pondering has focused on equality and how Free Software enables it.

Watch the news and you get the impression that the world hates itself. There are few fields of study where people are deemed truly equal – Free Software is one of them. The notion that every person who contributes to the Free Software movement is equal and valid empowers people to improve and extend themselves. Those of you who are regulars of *Linux Format*, or have used Linux for a while, may take this freedom for granted, but it really is deep rooted within the general opinions and ethics of the community.

I was once quite critical of Stallman for his views and I felt he was often pedantic and extreme in his methods; recently though I have been particularly sensitive to political climates (material for my songwriting) and I have really reassessed his message to the world.

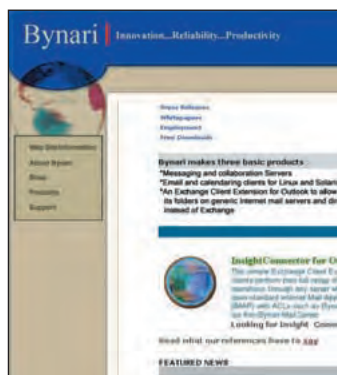
One thing that we need in life is freedom, and people who fight for freedom are true heroes. We all know that Linux is a cool OS, it is stable, it is supported and it has a great community that goes with it, but the true social implications of it are something that has taken me the best part of four years to fully digest.

There is no doubt that this freedom has distinct implications in terms of businesses making money. The question that I ask, however, is how close can the line between riches and rags be drawn when the playing field is all about openness and freedom. Will closing freedom to make money suffer the same fate on a free OS as opening up freedom had on a closed OS?

## LINUX DESKTOP AND MS EXCHANGE

## Getting connected

**M**arch saw the official release of *Connector* for Microsoft *Exchange* from Ximian, the software that allows users of Ximian's *Evolution* groupware suite to share information and collaborate with their Windows-based colleagues. With *Connector* installed, *Evolution* operates as an *Exchange* 2000 client providing access to email, personal calendars, group schedules, address books and tasks lists using existing *Exchange* 2000 servers.



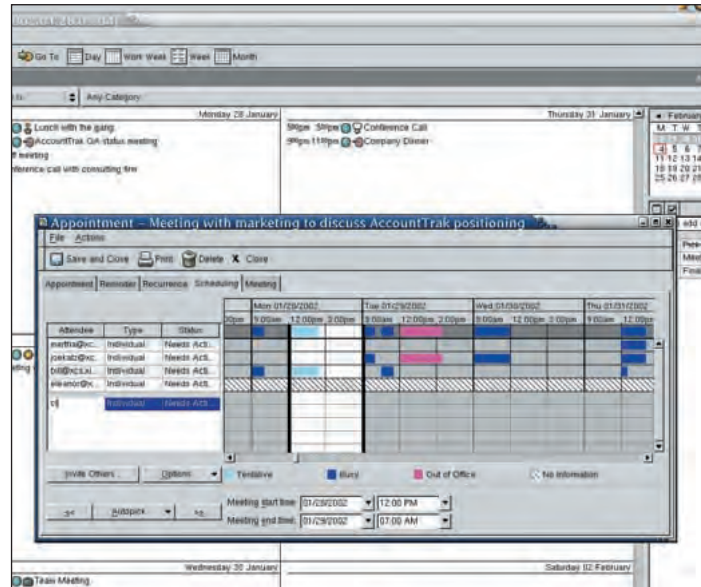
Bynari's *InsightClient* allows users to drop *Exchange* completely. It's cheaper too.

Robert Mahowald, senior analyst at IDC, said integrating Linux workgroups into larger Windows-based enterprises has been a barrier for take up in the corporate world. "Ximian *Connector* for Microsoft *Exchange* is designed to break down those barriers," he said.

Ximian's David Patrick said the company had received an overwhelmingly positive response from customers using the system. "By providing this key missing ingredient to interoperability with office document and messaging standards, Ximian is helping companies support their Linux users without compromise while eliminating the costs normally associated with the purchase of second PCs or mail systems," he said.

*Connector* integrates with Red Hat, Mandrake, SuSE and Debian and can be purchased from Ximian.com, priced \$69 for a single user license (\$599 for 10 pack, \$1,449 for 25 pack).

Bynari.net have a competing product on the market (also proprietary, but cheaper) with the stated aim to 'unplug Enterprises from Microsoft *Exchange*'. *InsightClient* 2.9 for Linux and Solaris lets non-Windows users work with *Outlook* through the



*Connector* can hook up *Evolution* to an MS *Exchange* 2000 server.

server. The application hooks into Bynari's *Exchange* Client Extension, which lets *Outlook* users move from *Exchange* to IMAP servers using while retaining *Outlook*'s native components. *InsightClient* also offers server side calendaring, meeting management

and user level shared personal folders on Linux and Unix desktops – it can share its calendars with all versions of *Outlook* in either an *Exchange* network or on Bynari's *InsightServer* 3.0.

[www.ximian.com/products/connector/](http://www.ximian.com/products/connector/)

## Embedded Linux News

● Empower Technologies have launched their 'Palm-like' PowerPlay V which is built around a 16MHz Motorola Dragonball system-on-chip processor. It has 8MB of RAM and is available for \$149 from the company's website.

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

● Sharp's consumer version of the Zaurus is now available. The launch was marred slightly when the online store informed user of *Opera* and *Mozilla* that they needed to 'upgrade' to *Internet Explorer* or *Netscape 4.x* in order to enjoy the site. Hang on; doesn't the Zaurus use *Opera*? <http://sharpelectronics.com/products/FunctionWhereToBuy/1,1182,29,00.html>

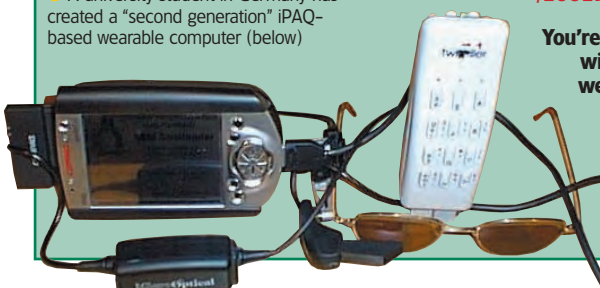
● A university student in Germany has created a "second generation" iPAQ-based wearable computer (below)

running Linux. The device, which dual boots PocketPC and Linux, features a Twiddler chording keyboard, a MicroOptical head mounted display and a text-to-speech audio interface.

[http://www.iptel-nov.de/PROJECTS/WEARABLE/2G\\_IPAQ\\_WEARABLE/2g\\_ipaq\\_wearable.html](http://www.iptel-nov.de/PROJECTS/WEARABLE/2G_IPAQ_WEARABLE/2g_ipaq_wearable.html)

● IHL Consulting have published research which suggests that Retail Point of Sale Terminals running Linux in North America has increased 80% over the last year. The bad news is that the company had expected growth to be in the region of 200-400%, but the figures were hit by "two large retail defections from planned rollouts of POS units." [www.ihlservices.com/en/2002Press4.asp](http://www.ihlservices.com/en/2002Press4.asp)

You're never alone with a talking, wearable iPAQ



## RELATIONAL DATABASE

## Firebird taking off

**The Firebird Project** has announced the first official release of its open source relational database for Linux, Windows and OS X. The software is based around the version of *InterBase* 6.0 that Borland released as open source (under a variant of the Mozilla Public Licence) in August 2000, but because Borland quickly retreated back into closed source development, *Firebird* has become the definitive open source version of the application. This is a fully relational database offering many ANSI SQL-92 features, excellent concurrency, high performance, and powerful language support for stored procedures and triggers. It is almost 100% compatible with *InterBase* 6.0.

Though the project, which has an active development team of 60, is now independent of *InterBase*, it is still essentially the original application with 'a lot of bug fixes'. However, the team don't only squash bugs, as "*Firebird*



*Firebird* is a fully featured, open source relational database.

developers are mostly also *Firebird* everyday users, they are focused on features and improvements that really 'scratch an itch', rather than features for features like in many commercial products," the project leaders said.

<http://firebird.sourceforge.net/>



## RESPONSIVE OFFICE APPS

## Moving Offices

**Currently available for Windows** and 'imminent' for Linux, Gobe Software's *Gobe Productive* is a productivity suite with a full range of office apps, Office file compatibility and a unique approach to licensing.

*Productive*, which was previously available for the now defunct BeOS, has been created by the developers responsible for the *ClarisWorks* (now *AppleWorks*) suite and features word processing, spreadsheet, graphic, illustration and presentation apps.

Bruce Q Hammond, the company's CEO, said the suite's strongest point was integration. "It brings such a wide variety of powerful tools to average computer users and does this in the one application that is used for everyday tasks. The user doesn't have to buy – and master – several professional apps to get access to these tools."

Gobe's product really breaks the mould in its licensing policy. Instead of restricting users to installing the software on one machine, the company is pioneering what it calls

the 'family license' – buyers can install the software on each of their home PCs, as well as having a 'take to work' copy to use in the office.

The Linux edition is scheduled for mid-2002 release.

## OpenOffice frozen

As we approach deadline, the project leaders at [OpenOffice.org](http://OpenOffice.org) have released build 641d and said this will be the last offering before 1.0. They have also reiterated their commitment to open source and Free Software in light of Sun's decision to put a price tag on *StarOffice 6*.

However, before the official product can be released, the project is asking for users' help in nailing down the final bugs.

[www.openoffice.org](http://www.openoffice.org)



## GNU, AND JUST GNU, TO BE RELEASED

## I Hurd GNUs today, oh boy!

**Richard Stallman used a GNU/Linux** conference in Bangalore, India to set out his vision of a GNU future, and claimed a primary release of his Hurd kernel would be available by the end of the year. "We have the GNU kernel working, and now we can produce a GNU system, as opposed to a GNU/Linux system," he said.

Hurd was Stallman's microkernel dream when he founded the GNU project in 1984, but he soon found it upstaged by the emerging Linux kernel, which was adopted as the core of the open source OS. "Linux is a kernel, and now we have our kernel, which is an alternative to Linux, and they both work in the context of the

overall GNU system," he said.

The release would also squash a bug that has been playing on Stallman's mind for the last ten years: the use of the term 'Linux' to describe the entire OS. "It is actually GNU/Linux with Linux as the kernel. It is really devastating for us when people write about our work and they don't call it by our name, and we get forgotten."

More seriously, Stallman claims Hurd's microkernel architecture makes the OS more powerful than the 'monolithic' Linux. "It has an open-ended collection of server programs that together do the high-level jobs of a kernel like the Unix kernel or Linux." [www.gnu.org/software/hurd/](http://www.gnu.org/software/hurd/)



Get ready for GNU. The Free Software Foundation's Unix replacement nears completion.

## Stallman on victimised developers

At the behest of the Foundation for Information Policy Research, Richard Stallman was in Cambridge recently, giving a presentation on software patents (SP) to the University's Computer Lab. He began by equating SPs to a lottery that rarely brings benefits to people. "Lotteries," he said, "invite you to think about winning, never about losing, and it is the same with the patenting system."

Suggesting ways that patents can stifle creativity at every stage, Stallman said "The first thing you do after you have an idea for what sort of software you want to write, is to find out what patents may cover the area. This is impossible, because some patent applications pending are secret. They may get published after 18 months, but that is plenty of time for you to write

your own program and publish it"

This process, he says, leads inexorably to trouble, citing the LZW case in which the patent was applied for in 1984 but not granted until 1985 – by which time Compress (a competing product) was available and distributors were receiving threats from the LZW developer.

A section of his presentation dealt with three methods developers could use to avoid the patent trap: avoidance, licensing and litigation. These all result in some loss of freedom for developers

The real solution, Stallman says, is to treat software like a symphony. "You use many notes and many ideas that have been used before," he says, but the end result is something new.

## David Cartwright

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



## COMMENT

## Second's not so bad

**I extol the virtues of Linux** as a server OS, but not on the desktop. It's true that my own HP notebook dual-boots between Windows 2000 and Red Hat, but every time I sit there bashing code into vi I contemplate whether I'd move my department, or my current client, to a Linux desktop.

The arguments in favour of Linux are growing in number.

Decent GUIs are now available to all, not just those brave enough to fight with *Xconfigurator*. And with good guys like Sun doing neat things like producing full-featured office software suites, the amount of software available for Linux desktop users is getting greater and greater.

The problem is that MS has been very clever by producing vast amounts of integrated, supported, easy-to-use APIs, and some really neat development tools (in the form of *Visual Studio*) to use them. Linux can't currently come close for desktop software development – as a die-hard Linux/C developer I just couldn't believe that I wrote a Visual Basic program that did GUI stuff, ODBC to the Oracle database and file I/O in a single morning ... having never touched VB before.

I have always had huge respect for the whole Open Source/GNU movement under Linux, and I still find it astounding that vast, complicated products like GCC are out there, and they're free. The problem is, though, that Microsoft simply has the size and the money to go out and probably not do it better, but do it in a way that's easier for the average user to get to grips with.

So let's keep fighting Linux's corner. But don't be frightened to believe that MS will always be more popular. Being number one is, of course, everyone's aim; but there's no shame to being a bloody good, highly-respected number two.

# Making Linux accessible

**Robert Rice** reports on the successes of Sun Microsystem's Accessibility Team at a technology conference for people with disabilities, and discusses the social responsibility of developers, end-users and corporate management to ensure accessibility in technology.

**Earl Johnson, Program Manager and founder of Sun's Accessibility Program Office demonstrates GNOME's accessible on-screen keyboard using a head pointer in lieu of a mouse.**



Over 4200 attendees and 156 exhibiting companies convened at CSUN's (California State University, Northridge) annual Technology and People with Disabilities conference in March to discuss, announce and share innovative ideas and emerging technologies that aim to improve the lives of people with disabilities.

Making websites usable and accessible to the visually impaired, using online videoconferencing as a means of communication for the deaf and employing the use of new

products and services to improve technological access were only a few of the issues discussed.

## What's accessibility?

In the world of electronic and information technology, "accessibility" refers to the possibility for everyone, including people with disabilities, to access and use technology and information.

The World Health Organization estimates that 750 million people worldwide have a significant disability that prevents them from a major life activity. For many, this life activity is

the use of computer technology.

Many people now use a new generation of assistive technologies and applications to improve access. People with visual disabilities can use screen readers which present audio or braille output of screen GUI content or screen magnifiers that enlarge screen content. Those with physical disabilities can interact more easily using hands-free mice or on-screen keyboards.

The deaf or hard-of-hearing may not hear a computer's bells or whistles. Because of the increase in the use of sound, video, plug-ins and multimedia on the web, content is increasingly become inaccessible to those with hearing disabilities. Online news or educational videos are rarely captioned and scripts are usually not available to accompany audio.

Those who are blind, or have low vision, experience difficulty with the informational use and manipulation of web-based content. Many online images do not have "alt" tags that typically are descriptive of the image itself. Additionally, more complex images, such as charts or graphs, need to be detailed elsewhere on the website in addition to having the alt tag. Web developers often create website designs that use colours that are not distinguishable to individuals with colour-blindness.

## Conference notes

The interaction of assistive technologies with computing environments was a conference highlight of the *In Your Own Words* exhibit, a project collaboration between CSUN and Microsoft. Computer users with varying disabilities were brought in to demonstrate how they were able to improve their professional and





Mike Jeffers demonstrates the use of a just-point technology produced by Avid Technologies. This technology allows the use of the finger to replace the mouse.

### Three quarters of a billion Needing accessibility solutions

There are 750 million people in the world who have disabilities. One family out of three has a member with a visual, hearing, mobility, speech or cognitive disability. 17 million Americans use technology to assist them in their daily routine. One out of five non-disabled young men will acquire a disability before they turn 40 years of age.

personal lives by leveraging a new generation of assistive technologies.

The conference's keynote address, delivered by Gregg Vanderheiden of the Trace Research & Development Center, a part of the College of Engineering, University of Wisconsin-Madison, focused on the issue of universal design and the need to have corporations become more responsible for the implementation of its principle. Universal design refers to the design of products or environments to be usable by all people to the greatest extent possible without the need for product modification. The universal design principle can be found in the kerb cuts that exist at the end of

pavements that allow wheelchair users, bikers or travellers with luggage to be more mobile. Another universal design example is the use of instant messenger programs that allow both hearing and deaf computer users to easily communicate with others.

In an effort to improve accessibility for computer users with disabilities, private and public sector organizations including Sun Microsystems are implementing innovative accessibility initiatives and product development guidelines.

### GNOME Accessibility

The increasing popularity of Sun's GNOME desktop is bringing about the need to develop accessible desktop

software for Unix and Linux platforms enabling companies to provide functional solutions for their employees with disabilities. In addition, the U.S. government and its agencies require its suppliers to provide accessible technology, as stated in Section 508 of the Rehabilitation Act.

Sun's GNOME Accessibility Framework is based on an open architecture, and provides a comprehensive toolkit and APIs that developers can use to create accessible applications, and interface them with assistive technologies on any Unix or GNU/Linux platform running the GNOME 2.0 desktop. With the combination of the Accessibility Framework and



**“Working with mature platforms means accessibility comes late to the table and is never really part of the underlying design. On the other hand, working with young platforms means making a gamble. Will the platform be important enough in the future for accessibility to be considered?”**

Peter Korn, Sun Microsystems



Martha Prine demonstrates Cyberlink, from Actuated Technologies. This enables the wearer to direct a computer mouse with a headband.



Devon Ferguson, a teenage quadriplegic, and his sister Candice, using head pointer technology in lieu of a mouse or joystick for video games.

# Accessibility

## Useful Links

Freeing software for everyone

Developer information for application developers, interface designers and web content developers

<http://www.sun.com/access/developers/index.html>

GNOME developer information (Sun are currently funding a number of engineers on accessibility work.)

<http://developer.gnome.org>

IBM guidelines for writing accessible applications using 100% Pure Java

<http://www-3.ibm.com/able/snsjavag.html>

IBM Java accessibility checklist

<http://www-3.ibm.com/able/accessjava.html>

Trace Center at the College of Engineering, University of Wisconsin-Madison

<http://trace.wisc.edu>

CSUN Center on Disabilities

<http://www.csun.edu/cod>

Compaq's Accessibility video

<ftp://ftp.compaq.com/pub/corporate/accessibilityrev.wmv>

**“It doesn’t take a visionary to know building accessibility into the desktop is the right thing to do for your business environment. Making accessibility a reality provides users with disabilities a higher quality computing experience. It levels the playing field for your most important commodity – your employees.”**

Scott McNealy, Chairman and CEO Sun Microsystems, Inc.

## About the author

Robert Rice is President of Rice Consulting LLC, a management and technology consulting firm specializing in technological accessibility. He can be reached at [RiceConsulting@aol.com](mailto:RiceConsulting@aol.com)

◀ assistive technologies, users with disabilities have the tools they need to efficiently use computer technology.

Believing in the need to evangelize accessibility in products at the beginning or peak of the development life cycle, Sun's Peter Korn, an Accessibility Architect, states, “Working with mature platforms means accessibility comes late to the table and is never really part of the underlying design. On the other hand, working with young platforms means making a gamble. Will the platform be important enough in the future for accessibility to be considered?”

GNOME 2.0 also provides users with alternative ways to enter text and manipulate the GUI of applications and the GNOME desktop.

“By providing rich and consistent interfaces for both assistive technologies and applications, the accessibility framework enables flexible and comprehensive accessibility solutions for GNOME,” said Bill Haneman, GNOME Accessibility Architect, Sun Microsystems, Inc. “A key aspect of the GNOME Accessibility Framework is support for user interface component sets beyond the built-in GNOME component set,” said Peter Korn.

## JAVA Accessibility

Java application developers can take advantage of a relatively easy way to incorporate accessibility into their designs by taking Java Foundation Classes, also known as Project Swing, and its implementation of accessibility methods in the user interface components.

There are two accessibility programming packages of importance:



**Ms Prine demonstrates the Datahand ergonomic keyboard. With four modes, shifted by the thumbs, hand movement is no longer required to perform keyboard work. A boon for repetitive strain injury sufferers.**

The Java Accessibility APIs and the JAVA Accessibility Utility APIs.

The Accessibility APIs, which are contained in the `javax.accessibility` package, define application-side interfaces and classes that make components “accessibility friendly.” They provide a wide array of detailed information about GUI components. The Accessibility Utility APIs rely on this information to implement the functions of assistive technologies.

The Accessibility Utility APIs are designed for developers of assistive technologies. They offer support for locating and querying user interface objects, tracking events, installing event listeners, and loading the assistive technology into the virtual machine. JavaSoft bundles the Accessibility APIs with the Project Swing libraries. The Accessibility Utility APIs are downloadable independently.

## Taking Responsibility

As the world's population ages, it will become more prone to disability. One

out of five non-disabled young men will acquire a disability by the time they reach 40 years of age.

Consequently, making technology accessible to everyone becomes a greater priority.

Developers need to take advantage of frameworks and tools similar to Sun's GNOME Framework allowing them to efficiently create accessible applications and interface assistive technologies with a system usable by everyone regardless of disability.

Disabled end users need to begin collaborating with colleagues, retrieving and manipulating information, and communicating with loved ones electronically, alongside their peers. And finally, upper management and executives need to embrace and evangelize accessibility and empower employees with disabilities, while also providing new opportunities and capabilities for entire workforces. **LXF**





# Mailserver

Share your opinions, right wrongs and demand justice by writing in 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)

## DMCA

I noticed in your latest issue of *Linux Format* there was a letter regarding the Digital Millennium Copyright Act (DMCA) where someone reported having contacted their MP about this issue, and was encouraging other people to do the same.

I have also written to my MP about it, and I am in fact part of the UK Campaign for Digital Rights, who are working to affect the governments policy on issues like the DMCA (or the European Copyright Directive, which is the EU equivalent). I would encourage anyone who is interested (and this is a subject that affects us all, particularly with the new CBDTPA laws being proposed in the US at the moment) to visit our website <http://uk.eurorights.org/>

If you can get to London or Edinburgh, we are running seminars there shortly, to give people more information about the issues at stake, and what we can do. **Matthew Johnson**, *via email*  
Thanks for the info, prepare your servers for much traffic!

## Familiarly Familiar

*LXF 26* just dropped through my letterbox – I was pleased to see the letter from Jonathan Chetwynd concerning Linux installations on the iPAQ.

Familiar and intimate are two of the Linux distros designed for PDAs

**“Fortunately it seems as if the ‘new school’ of Linux programmers seems to be waking up to this and newer applications feel more intuitive.”**



**Software patents and other restrictive legislation may make Free Software illegal – but not if the Campaign for Digital Rights can help it.**

– Familiar being supported, I believe, by Compaq (<http://familiar.handhelds.org>)

I have installed Familiar on my iPAQ 3630 – it's easily done. However installing the packages, with which to make it usable has been a bit problematic! The *ipkg* system is supposed to retrieve the appropriate packages from [familiar.handhelds.org](http://familiar.handhelds.org), but I just cannot get the IP forwarding to work through my Mandrake 8.1 box. Don't know why!

I would certainly welcome an article on the subject. **R.D.Saunders**, *via email*  
Thanks for the enlightenment. Now we know what's going on, we may be able to sort something out!

## While ROM burns

Like C. Anne Wilson in “No Boot” (*LXF25* page 21), I too have tried to burn installation CD iso's using *Nero 5* under Win98. It can work, but I have no idea how! I managed to rig it to work once, but every succeeding attempt has failed. Luckily I have another spare PC that I use for experiments and that has Mandrake 8.1 on it. Burning those iso's from the command line using *cdrecord* is very easy, but, of course, you have to have Linux installed in the first place.

One of the distros I tried recently was Redmond Linux from a recent *LXF* coverdisc. I do agree with the review that it is an excellent introduction to Linux. I would almost be using it now were it not that the old and battered hard disc I installed it on finally seized up for the very last time after only a couple of days of playing with it.

I think that one of the ruts that Linux dug itself into is finally going away. That is the all encompassing

hatred of Microsoft. While I agree that there is much to criticise Microsoft for – dubious business ethics and flaky software – I also think that they do deserve some praise for at least one thing, and that is for the design of the human interface. Windows software seems to be given easily remembered and relevant names, and the usage of most programs is very intuitive (at least for basic purposes).

Fortunately it seems as if the “new school” of Linux programmers seem to be waking up to this and newer applications feel more intuitive than some older ones. After all, in this glorious multimedia world, using some applications is hard enough without the complexity of knowing “what is going on under the bonnet”.

All that may give the impression that I am somewhat of a newbie to Linux. In some ways yes and in others no. I have actually been dabbling with Linux, off and on, for a couple of years now. I guess I understand maybe 10% of



**Redmond – learning GUI tricks from Xerox PARC, via Mac and Windows.**



## ★ Letter of the month

This month's winner receives a copy of **Mandrake Gaming – The Sims**

### No burning required

I've been reading your forums anonymously since I am reluctant to register for anything anywhere, and so I must send you email with this suggestion. Please do not add me to any lists.

A lot of your readers are complaining about burning CDs with the distros from your DVD. There may be an alternative if you have a home network or are installing a distro that allows installation from the hard disk, like Mandrake or Red Hat.

For a hard disk install:

- 1) copy the iso's to the hard disk
- 2) create a /home/distro directory
- 3) **mount -o loop -t iso9660 /path/to/iso/disk1 /mnt/iso** (assuming /mnt/iso exists)
- 4) **cp -r /mnt/iso /home/distro**
- 5) **umount /mnt/iso**
- 6) repeat steps 3 to 5 for each disk image
- 7) make a boot floppy with the hard disk image from the distro, usually **hd.img**
- 8) reboot from the floppy, then point the installer at the correct

partition with the path /home/distro.

The installer should be smart enough to mount the partition and find the distro. Of course you will be in trouble if you previously or subsequently chose to zap that particular partition.

If the distro cannot install from the hard drive but has a net installer, follow the same procedure but export /home/distro via NFS, FTP, or HTTP on box A and use the network boot floppy image to install on box B. This would also allow you to blow away all partitions.

I have yet to do this myself (although I plan to tonight) but others have done so, usually with downloaded iso's. I wonder if you could do something similar by using the hard disk installation boot floppy and pointing the installer directly at the path to the distro on the DVD, since some installers (Mandrake's) will work



directly with ISO images. You may want to test it yourself.

Michael Walmer, via email

Thanks for this helpful guide. Yes, it is often possible to find other ways to install distros, including HD and network installs that don't require you to burn disks.

The only problem is that this is a little more complicated, and often distro specific. Given the number of queries we get about burning ISO images to CDs, we haven't really been too keen to promote this way of installing! However, it is possible as you point out, but don't anyone come running to us if it doesn't work for you...

To thank you properly, we'll be sending you a copy of Mandrake Gaming Edition, complete with *The Sims* for Linux!

As someone used to building an NT Server in the Office in an hour and it working first time this was mildly frustrating. Analysing the scripts reminded me of those days when magazines published pages of BASIC listings for the ZX81 in the early eighties only to find a printing error prevents the program from running. Suffice to say I never did find the script error and at 3am re-installed *sendmail*.

For me it was a bit of fun that didn't work. But if I were a business user the language of tarballs, compiling and writing scripts would be frustrating. Compare that with typing **SETUP** in Windows and just accepting two or three default install questions. Until Linux gets a common directory format, a common install/deinstall format, and most of all, a common binary format it will be long time before it moves to the business desktop.

Steve Townsley, via email

Well, here's another question for you – why on earth did you want to install *qmail* anyway? There are plenty of commercial mailservers for Linux that come with installers, and set up fine – *GLMail* from Gordano is one example. Some Open Source software is easier to use and install than others, but as most people use *sendmail*, and as (although it's config files are lengthy and confusing) it works well, I'm not sure what the problem is.

### Map madness

Just been looking at the map on page 108/109 of the latest *Linux Format* magazine, which shows the locations of the various Linux User Groups around the country.



what appears in *LXF* so I still regard it as a valuable source of information. It is such a vast subject that I don't think you should assume that any reader is not a newbie in some area or another.

One final plea. If you (that is the whole Linux world – not just *LXF*) want me to become a total convert to Linux, then will you please write a program to replace the most excellent sound editor – Syntrillium's *Cool Edit*. I did once attempt to run it under *Wine*, but the whole thing was a complete disaster. Perhaps I ought to dual boot, but that rather defeats the object.

Bill Kelsey, Catford, London

A number of people seem to have problems with this *Nero* software. Nobody here has ever used it, but the principle is the same as with other burning software. The ISO images are just that, images of a disc. There is usually some way to 'import' or otherwise load such images into CD-writing software to burn the disk. We

can only advise, at this point, that you read the manual for the software you are using.

Many people find the 'Windows-style interface' intuitive (though you're inviting argument for claiming that Microsoft are responsible for it!), but this is probably just because they are used to it. Mac owners feel the same way.

I've never come across *Cool Edit* (I assume not related to the Linux tool *Cooledit* by Paul Sheer, which is a text editor), but there are plenty of excellent sound editors for Linux. Check for yourself on [www.freshmeat.net](http://www.freshmeat.net)

### A big Question?

Perhaps the biggest question that lurks around the corner for Linux is will it ever grow from being an enthusiasts dream to being a real force on the nation's PCs. A recent experience I had says "no"!

I wanted to replace *sendmail* on Suse Linux 7.3 with *qmail*. I went to [www.qmail.org](http://www.qmail.org) and discovered

there was no binary install so I had to use source code. It compiled fine and then I spent about two hours installing the thing. I made directories, copied scripts, changed permission, unzipped tarballs etc. Finally it was all ready to run and I typed **qmail start** – within seconds three scripts failed.



Put yourself and your LUG on the (nearly accurate) map.





« I thought I'd mention that the '12' for the Moray group is positioned slap bang in the middle of Inverness – which is not actually in Moray. The '12' should be 15mm to the right and 5mm upwards (about half an inch and quarter of an inch in old money) which will put it slap in the middle of Elgin (I'm an Elgin exile living in Bradford) – which is in Moray.

Please take this as constructive!!

I love the magazine and have subscribed to the DVD issue once I had read the November edition; mind you, I had an ulterior motive for reading that one – you featured my website contents on the coverdisc as part of Simon Goodwin's Sinclair Emulators article that month.

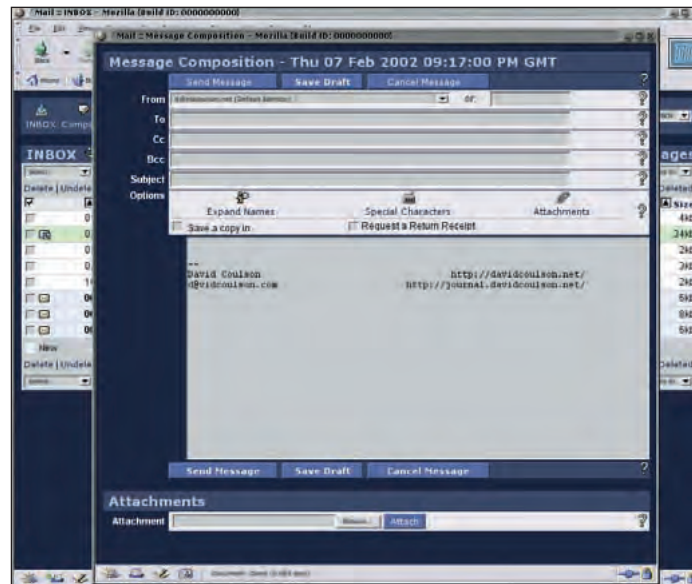
Norman Dunbar, Bradford

Thanks for the update. We do try to be precise with the little blobs, but they have a life of their own and seem to migrate all over the place. If you spot any more that are wrong, email us!

**“My scripts are hacks. Professionals would throw up their hands in horror, but it worked OK for me. So could we have more ‘quick and dirty’ scripts.”**

## Mail Serving

Following up from the letter by Christopher Astbury, if the lemmings insist on using *Outlook*, Caldera International recently released a product that supports *Outlook* clients – Caldera *Volution Messaging Server*. The base product supports 25 users, and costs less than \$1000 US. It should be run



Support your *Outlook* clients (if you must) from a stable platform, with Caldera *Volution Messaging Server*.

alone on a server and requires a late model CPU, apparently planning to power a large enterprise.

It also offers support for virus

scanning, which *Outlook* obviously screams out for. Check it out at [www.caldera.com](http://www.caldera.com)

Thanks for a great magazine and attractive website...

Tom Younker, Atlanta, GA USA

Yes, we reviewed *Volution MS* in issue 25, but thanks for pointing out this capability! And I'm glad we've finally found someone who likes the look of the website!

## Doing something

Love the mag – been getting it for almost two years now. One suggestion, I like to file a lot of the articles, and that's hard where the same leaf has maybe items on *Kylix* (which I'd like to put with the other *Kylix* articles), and the other side of the leaf has Java (which I'd like to file with Java stuff). Maybe space the ads?

Anyway, I'll be looking for a plug shortly because I believe so much in Linux that I started my own business to sell it. I got appointed as the distributor for Desktop/lx (formerly Redmond Linux) for the UK and Ireland. Next up, I'll be trying to get a dealership for laptops so I can offer them preinstalled with it. Will let you know when that happens. I'm currently putting up my website – it will be at <http://www.linuxfactory.ie/> before the end of February. I'd like to add a list of links to LUGs in the

UK and Ireland – the same one you include in the mag, maybe. Could you save me some typing and email the list to me – some feedback on the site would be nice as well, if you're feeling generous.

Mel McWeeney, Longford, Eire

What adverts? I haven't noticed many. There is simply so much editorial in this magazine that it would be difficult to arrange it so that every feature or tutorial didn't overlap with anything else. We used to plan the tutorials to start on a left for this reason, but it became impractical when we started doing tutorials of different lengths.

May I suggest you download PDFs of the tutorials when they become available on the website – you could then print out the missing page...

I wish you luck with the site, you should already have received the list of LUGs and some suggestions!

## Return of the Glue

First of all, thanks for the great magazine, you are doing an excellent job! But there is one point where some urgent and drastic change is required:

The way the CD boxes are glued to the front page of the magazine. I am not sure it is a general problem or only Holland specific, but the boxes are glued to the magazine with some plastic glue like of sticky stuff, which first of all makes it impossible to get the CD box out of the magazine cover without damaging the cover page.

Secondly even after the box was removed, it keeps sticking to everything on my desk effectively destroying any CDs and DVDs hanging around. Lastly, it makes it impossible to put these boxes next to each other (what a tidy person would do) or pile them up as I used to do. (Actually, putting them on top of each other is easy, but then

**Helpdex**  
BY SHANE COLLINGE  
[shane\\_collinge@yahoo.com](mailto:shane_collinge@yahoo.com)







they will form a solid, brick-like object, which severely diminishes their usefulness). <grin>

Sandor Laza, *Holland*

AAArgh! I thought the glue issue was well and truly buried. In the UK, the CDs are mounted on a backing card, so no glue is used, but we cannot do this for overseas versions. However, a lot of time and effort has gone into addressing the glue problem. Apparently there is a consistency problem – sometimes it isn't sticky enough, but it varies at what stage of the cycle your individual magazine has the disks stuck on. I appreciate it is annoying, but the sticky tape was worse (it either didn't stick or the whole cover was ripped off when you tried to remove the disk).

I hope this is not going to be a continuing problem for you, I'll have our glue experts look into it again.

## Parallel processing

I have just recently come into the world on open source and find it a very exciting place to be.

I wish to contribute to the community by using the power of my PC to fund Linux projects. Is there an organisation similar to the likes of the SETI project which could use my computer to number



Coming unstuck – or not. We must be using the wrong kind of horses..

crunch and use that power to be sold to companies that need lots of it, like animation or the scientific community. If it a financially viable proposition to fund full time programmers and I could choose where the funding went I would be very willing to do this.

Robert Kerr, *Edinburgh*

Well of course, you can contribute to the Seti project on Linux, but I don't know of any similar, Linux-benefiting projects. Anybody?

## Useful Scripts

Just thought I should drop you a note to say that this issue is the best yet. I've been getting the mag almost since it came out, though I decided not to subscribe because I thought that if copies didn't move off WHSmith's shelves they wouldn't give it shelfspace, and then people wouldn't see it, and then Linux would get less eyeball-space, and so on.

The early issues were just a trot around the Linux world, and then you started doing articles which went into things in a bit more depth, but seemed to me to be missing out on going the last mile, e.g. "you could do this ..." but not actually "this is how it would work in this particular setup (YMMV) ...".

But this issue really is outstanding – a good CD burning roundup, an excellent VOIP overview, a thought-provoking article on HA networking (which I had never heard of, but I now want to test), excellent backgrounders on devfs and ext3 (where did you find Biagio – can we have more articles by him?), a detailed look at partition backup, and (for me the icing on the cake), your Python script. All good solid stuff, a real step up from previous issues.

Your script may or may not be

We will try and include more scripts in future issues of LXF.



perfect, but I don't really care. It so happens that I was speaking to someone at a client's on Friday who wanted to set up online questionnaires, and I was telling him how he would have to spend a bit of time reading up on PHP and MySQL, and fit that into HTML. He agreed, but I could see him getting restless at how long this was going to take when his boss wanted them up a.s.a.p.. I can now fiddle about with this script, and give him a tool that will do at least two of the things he needed to do without putting too much of a burden on his brain.

I think most scripts are hacks – I did one two weeks ago on PHP to compare two MySQL tables, and take ones that are in table A out of table B and put them into table C. Most professionals would probably throw up their hands in horror if they saw it, but it worked OK for me. So could we have more "quick and dirty" scripts – you never know how much they might help someone else in the same predicament!

Thanks again for a great mag – I'm looking forward to next month's LTSP article; I got this working last year on

SuSE 7.2, and it's very impressive. (Drat! I should have sent you an article about it last year, instead of reading someone else's next month! Doh!)

Kevin Donnelly, *via email*

Thanks for your kind words, I'm glad you are enjoying the magazine.

I will try to include more useful scripts and bits and pieces that we use in future issues. They will probably be along the same lines – quick and dirty stuff we have used for creating CD interfaces, or managing SQL data or other things from our website. I think it is useful to give people access to this stuff, that's what open source is all about after all.

If you or anyone else is interested in writing for the magazine, we'd be more than happy to hear your ideas. Do get in touch! [LXF](mailto:lxf@futurenet.co.uk)

## 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 question – 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?

**Value for money:** Does it have a competitive price?

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.



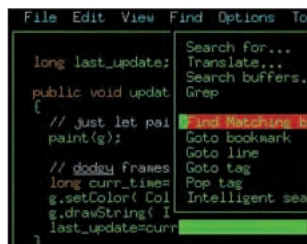
## THIS MONTH...

### Heavy Metal FAKK 2 >>

Find out if the gameplay is better than the plot as you pit Lara-like Julie and assorted weaponry against various improbable enemies **p22**

### MoneyDance

Money makes the world go round. Now Linux has a slice of the pie in the form of this accounting package for small business and home users **p26**



### << Crisp

A text editor that wishes to replace Vi and Emacs. Against them it pits extensibility, source code analysis tools and a flexible approach to windowing **p30**

### ProVSD

If you're looking at providing hosting facilities with plenty of features, ProVSD offers an alternative to buying an army of machines **p32**

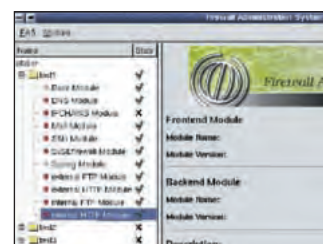


### Ice Pack Linux

Friendly new graphical distro with a good choice of window managers and an easy install. Are the configuration tools enough to tip the balance in its favour? **p24**

### >> SuSE Firewall

We try out the easy-to-configure firewall-on-a-disk from those masters of hardware detection and user-friendly configuration at SuSE **p28**



### << Books

Designing with JavaScript; Web Security, Privacy and Commerce; Python and XML; and XSL Essentials should keep you in your deckchair **p34**

## COMING UP SOON...

### Mandrake 8.2

We may have the download edition on the coverdisc, but we still intend to put the boxed version through its paces.

### SuSE 8.0

Expect a review of Germany's top distro soon.

### Volusion Manager 1.1

We've finally got our hands on the latest version of Caldera's network management system and the review will appear in issue 28.

### Morphon XML Editor

This validating XML editor is on test next issue.

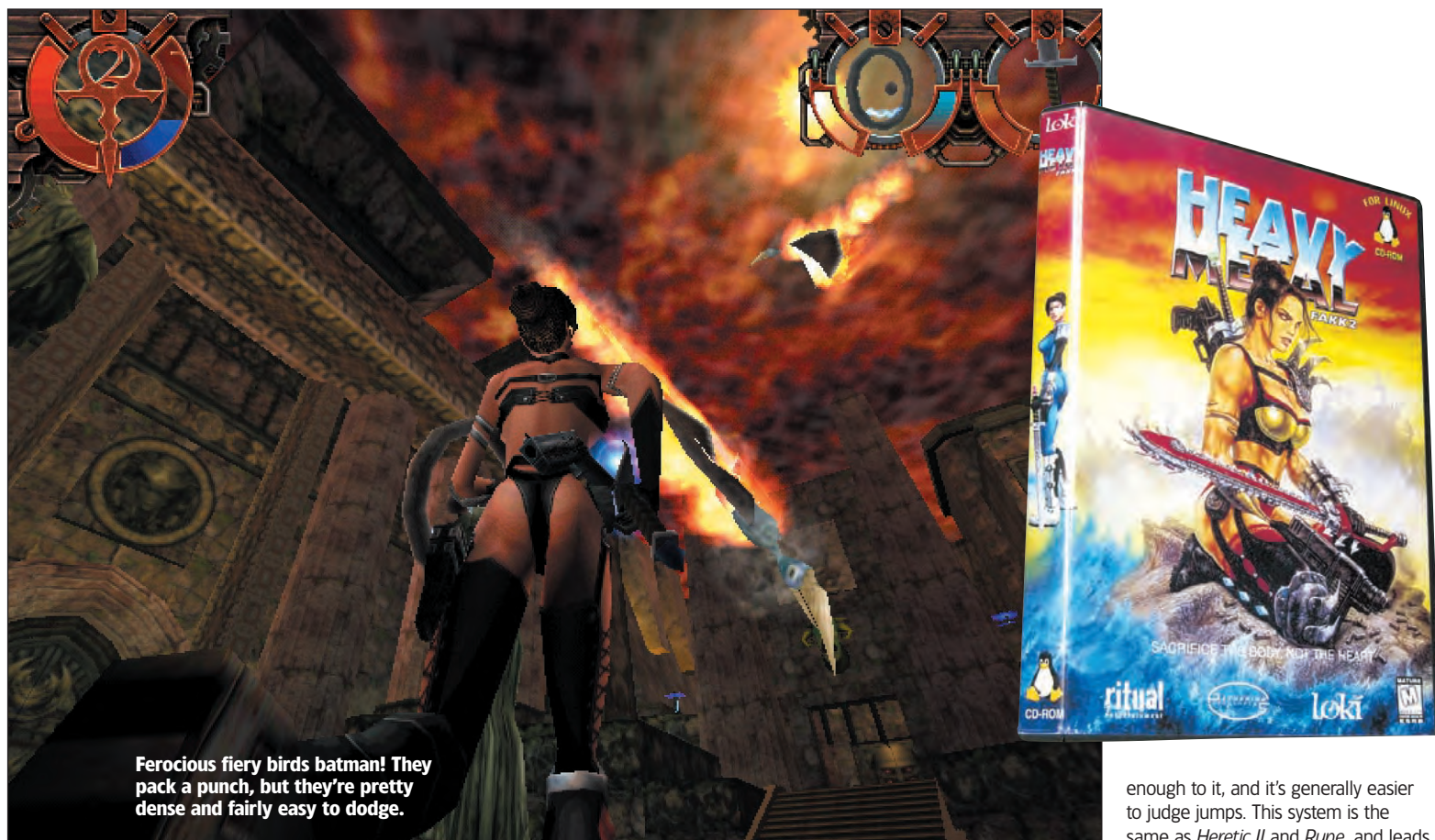
### Cyclades TS2000

Get the lowdown on Cyclades Linux-powered terminal server range as we road test the 32-port TS2000 in issue 28.

### Mozilla 1.0

It's taken four years, but the monster browser is almost ready.

## LinuxFormatReviewsHeavyMetal



Ferocious fiery birds batman! They pack a punch, but they're pretty dense and fairly easy to dodge.

## 3D SHOOT 'EM UP

# Heavy Metal: FAKK 2

With a flaming sword in one hand, a smoking pistol in the other, and a pert bum bringing up the rear, Julie Strain saves Eden from the evil alien hordes. Paul Cavanagh guides her along.

**Tomb Raider-style action game, but with far more character interaction. cf Half-Life.**

- **DEVELOPER** Loki Entertainment Software
- **WEB** [www.lokigames.com](http://www.lokigames.com)
- **PRICE** £30

**A**t the end of F.A.K.K. (it stands for something completely daft – there'd be no point in me telling you), Julie defeated some dark lord or another and she and all her chums found a cosy new planet to live on, named it Eden, and started settling to a bit of colonisation. Bless 'em. Years passed, but nobody aged because of the special water on Eden, which was nice. Peace, love and a spot of fishing hardly make for a decent computer game though, so before long something had to go wrong. A shield generator protects Eden and a beacon fools would-be invaders into thinking the planet is

barren and inhospitable. So when the shield generator starts going a bit wonky, you can bet it's a precursor to an invasion of monumental scale. Oh yes, the plot really is that bad, and the heroine of the piece really is called Julie (no offence to all the Julies out there, but it's hardly an heroic name). Thankfully for *Heavy Metal*, games are rarely judged on the quality of their narrative alone.

## All the right moves

Despite having a silly name, Julie is a worthy warrior, and is fairly nimble. Apart from your bog standard running and jumping, she's capable of climbing and swinging on ropes, swinging across monkey bars, crawling through tunnels, shimmying along ledges and lugging boxes about. When you consider all of this along with Julie's feminine attributes (she's got a wiggly bum and wears a selection of skimpy outfits), the words Lara and Croft may start to form in your mind. And you'd only be a bit wrong – it's obvious that the

developers wanted to benefit from the *Tomb Raider* success, but the control system here is simplified – Julie will always grab a ledge if she's close

enough to it, and it's generally easier to judge jumps. This system is the same as *Heretic II* and *Rune*, and leads to faster, more action-based gaming that's far easier to control with a mouse and keyboard.

## Game of two hands

*Heavy Metal* really comes in to its own with the weapons on offer. Julie's arsenal consists of swords, a sling with various ammo, guns and shields. The clever bit is that you get to decide on how to combine these tools of carnage – the idea being that the left and right mouse buttons correspond to Julie's hands. Because she's naturally right-handed, you can only carry swords in that hand, but you can choose whether to use her left for a



One of Julie's Lara-esque abilities is to shuffle along walls like this. Note the pretty waterfall in the background. Aaaah.



## Sound Fix

### Patch that buggy sound problem

Although some have reported the sound works OK, for a lot of people, the game crashes when trying to access certain sounds. Sadly Loki aren't around to fix the problem, but thanks to the work of Taylor Richards

and Gary Briggs, the problem seems to have been sorted.

Simply download the file from our website (or copy from the coverdisc) and place in the /usr/local/games/fakk2/fakk/ directory. That's it!

shield, or a gun, or you could forgo the sword altogether and go for a two-handed weapon. These include a dirty great axe, a flame-thrower, a dual-action rifle, a rocket launcher and a sword with a chainsaw built in.

Most of the time you're best off with one of the swords and a shield. This set-up allows you to get in close in relative safety using the shield and then unleash a vicious assault with the sword. Careful timing with left and right mouse-button presses will cause Julie to perform a deadly kendo-style combination assault. If this sounds familiar, it's probably because you read the review of *Rune* a few issues back.

While the sword techniques are identical to *Rune*'s, combining swords with guns is unique to *Heavy Metal*. Like *Rune*, there's a fiery sword here, and a very impressive electrical one, too, but you'll need to keep topped up on water to get the full effect from either of these weapons – Eden's magical water is available all over the place, but Julie's supplies go down each time she takes damage.

## Vanquished foes

The trouble with having such an impressive array of weapons is that it makes fairly light work of most of the enemies in the game. Once you've got the fire sword and have mastered the

combo attack technique, it's possible to dispatch nearly all of the bad guys with a few well-timed sword thrusts. There are a couple of enemies who'll put up a good fight – Flesh Binders are equipped with multiple weapons and can inflict a lot of damage, and there are the deadly Soul Harvesters, who are capable of teleportation and healing themselves, but these only appear in numbers very late in the game. Every so often boss creatures put in an appearance, and it'll take a few practice runs before you'll suss out the correct technique for seeing them off (although using the rocket launcher repeatedly works against most of them). Because of the availability of health pickups, and the fact that you can use the shield so effectively, seasoned gamers might prefer to play on the difficult setting, or maybe try using guns instead of shields. There are plenty of monsters wandering about including irritating flies, spitting vegetation, and hulking, electricity-hurling brutes. On the whole they're all nicely drawn and animated, and there's a fair amount of variety, it's just that it's all too easy to get to the stage where you watch the gore fly and the blood spurt. Oh yes, it can get quite nasty.

Thankfully, just about everything in *Heavy Metal* looks absolutely stunning (although I did get a bit bored of

looking at Julie's arse, and what's she doing wearing suspenders under leather? Poor girl must be chafing something rotten). While *Rune* suffered from some quite lacklustre levels, all of Eden's environments look amazing. Utilizing enhanced *Quake III Arena* technology, the levels are colourful with multiple textures and great lighting. There's a marketplace with lots of people to talk to, underground caves with luminescent seams running down the walls, waterfalls everywhere that look more-or-less real, and a misty marsh with nasty gaseous plants.

## Down in the sewer

There's also sewer. Bit of a bugbear with me, sewers in games – there's always one level in a sewer, but thankfully this isn't your average dingy tunnel, it's got high ceilings and all sorts of weird machinery in it. So that's OK. Exploring the levels is fun, and gets better as you go along – you encounter traps and puzzles in the form of spikey, crushing things; powerful updrafts; giant plants that spit Julie up in the air; scalding steam coming from pipes and, later on, temples with floating platforms, disappearing bridges and strange artefacts to collect. It could be argued that the puzzles are a tad easy, but at least they're not that annoying either.

There are some particularly good details, like plants with dandelion clock heads that disperse in the wind when you approach them, and the amazing graphical effects when using the flame and electrical swords. Disappointingly, switching dynamic lighting on didn't make the fire sword light up the area around Julie – that was one of the best features of *Rune*. Oh well, you can't have everything.

While the unofficial patch that we

installed (see *Sound Fix* box) prevented *Heavy Metal* from repeatedly crashing, it didn't sort out all of the problems relating to sound. Occasionally the sound would just pack up altogether, which could be temporarily resolved by quitting to the main menu and reloading the last saved game. Sometimes it was practically impossible to hear dialogue during cut-scenes because background noise was too loud. And sometimes everything just sounded plain awful. I'm not prepared to say hand-on-heart that the sound setup on our test machine was not to blame, but as there's a user-developed patch out there, it has obviously hindered other players. It should also be noted that both the Windows and Mac versions of this game were reported to have been buggy on release. What I'm saying here is be prepared for problems – you may well need to download patches or fiddle with your system before getting this game to work properly and this is reflected in the overall rating for *Heavy Metal*. Anyway, when the sound does work, it's OK – the Soul Harvester enemies give out spine chilling shrieks, and the sword swishy sounds are all pretty good. **LXF**

## LINUX Format VERDICT

Presentation	9/10
Features	8/10
Gameplay	7/10
Value for money	5/10

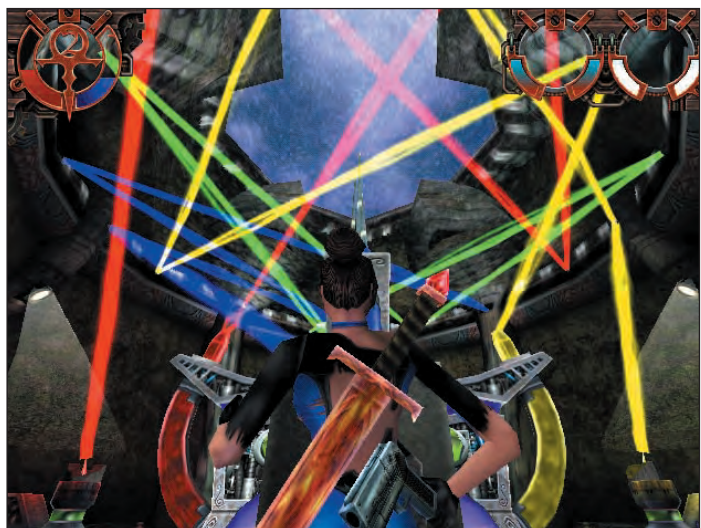
A very playable third person action/adventure game with great looks which should be a bit more of a challenge.

## LINUX Format RATING

7/10



A good example of the stunning visuals in the game – the fire sword graphics look even better when they're animated.



Eden's shield generator – very groovy. So where are Bungle, Zippy, Rod, Jane and Freddy then?

## DESKTOP DISTRO

# Icepack Linux 2.0

Still not found the perfect Linux install for your desktop? **Richard Drummond** finds out if this cool distro fits the bill.

A full-featured Linux distribution for the desktop or workstation.

- **DEVELOPER** Icepack
- **WEB** [www.icepack-linux.com](http://www.icepack-linux.com)
- **PRICE** 29.90 Euros (with Basic support) 34.90 Euros (with Premium support)

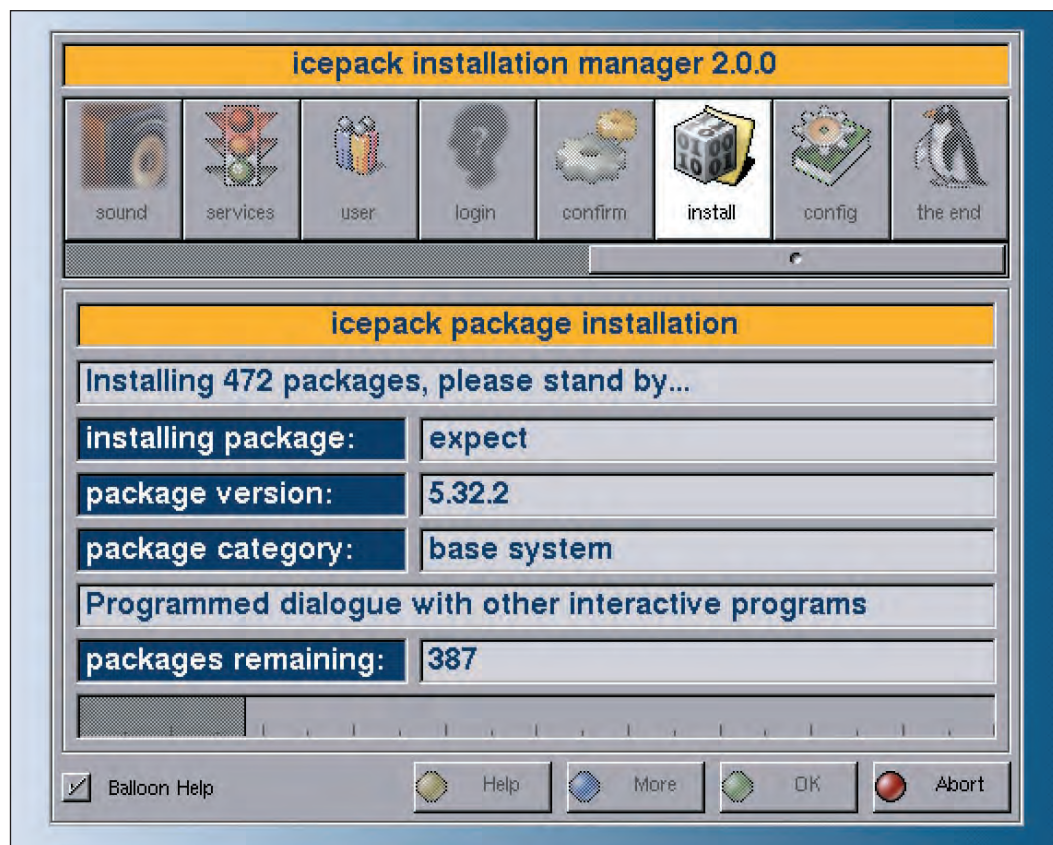
**S**USE may be the best known, but it is certainly not the only Linux distribution to hail from Germany. Icepack is an independently-implemented, RPM-based distro that aims to provide an easy-to-use desktop install of Linux, pre-configured with a wide selection of software. Icepack is optimised for Pentium machines, ships on 4 CDs, and includes a set of custom installation and administration tools. It offers the user a choice of desktop – either GNOME, KDE, Xfce or *Window Maker* – and includes a wealth of development tools, multimedia tools, office applications and games.

## Putting on ice

Installing Icepack is straightforward with its custom, graphical installer. Like all of Icepack's custom tools, this is based on the *WINGS* toolkit from the *Window Maker* project. This toolkit provides a colourful and refreshing *OpenStep*-like look, which makes a change from the Windows look-and-feel that is currently in favour with most other distro vendors. The GUI has design problems, though, and navigation is not always obvious. A favourite device of the authors is use indicators lamps – coloured blobs which reflect something's state – but it's often not clear what the colours are supposed to mean.

## Further information

**Icepack Linux ships with:**  
**Kernel 2.4.17**  
**glibc 2.2.4**  
**XFree86 4.2.0**  
**GNOME 1.4.1**  
**KDE 2.2.2**  
**Window Maker 0.80.0**  
**Xfce 3.8.12b**



The Icepack installer is colourful and easy to use, but its navigation controls could be made more intuitive.

Hardware detection works well in Icepack, and no real problems were unearthed during testing, only a few minor niggles. The detection of monitors failed on all the machines we tested – all of which had plug-and-play graphics card and monitor combinations. It even failed to set the correct frequency when the monitor being detected was one that was in Icepack's list of known monitors. Another disappointment is that NVIDIA's own X drivers are not included, so users of TNT and GeForce-based cards won't get hardware-accelerated 3D support by default. Also, the supplied kernel contains only the generic IDE driver: it does not contain support for specific IDE chipsets, so the higher DMA access modes are not supported out-of-the-box. Users with UDMA100 drives and chipsets, for example, will really notice the under-performance. On the plus side, Icepack offers a choice of ALSA or

OSS drivers to power your sound card.

The installer provides a friendly front-end to GNU *Parted* for creating disk space, and this supports the non-destructive resizing of Windows partitions. Installation to *Reiser* and *ext2* filesystems is allowed, but there is no support for *ext3fs*. A limitation with partitioning in Icepack is that package selection is performed before the creation of the filesystem, and it won't let you split the install across multiple partitions – you must at this point pick a single partition that is big enough to contain the complete install.

Package selection is perhaps the most confusing aspect of the installer – partly due to the coloured indicators I mentioned above, but partly because of the lack of control or information it provides. Icepack categorises its software into package sets. The base sets are always installed, and the installer lets you choose which of the optional sets you need. These are classified firstly according to which

desktop environment they belong to – KDE, GNOME, Xfce or *Window Maker* – and secondly according to type – utilities, multimedia, development and so on. Additional package sets include extra games, extra documentation and the *OpenOffice* and *StarOffice* suites.

The safest bet here is to go with the default, complete install, because, since you don't know exactly what software is included in what set, it's often difficult to tell what the result of de-selecting a particular package set will be. What's more, the configuration of the installed system does not adapt with your installation choices: you get the same menus whether you install all the package sets or not. It is incredibly frustrating to try to launch a program from the desktop menu only to be told that it could not be found. It's even possible to select Xfce, for example, as your default X session regardless of whether Xfce is installed.

Icepack has a custom boot menu system and an easy-to-use tool for





**Icepack has plenty of custom tools, including its own login manager.**

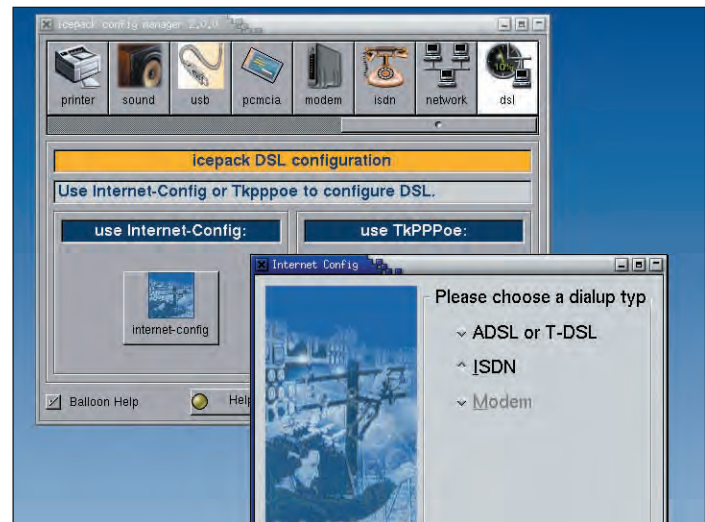
configuring this. The boot menu can be installed on a floppy or the MBR of your hard disk. A nice touch is that during installation, you can add as many entries to the boot menu as you wish. This makes a change from distros that wipe over your MBR without caring what existing operating systems are installed on your system. The GUI only lets you choose which partitions that you wish to be able to boot: you cannot specify boot options such as which kernel image to use. This is because the boot menu requires a boot loader such as *LILO* be installed on a Linux partition to actually do the job of loading up Linux. It knows how to start Windows itself, though.

### More than one way

When Icepack finishes installing from the first CD, you are asked to reboot and start up Icepack. When loaded, Icepack will then continue to install

any software required from the remaining 3 CDs of the set. Afterwards, it presents you with its custom login manager, and you can begin using Icepack properly.

Icepack provides a central point for most system-oriented configuration jobs: the *WINGS*-based *Icepack Configuration Manager*. This echoes the look and feel of the installer and provides run-time analogues of some of the tools used by the installer. Thus, here you can mount partitions, reconfigure X, administer users and the login manager and so on. Other tasks, like package management, must be done with external tools that can be launched from the central Configuration Manager (providing you didn't unknowingly de-select them during installation). In the main this system works well – with one caveat. The Config Manager requires root access, but there's no graphical shell



**The configuration console simplifies many of the basic admin tasks.**

for *su* launched – such as the one KDE desktop provides – when you start it up. You need to become root manually.

When you first log in, the first task is to do the configuration jobs not done by the installer, namely setting up a dial-up Internet connection and your printer. As with everything, here Icepack overwhelms the user with choice. From the *Icepack Configuration Manager*, you can launch either the web-based *CUPS* administration interface or the KDE-based one – or opt to use *TurboPrint*. Similarly, there is a choice of four admin tools for setting a modem connection, and yet more options for ISDN and DSL. While choice is good, the user isn't provided with any information about what are the pros and cons of the various options. You either require previous knowledge or you'll have to suck-it-and-see. In this light, Icepack is not a good distro for beginners.

The user is bamboozled by choice when it comes to application software too. You get at least five browsers by default, including *Netscape 4.7*, *Netscape 6.2*, *Mozilla*, *Konqueror* and *Opera*. There's nothing wrong with that, except what if the user didn't really want five browsers? This is a corollary of the inadequate package selection tools during installation. Similarly, you get several different picture viewers, multimedia players and so on. Office applications include *KOffice*, *AbiWord* and you also have the choice of installing *OpenOffice* and *StarOffice*.

Talking of packages, package management in Icepack is done with manual tools such as *KPackage* and *GnoRPM*. There are no tools for automatically fetching and installing packages or updates from the web. You'll even have to manually install any additional packages from the Icepack

CDs. We have come to expect automated tools such as Debian's *apt* and Mandrake's *Update* and the lack of this facility in Icepack is a point against it.

Icepack Linux is a competent, no-frills desktop edition of Linux that offers excellent value for money. It provides simple installation, some good basic configuration tools and a wide (perhaps too wide) selection of application software. What really lets it down, though, is attention to detail and a poor package installation system. Consequently, I wouldn't recommend it to the complete novice. The brittleness of the installer means that its far too easy to end up with an installed system missing key tools. This is a shame, because there are some nice features for the beginner in Icepack. For example, the boot manager and user admin sections of the *Icepack Configuration Manager* are well implemented and easy to use. Moreover, it's refreshing to have a choice of four desktop environments. If you like *Window Maker* or *XFce* then Icepack is a good choice to provide a quick, application-packed install. Beginners who want a slick GNOME or KDE-based desktop would be better off looking elsewhere. [LXF](#)



**The option of *Window Maker* makes a change from just KDE and GNOME.**

### LINUX Format VERDICT

Ease of use	8/10
Features	7/10
Performance	6/10
Value for money	9/10

Good configuration tools and plenty of applications, but lacks polish and any kind of automated update facility.

### LINUX Format RATING

7/10

## ACCOUNTS PACKAGE

# Moneydance 3.2

**Chris Howells** looks at an application to help handle your finances.

Home/small business accounts package. Competing with GNUcash and Kapital.

- **DEVELOPER** Appgen
- **WEB** [www.moneydance.com](http://www.moneydance.com)
- **PRICE** \$39.99

For a long time, the ubiquitous-but-generic spreadsheet has been used to handle finances. Recently a plethora of applications has sprung up to help handle your personal finances. From *GNUcash* to theKompany's Qt-based *Kapital*, there is now a fairly large range of GNU/Linux-based financial software.

## Supported platforms

*Moneydance* is quite an interesting product in that it is written in Java, and consequently can run on just about platform that has a Java VM (virtual machine). Officially supported platforms include Linux, Windows, MacOS (including OS X), FreeBSD and Solaris. Rather impressively, this will make switching between multiple platforms extremely easy since the program should work in exactly the same way on different platforms.

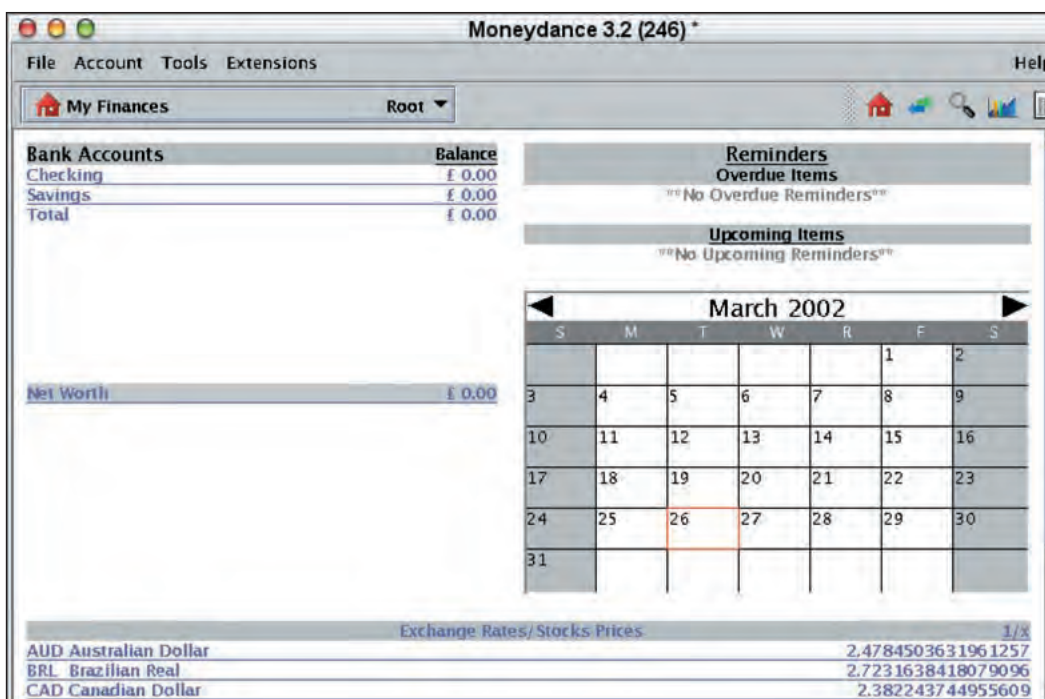
Unfortunately, since Java is interpreted by the VM, execution is not quite as fast as natively compiled code.

## Installation

*Moneydance* is supplied on CD, along with all of the software required to run it on different operating systems. In addition to the CD, a *Getting Started & User Manual* is provided.

The installation of *Moneydance* is simple and described adequately well. The instructions in the handbook that we received for review were based around a *KDE1* desktop, with the *kfm* file manager. This is now considerable out of date. That said, *KDE3* is sufficiently similar that the instructions are virtually identical. Under other desktop environments, if the file manager does not support running executables by clicking on their name, installation can be started by simply running the *install.sh* shell script from a terminal window.

The *Moneydance* installer is certainly one of the better ones that we have seen recently – not graphical, but quite versatile. After detecting that the user is not running as root, the



The Home screen provides access to most areas of the program.

installer asks whether to become root and install into a system location such as `/usr/local/moneydance`. The other requirement for *Moneydance* is a Java VM. On my Debian 3.0 (Woody) installation, I simply used *dselect* to install *jdk*, a Java VM.

To finish things off, a .desktop file (as used by both KDE and GNOME for menu entries) is provided that can be placed onto the desktop in order to provide an easy link to the application. Very impressive. However, in subsequent versions it would be nice if the installer could automatically place the icon in the right place without requiring the end user to do so.

## First time

On loading the program for the first time, you are presented with a screen which allows you to start a new file or open an existing file. The first time the program is started, the option to create a new file should be taken. After choosing this option, basic characteristics can be set, such as the main currency to use for that particular database.

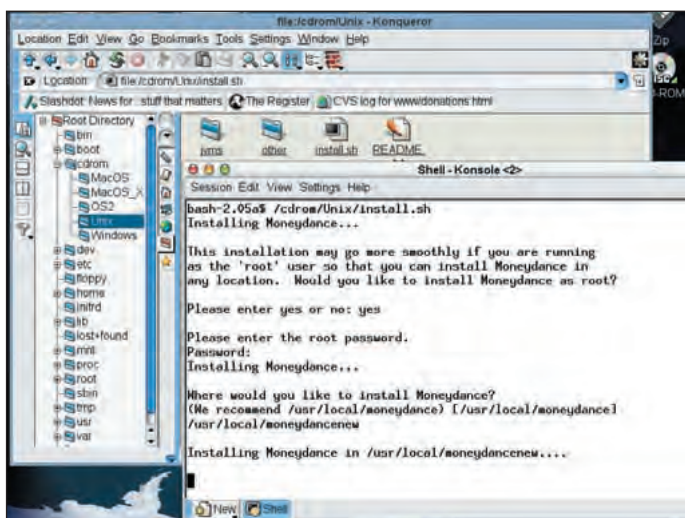
*Moneydance* features two different user interfaces. One is aimed at more experienced users or accountants and the other at home/SOHO users. The

idea is that accountants will be happy with a more powerful but more difficult to use interface, while home/SOHO users should be happier with the more simple interface. In this review, we've looked at the more complex of the two interfaces.

Once a file is loaded, the "Home Page" area of the program is opened. The Home Page acts as a central way of accessing other areas of the

program. Amongst other things the screen offers a calendar, reminders, and details of the accounts, all at quick glance.

After selecting a currency of "British Pounds", *Moneydance* correctly set up the program to use the expected date format `dd/mm/yyyy`, but the locale was still set to *English* rather than *English (UK)*. Thankfully this, and many other settings, are easily



The installer is fast and works very well.



## System requirements

The following requirements are recommended for the Linux version of *Moneydance*:

- *glibc 2.0, glibc 2.1* recommended
- **200MHz** processor
- **64MB** RAM
- **2.x** kernel

Requirements for other operating systems are similar

changed in the Preferences dialog.

Being Java based, the look and feel of the program depends on the Java VM – the typical widgets are not very beautiful, but functional enough.

## Features

After the program is set up, you're ready to start creating the "Chart of Accounts", the database which is at the centre of *Moneydance*'s accountancy features. The Chart allows you to add details of various money related resources. This includes credit cards, bank accounts, assets, and loans. For instance, you may wish to add details of relatively high-value items like cars or computers. This is all possible through the Chart interface.

After adding details of all your accounts/assets, it is then possible to add details of transactions. This can be accomplished by going to the Home Page, where all of the items are listed. After clicking on the account/assets that you wish to edit, you are taken to a new screen where details can be entered. While it has been reasonably intuitive up till now, the entry screen could be improved slightly. Overall though it's quite easy to set up details of all finances.

The user interface is reasonably clear, although there are too many jargon-filled options (looking at the manual is therefore required) for the average home user to get into the program quickly.

*Moneydance* also includes an option to use the program for online banking, although at the moment only two American banks are available – not so useful. Hopefully some UK-orientated facilities should be available soon. Additionally, we were unable to get the option to download the list of available banks to work, due to a **java.net.UnknownHostException** error. Therefore the program loses marks in terms of robustness and reliability. That said, this was the only major error that we saw.

The program is supplied with a set of exchange rates for different currencies. Inevitably, these will change, and *Moneydance* supports the ability to connect to the Internet (via a proxy, if required) to download the latest exchange rates. Updates can be set to download on startup. It is possible to force an update at any time, although unfortunately it's not particularly obvious how to do so.

In addition to being able to organise standard accounts, *Moneydance* goes a long way in assisting with other money related matters. For example, it's possible to convert values into their equivalent amount in different currencies, based on the current exchange rate.

*Moneydance* features the ability to import from *Quicken*, a popular Windows-based accounts package. This works quite well, although it would certainly be useful to see support for other formats added, such as other Windows-based packages, perhaps in addition to the various open source programs, in order to make the switch

over as easy and painless as possible. *Moneydance* can also export in *Quicken* format.

Various graphing options are provided. This is a very useful option since at a glance it's possible to see how your finances are progressing, and take appropriate action, if necessary.

For security purposes it's possible to encrypt data with a password, although it's not clear exactly how strong this encryption is.

Rather cleverly, *Moneydance* allows its capabilities to be extended using a plug-in system. At the moment, various useful plug-ins such as a Euro converter and python scripting interface are available.

One of the more unusual options is an address book. This does seem somewhat out of place in money management software, although it will no doubt be useful to some.

## Documentation

*Moneydance* is supplied with a useful printed handbook. This is a fairly substantial document running to 70 pages including screenshots, although it is only printed in black and white. In addition to this, the program features online documentation via the Help menu. Unfortunately sometimes the program is not very verbose, and it's not clear what certain feature do. It is therefore unfortunate that context sensitive help is missing from the program.

The online documentation is presented via a browser style interface, and essentially contains similar content to the printed handbook.

## Localisation

Localisation is an important part of any software. While *Moneydance* makes an attempt at using settings (dates and spellings) which will be suitable for

British English speakers, some other areas could be improved. For example, the phrase "ATM" is used, rather than the more common "cash machine" in the UK. Additionally, *Moneydance* refers to various US taxes which are named differently or just do not exist in the UK.

This could be a source of great annoyance to anybody using the program, and you should ensure that you are happy with the alternate names before purchasing the program.

## Conclusion

*Moneydance* starts off very nicely. The installer is extremely pleasant, and the program is reasonably quick and easy to get in to. However, slightly further in, things start to get unpleasant with various accounting jargon. Really there is no alternative here to sitting down with the manual and learning the terms, which should not take too long.

After a little practice, *Moneydance* becomes an efficient and effective program to use. However, with competing programs such as *GNUcash* and *Kapital* providing similar features, it would be wise to look at these alternatives as well, since they are feature rich, and may well provide all that you need. **LXF**

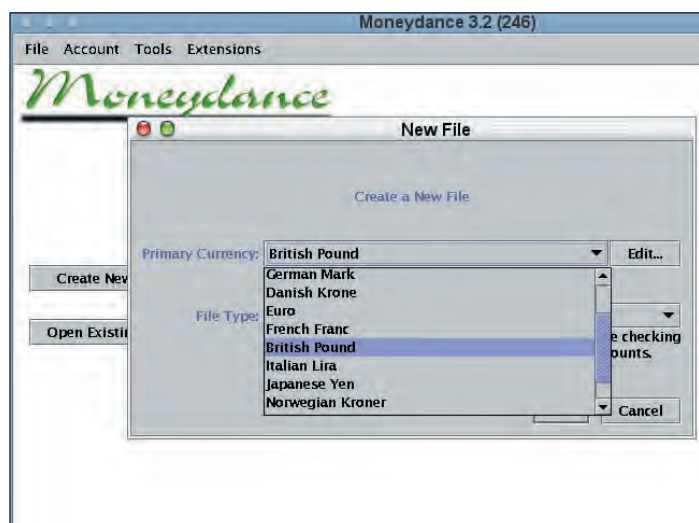
## LINUX Format VERDICT

Performance	<b>6/10</b>
Ease of use	<b>7/10</b>
Features	<b>7/10</b>
Value for money	<b>8/10</b>

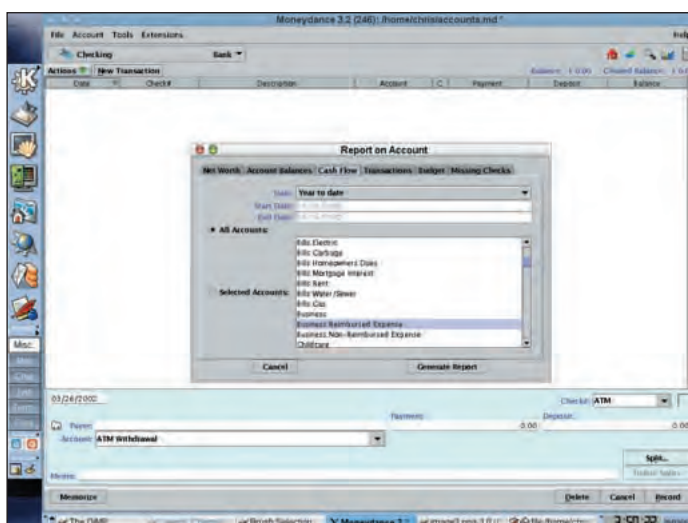
Useful and quite easy to get into, although you'll need help from the manual for certain features

## LINUX Format RATING

**7/10**



On choosing a new file, you can set preferences such as the currency.



Transactions can be entered, and reports based on them created.

## FIREWALL ON A DISC

# SuSE Firewall VPN

Want a secure firewall whose scripts cannot be overwritten by crackers? **Tom Wilkinson** investigates a product which runs entirely from read-only media.

CD-based firewall system developed by SuSE. Includes support for web proxying, VPN and bail handling. Compare Mandrake *Single Network Firewall*, Smoothwall and Trustix *XSentry*.

- **DEVELOPER** SuSE
- **WEB** <http://www.suse.co.uk/>
- **PRICE** £933 + VAT

Security is big business these days. Whether it's for a single machine connected to the Internet via a dial-up account, or hundreds of PCs sharing a leased line. Unlike other distributions' firewall products aimed at the home to small business users, SuSE's product aims high, supporting multiple internal security zones, HTTP proxying and filtering, and logging.

Unlike many competing products, SuSE firewall actually requires two separate machines to administrate the firewall – a firewall management machine, and a machine for the firewall itself, which resides on a read-only CD. Additional configuration options are kept on a write-protected floppy, meaning it's possible to run the firewall without any possibility of changes being made to the system by malicious intruders, if they do get that far.

The firewall management machine runs a special install of SuSE 7.2 designed for that specific purpose. In addition, the firewall management software can be installed on any existing SuSE Linux workstation. It might have been useful to be able to install it on any existing linux machine, such as Red Hat or Debian. This option is not available, however. I suspect this could be for marketing reasons.

Inside the box are three CDs – an installation CD for the *Firewall Administration System*, one for the firewall software itself, and a third CD containing source code pertaining to the first two. In addition, there are two manuals, one for the firewall software and a second, more general manual for SuSE 7.2. This is the standard manual as supplied with all versions of SuSE. Something that might have been a nice inclusion was a blank



The Firewall Administration System, showing a number of the modules available to the user.

floppy disk. Like many people, I hadn't used such an esoteric piece of technology in years, and finding a (working) floppy proved quite a challenge. Perhaps a future version of the software will support using a CD- writer to create a unique CD for each firewall machine.

The manual is, typically of SuSE, excellent. It starts off by describing a number of common firewall setups, including the network with a single machine between it and the rest of the world, and much more complex topologies such as networks with demilitarised zones and those with multiple subnets, including a brief history of the firewall as a concept. It then goes on to explain the procedure for installation of the "adminhost" machine in detail, including how to install the software on an existing SuSE workstation as well as installing as a fresh machine.

It does include some extra detail of the files modified by the setup program. It then goes on to list in

detail a number of the functions of the firewall itself, and how to set it up.

## Installation

Installation of the firewall administration machine is quite straightforward and follows the usual SuSE steps through *Yast2*. However, on our test machine, an 800MHz PC with 256MB of RAM, the install crashed consistently at the same point. Nothing I attempted could fix this, however I was able to install it on a second PC of very similar specification. I suspect a problem with the original graphics chipset, however I was unable to find any other distribution that failed similarly. In the end, I was able to get the installation to triple-boot with Windows XP and Red Hat Linux 7.1 – not the most ideal of setups. However, it worked, which is more than can be said for the first installation attempt.

The custom installation script creates a single non-root user, fwadmin, which is the login used to administrate the firewall. Firewall

## Hardware requirements

**Processor:** Pentium or better  
**Memory:** 128MB (Firewall), 256MB (Administration machine)  
**Disk Space** 1GB – 10GB (Firewall, for logfiles and cached data), 2GB minimum (administration machine)  
**CDROM drive**  
**Network card** (2 in firewall)

administration can be done from any SuSE machine with the *Firewall Administration System* installed – the firewall customisation software runs as a central daemon, ensuring that a central repository of confirmations can be made, keeping version control problems to a minimum. The default desktop is *KDE2* – in fact, it's the only desktop available but there's no need to install any other if the machine's being used as intended, for firewall configuration only.



## Administration

Once the firewall administration machine is up and running, it is a simple matter to start the firewall administration program. This presented an empty window with a menu bar across the top. The two options available are to create a new user, or to log in to the *Firewall Management System*. I couldn't get the username and password I'd created during the setup process to work, however, so had to recreate a valid user. Once I'd done this and logged in there were more options to access, including the creation of a new firewall configuration.

Although an online help system was displayed, it was of very little assistance – it instructed me to click on an option that wasn't visible on the screen to start configuration. It eventually turned out that the left pane of the window was a file-manager-like dialogue, and clicking in the right place expanded the tree showing a number of different attributes for the firewall with red 'lights' next to them, indicating incomplete configuration, or grey, indicating configuration should not yet be attempted.

Considering how easy *YaST2* is to use, I was surprised at the quite ugly design of the system, both in terms of ease of use and layout. A great deal of the screens feels cluttered, even

## Software Included:

**Kernel 2.2.19**  
**Squid HTTP Proxy**  
**Postfix MTA**  
**Bind 8**  
**http/tinyproxy (Used for content filtering)**  
**OpenSSH (for secure shell access, but you should update to latest version!)**  
**SuSE FTP proxy**

though there's little information presented, and the whole application has the feel of something a novice Windows programmer might create in Visual Basic. I hope this improves in the future editions. User interface design is one of the most important parts of a program, and in this aspect it falls down pitifully.

Problems were also encountered after rebooting the machine. While it was possible to log in, either as root or the fwadmin user, the KDE desktop failed to appear correctly, and the only way I could access the system was to switch to a text-based terminal, and log in. As the *Firewall Administration System* is graphical-only, this made administration impossible. Notable, however was the rather draconian warning displayed at the top of the screen warning that unauthorised access to the system was illegal and

that by logging in I agreed to have my actions recorded. That's fine, but surely if the firewall had been compromised and intruders had got that far, they'd be ignoring the warning anyway? In addition, I was unable to restore the KDE desktop to working order. This meant a reinstall of the *Firewall Administration System*.

Once that was complete, I made sure to complete the configuration of the firewall in one setting this time in case similar problems arose again. This involved a set of steps through a variety of wizards, selecting such options as where to log to (either the local hard disk, or remotely to a machine running the Firewall Administration System, and the routing that the system needs to do. This includes what IP addresses belong to each interface, what to use as DNS servers and so on. A notable option here is whether or not to allow the root user to log in to the console. This aids security in a way that one-machine solutions such as Smoothwall and Mandrake Firewall fail to. If there's no shell present on the server this cuts down seriously on the number of potential exploits available for a machine – a great many rely on being able to start a shell.

A number of optional modules exist in addition to the basic configuration. Some are more optional than others, however. A good example of this is the *IPChains* module for

firewall configuration. A good ten pages of the manual are devoted to this, merely as a reference. The configuration script can be run separately from the *FAS* system, and consists of a set of questions which the user must answer. The script has a number of lofty aims which appear to have been achieved. These range from preventing configuration errors from opening holes in the packet filter to ensuring that setup is easy and has support for a diverse range of interface configurations. This seemed contrary to earlier parts of setup which showed that each interface required a static IP address. Here you could define IP addresses on any side of the firewall (external, DMZ, internal) to be static or dynamic. There also seemed to be confusion over the types of interface supported – the manual mentioned PPP and ISDN – but I didn't find any way of entering them into earlier screens for later use.

One of the most important parts of this product is its support of Virtual Private Networking (VPN). This is a networking technique whereby several physical networks, separated by large distances, can be used as if there were a single physical infrastructure. This is attained through the use of encryption and transmission of the data over the Internet at large. The sufficiently high grade of encryption means that it's unlikely that anyone will crack it, and as a medium it's much cheaper than leasing lines between sites. The manual, however, didn't offer much in the way of instructions on how to set this up. Perhaps it's presumed that if you're using a VPN you'll already have some experience of its setup.

There is, however, as with all SuSE products, 30 days-worth of installation support included in the price. Once this is over, there's also the option of taking out a further support contract with SuSE. This could be a good idea if administrating a large network with a number of SuSE linux installations. However for experienced administrators this may not be worth it. [LXF](#)

*Yast2* is used in the installation of the administration box, but is less flexible than in a normal installation.

## LINUX Format VERDICT

Ease of use	7/10
Features	9/10
Performance	8/10
Value for money	7/10

An excellent option for administrating firewalls for larger networks than most of the cheaper competitors.

## LINUX Format RATING

8/10

## PROGRAMMER'S EDITOR

# CRISP 8.0.3

Can any text editor seduce **Richard Drummond** away from his beloved Emacs? Read on to find out.

Cross-platform, programmable text editor comparable to Emacs and Visual SlickEdit

- **PUBLISHER** Vital
- **WEB** [www.vital.com](http://www.vital.com)
- **TEL** +1 972 818-2424
- **PRICE** \$150 (per user, non-commercial) \$250 (per user, commercial)

Nothing inspires religious devotion in developers like their choice of text editor. So what happens when you move to a new platform and your favorite text-basher isn't available? *CRISP* is a commercial editor that helps overcome this pain. Not only can it mimic the keyboard-mappings of such favorite editors as *Brief*, *Vi* and *Emacs*, but it is available on a host of platforms including must Unices, Windows and MacOS X.

The Linux version ships with two executables: one with a *Motif*-based interface for X, and one with a text-based interface for running in a console. The latter includes most of the functionality of the X version, including drop-down menus and console-versions of the dialogs, and is great for running in a remote shell. When running on X, *CRISP* is none-too-pretty – it brings an even more

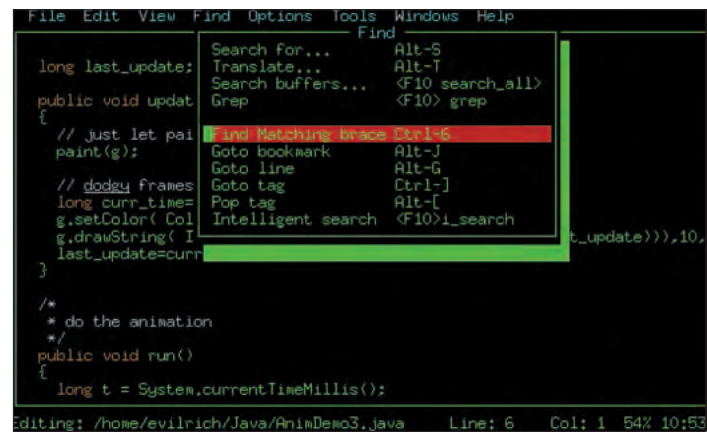
Windows-like look to *Motif* than usual – but it is clean and fast.

*CRISP* has a flexible approach to handling multiple documents. You can open any number of MDI-style windows within the main *CRISP* X window and you can open multiple main windows; each internal window can be split into multiple views. Each document opened is listed in a tab widget underneath the toolbar – and clicking on a file here displays it in the currently active view.

## Tag! You're it

*CRISP* offers all the usual bells and whistles for the programmer including syntax highlighting for dozens of languages, templates, completion, bracket matching, macros, source code beautification and so on. A neat and unique feature, though, is that it can spell-check source code – or at least identifier names and comments.

The majority of *CRISP*'s source code analysis and navigation tools are built around tag files. It automatically tags any document you load, and a tag builder is included to simplify the generation of tag files for projects. In addition, the tag manager can build tag files from your C/C++ system libraries and Perl modules by scanning the relevant headers. No support for building tags from a Java class library is currently available, however, which is disappointing. *CRISP*



The ability to run *CRISP* in a console can be a real life-saver.

## CRUNCH time

Extending *CRISP* with CRUNCH macros

*CRISP* can be extended by writing macros in its embedded C-like scripting language called **CRUNCH**.

**CRUNCH** macros are compiled for speed and *CRISP* includes all the

tools required for developing in **CRUNCH**. Much of the functionality of *CRISP* is itself implemented as **CRUNCH** macros. The source code to these is supplied and provides a good source of reference.

includes its own standalone tag command called **crtag**. This supports both *Vi* and *Emacs* style tag files, but by default, stores tags in a binary format for speed.

The content window – the tabbed window to the left of the main display – contains various browsers to view and navigate the contents or state of the current editor session in *CRISP*. Here, you have the self-explanatory clipboard browser, file browser and ftp client. Also included is the tag/class browser, which shows a tree of all the tags defined in your project and globally – clicking a tag here will open the document where that object is defined – and the section browser, which shows a tree-view of the structure of the files in your project.

*CRISP* provides some rudimentary project management tools and can integrate with various build environments to imitate a full IDE. In most cases, though, this will require some configuration. Here, *CRISP* basically acts as a front-end to *make* or similar systems. You can specify various compilation phases and the commands require to build them. *CRISP* can also capture the output

from a build phase and display it in its integrated output window.

## Conclusion

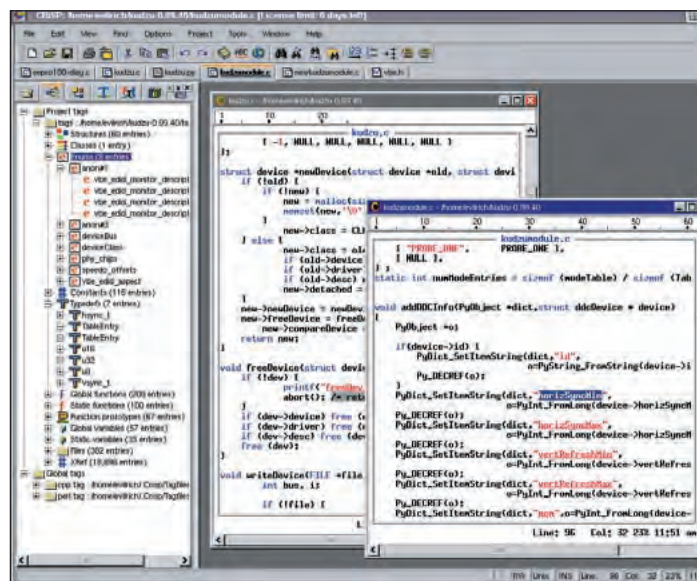
*CRISP* is an excellent product. It offers the power of *Emacs*, while being significantly easier for the new user to get to grips with. In particular, it requires a lot less configuration of the more advanced features – it ships out of the box as a fairly complete environment, suitable for editing C/C++ projects, Python, Perl and so on. Some languages, notably Java, are much less well supported, however. **LXF**

## LINUX Format VERDICT

Ease of use	8/10
Features	8/10
Performance	9/10
Value for money	7/10

A fast and friendly editor with good source code analysis tools, and which rivals the mighty *Emacs* in extensibility.

**LINUX Format RATING**  
**8/10**



*CRISP* offers excellent project navigation by building and using tag files.



## WEB HOSTING PLATFORM

## ProVSD 2.8

David Coulson looks at a neat hosting solution which provides virtual admin privileges to all.

Makes cheap dedicated hosting a reality. With limited competition from *FreeVSD*, along with the UML kernel patch, it has a fairly open playing field.

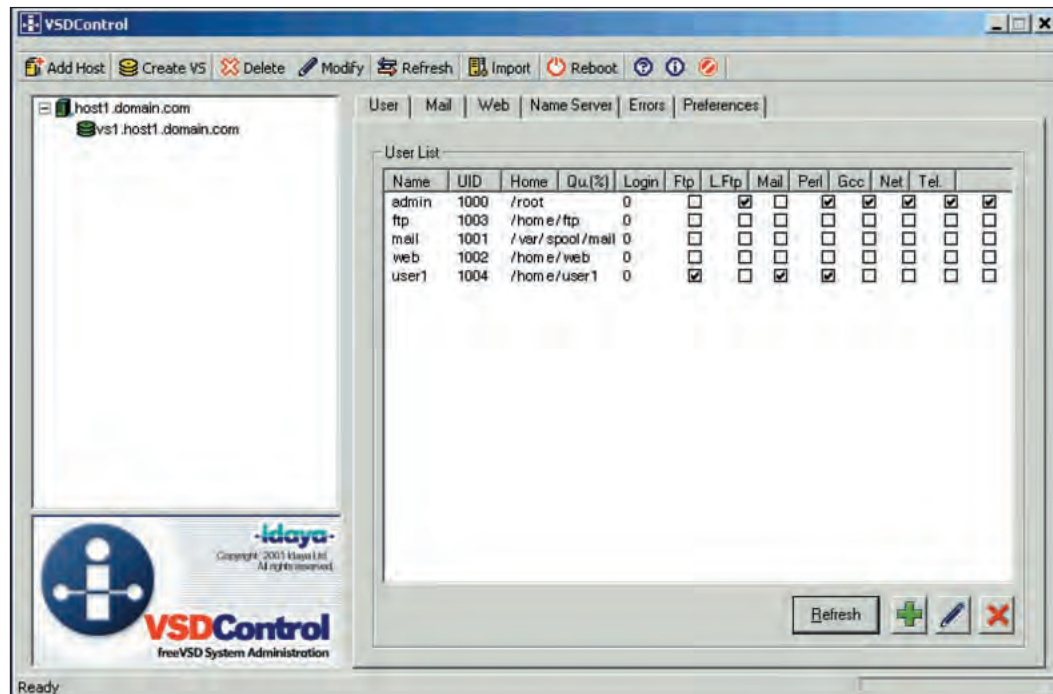
- **PRICE** From £110 per quarter for five virtual servers
- **CONTACT** Idaya Ltd
- **WEB** [www.idaya.com](http://www.idaya.com)
- **TEL** +44 (0) 118 949 7009

Everyone has to have their little home on the Internet, and their expectations of a hosting service have gradually become more and more demanding. Everything from CGIs, PHP and ASP through to MySQL, PostgreSQL with POP3 mail accounts and administrative tools so they can control it all themselves. Providing each of these services for people to use is not difficult, but giving users complete administrative control over the server configuration is near to impossible, unless you give each user a dedicated server with a separate installation of Linux – expensive.

Idaya produced *FreeVSD* for a UK ISP, with virtual hosting services in mind, and found it to be a rather popular solution. A freely distributable version followed rather quickly (reviewed in LXF23), but the majority of the people who need such an app have neither the time nor money to play around with unsupported code which might not work even after spending days struggling with it. To fill this demand, Idaya have produced *ProVSD*, a fully supported build of their code, along with the latest and greatest configuration tools for both Linux and Win32 platforms, and web-based administration.

*ProVSD* needs a Red Hat 7.2 system to function properly, although it will work on earlier versions with a few upgrades. Distros such as Debian and Mandrake are unsupported, but as it's much cleaner to have *ProVSD* set up on a brand new installation, you'll probably be removing whatever was on the machine in the first place.

For hardware requirements, it is entirely dependent upon the number of virtual machine instances you will be running, and the services which each virtual machine will be using. Generally, 1GB per virtual machine of



**VSDClient** gives users a very neat front-end to the standard services which are running on their virtual machines.

disk space – although since *ProVSD* makes use of symbolic links there is little overhead per virtual machine – and 6–8MB of RAM for a basic *ProVSD* configuration including Apache, MySQL and Sendmail. Suffice to say, the more memory and CPU power you can throw at it, the happier everything is. Idaya recommend dual PIII systems, although there is no reason why it could not be run on a single CPU machine, as long as the number of virtual machines were limited appropriately. *ProVSD* can handle more than 250 virtual machines on a single host, which should be more than enough for anyone, and once you pass a certain threshold, it is cheaper to have more than one host, and distribute virtual machines between physical hosts.

### Root privileges

Rather than being 'root', each virtual machine has an account called 'admin', who has root-like privileges to reconfigure and restart services within their own virtual machine. These accounts are purely regular user accounts on the host, but the permissions are set for the services for each virtual machine such that only admin can perform operations which

would normally be reserved for root. Within each virtual machine, 'admin' can create standard user accounts which are limited to that specific instance, although again, they are standard accounts on the host. When any user, including admin, shells into the virtual machine, via either *ssh* or *telnet*, they end up with a *bash* prompt which works exactly as if it were on a real host. You have everything living in a virtual "/" filesystem, so your home directories are in /home, and config files in /etc. *ProVSD* does this by having all logins start within a *chroot* environment; so as far as they are concerned, the virtual machine's fake root is in fact the real root for the service. Of course, 'admin' can still manage user accounts as root would on the real host, as *ProVSD* has modified versions of many standard Unix tools, including *rm* and *passwd*, so that 'admin' users can have complete control over user accounts within their virtual machines. However *ProVSD* is set up, the real 'root' of the host machine always has power over files, processes and services on virtual machines, so if there is anything misbehaving or otherwise being naughty, they can take control.

As you would expect, certain things

still need to happen as root on the host, including the rather important matter of binding to a port for any service. *httpd*, for example, runs as a non-root user, so is unable to bind to a port less than 1024, which *ProVSD* gets around by running *Apache* on a high port, 8080, then has a simple redirector running as root on the host which forwards all traffic from port 80 to port 8080 within the virtual machine. This makes the whole process fairly transparent and, as one would expect, avoids certain security pitfalls of running things as root. Many other services, including *telnet* and *ftp*, are handled via *inetd* on the host, and are forwarded to our virtual machine's *telnet* and *ftp* services. To facilitate this type of service, each virtual host has its own IP, which is really an IP alias on the host's physical ethernet device, although there is nothing which stops virtual machine services binding to ports on IPs which do not belong to it, however due to the use of redirection from low ports by *ProVSD*, port 80 on a virtual machine will always be forwarded to port 8080 on the same machine, and the individual admin users can set up which ports are forwarded and which are not. It's down to the host administrator to configure a

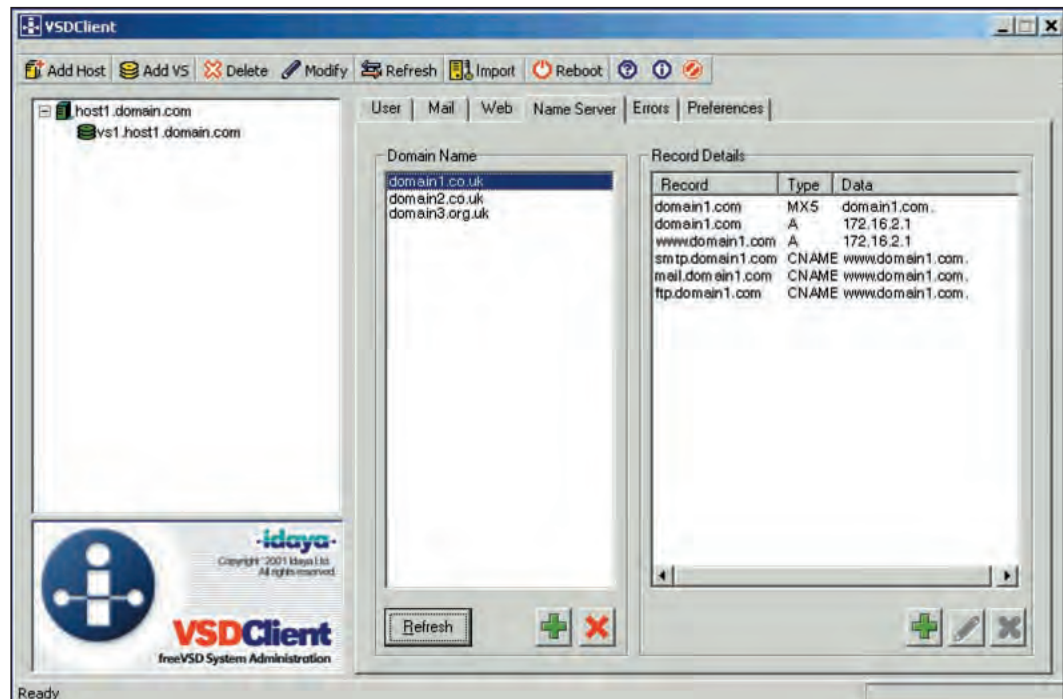
firewall on the physical machine, as individual virtual machine owners have no control over IP routing or packet filtering, as they do not have root permissions on the host.

## Configuration

Configuration of *ProVSD* is handled on a number of levels. At the top we have *VSDControl* which is used by host administrators to create and control virtual machines, as well as configure individual services and system preferences. Using this tool, it is very simple to start a new virtual machine, then set up which ever services you want on it. Anyone who has access to *VSDControl* can reconfigure any service, the user accounts and other options on all virtual machines on the host, so it's obviously important that very few people have the level of access to make use of this tool. Owners of virtual machines have a cut-down version of *VSDControl* known as *VSDClient*, which offers all of the administrative controls of *VSDControl*, but lacks options to create new virtual machines. As well as both of these stand-alone software packages, there is a web-based *VSDWeb*, which runs locally on each physical host for configuring virtual machines which are on it.

*VSDControl* and *VSDClient* allow you to control multiple virtual machines on more than one host, so if you have virtual machines distributed over many physical systems, then it's easy to keep track of what you've got set up on each of them. However, each of these is sufficiently complex that you will need an understanding of what you are doing, rather than simply jumping in and hoping for the best.

*Sendmail* is little short of a complete nightmare for experienced Linux users, much less beginners, so *ProVSD* offers *VSDMail* which is used



Through *VSDControl*, the host administrator can limit access to individual virtual machines on a per user basis.

to setup aliases, POP3 mailboxes and auto-responders. It would have been nice to see IMAP supported, but the more experienced *ProVSD* user could attempt to install it themselves.

## Normal services

*ProVSD* offers all of the popular Linux services, including *Apache*, *MySQL*, *Sendmail*, *SSH*, *ProFTPD*, along with command line tools like *Perl*, *GCC* and *Python*. There is also *PHP4*, *phpMyAdmin*, *Webalizer* and *Procmal*, which are setup from within the virtual machine, rather than via either of the config tools. You can compile your own code within the virtual machine and run it, either locally over *ssh*, or as a CGI through *Apache*. You may have problems with anything which needs to

change it's effective UID, such as *suexec*, which can't be done unless you're root on the main host.

Keeping *ProVSD* up to date is not too difficult, as most files within the *chrooted* filesystem are hard symbolic links to binaries on the host, so all you need to do is upgrade the host, and every virtual host is upgraded.

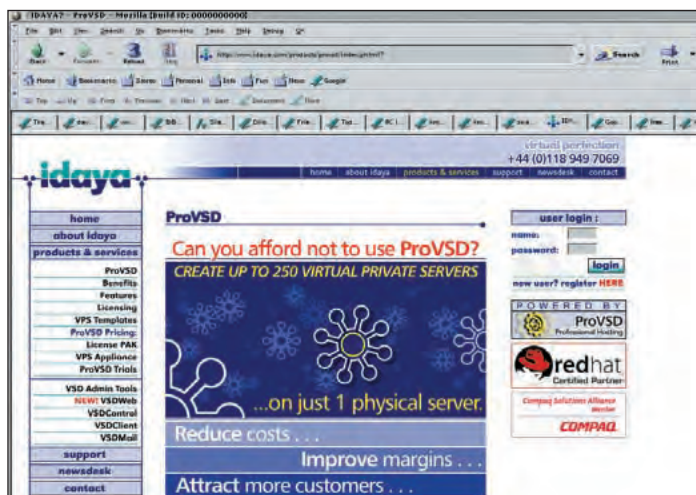
It is open to debate whether or not *ProVSD* is suitable as a security tool, as it is entirely dependent upon the *chroot* binary, which, if exploitable, can give a user access to the entire host machine and, from there, possibly gain root privileges through an exploit in another piece of software. However, *ProVSD* makes for an ideal sandbox for playing with new services or trying out the latest piece of software which might blow up in your face. This is not really anything you can't already do with a regular user account on a standard system, but being able to contain the entire "/" filesystem with *chroot* gives a little more realism to any testing and makes migration from there, to a real host, much less painful.

The nearest competition *ProVSD* has is from *User-Mode Linux*, which is a kernel patch for Linux which enables you to run an entire Linux instance within user-space, giving you far greater flexibility, although there is a fairly large overhead due to having to actually run an entire Linux kernel. At this stage, no one seems to be using *UML* for production virtual hosting services, mainly due to it's developmental status, but in the future *ProVSD* may be looking at some

serious competition from *UML*, particularly once *UML* has been ported to non-IA32 architectures, as it offers greater security and power to the virtual machine owner without impacting at all on the host.

## Conclusion

Ultimately, if you're going to be looking at *FreeVSD* for use in a production environment, you really should take a look at *ProVSD* before trying to work it all out yourself. The vastly improved installation, coupled with the GUI configuration tools, offers a far more user-friendly environment for potential customers, and makes life much easier for beginners. *ProVSD* is not cheap if you're looking at thousands of virtual machines, but if you look at the price per virtual machine, once you include the price of the physical machine, it can be quickly recouped through hosting costs. [LXF](#)



Idaya also offer Linux machines preinstalled and configured with *ProVSD*.

## LINUX Format VERDICT

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

*ProVSD* is a wonderful alternative to a data centre full of dedicated machines, offering all the capabilities of individual servers with limited overhead and a drastically reduced cost.

## LINUX Format RATING

8/10



# Designing with JavaScript

With a little help from this book **Jono Bacon** gets into web development with JavaScript.

- **AUTHORS** Nick Heinle & Bill Pena
- **PUBLISHER** O'Reilly
- **ISBN** 1-56592-360-X
- **PRICE** £24.95

**J**avaScript is a language which is frequently used in web development environments, and this book is aimed at teaching the user how to use JavaScript specifically for web development.

The book is published by O'Reilly (who are famed for their high quality Nutshell series), but it actually came as something of a surprise to me. Nearly every O'Reilly book I own is dominated with textual information with little illustrative content; *Designing with JavaScript* breaks this mould with plenty of illustrations and diagrams,

and it generally looks a visually appealing book.

The actual content in the book is not fully expansive, although what is contained here is written well and with a high degree of accuracy. The content is also well chosen; it does not provide arcane topics that will be of little use to the average web developer – each topic has a place in solving common development problems.

It was good to see some material on form validation in this book, because this is an area where JavaScript is often used. Also, the section on browser and plugin checking provides useful information. One area of the book I was little surprised to see was the section on Dynamic HTML; considering DHTML is really a combination of various technologies, this content was welcomed – along with the details on the Document Object Model.



Although the book does provide a good introduction to JavaScript and related technologies, I do feel this book would not be right for those with no previous programming experience. This volume does not cover things such as variables, data structures and input/output that may be required for any advanced JavaScript usage. The omission of such content is not a flaw in the book since it is clearly stated that it does not intend to cover so much of this kind of content and another book that does may be advisable to truly understand the full concepts of JavaScript.

In conclusion, *Designing With JavaScript* is a well written, well presented book. It provides a good range of useful content that is

applicable to common web development problems; if you are in need of a book to get you going straight away in using JavaScript to develop your site, I suggest you take a look at this one.

## Linux Format VERDICT

A well written, attractive book which provides some interesting and useful content. If you have a need for JavaScript on your sites, I recommend this book.

LinuxFormat **RATING**  
 **8/10**

# Python and XML

Is Python and XML a match made in heaven. **Nick Veitch** is the man with the answers.

- **AUTHORS** Jones & Drake
- **PUBLISHERS** O'Reilly
- **ISBN** 0 596 00128 2
- **PRICE** £28.50

**C**onjuring up a list of perfect partnerships, you might think of Morecambe & Wise, peanut butter and jam, and maybe Python and XML? As this book points out, there are some very good reasons for choosing Python as an application or middleware solution for XML tasks. Python has plenty of XML tools, is easy to use, object oriented and handles text very well. Since XML is all about human-readable text, it seems like a perfect solution. But how would you even start parsing XML documents with Python.

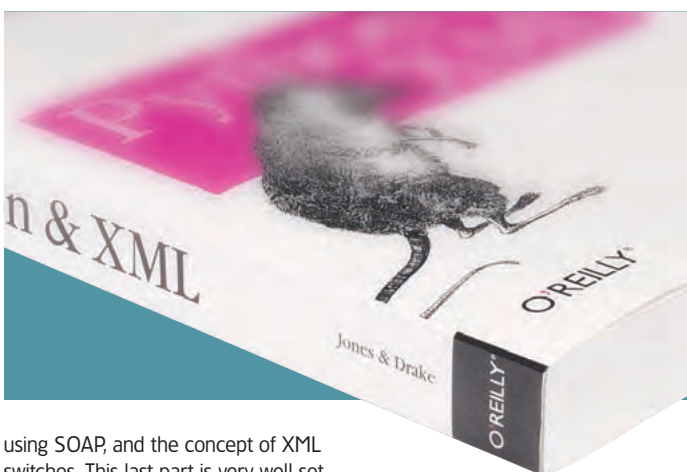
Well, fortunately there are one or two APIs for working with XML from Python. This book deals mainly with SAX, using the standard Python library

and PyXML, which has more wideranging features including Xpath, XSLT and DOM support.

The book starts with a thoroughly readable and informative introduction as to why you should use XML with Python, and a more detailed description of XML itself, and some of the things you'll have to watch out for. The book isn't aimed at python beginners, but you could quite easily pick up nearly all you need to know about XML from this work.

Subsequent chapters deal with practical applications of the combination for a variety of purposes, which starts off with simple data parsing and covers Document Object Model and web applications, XPath node searches, transformations and validation.

The last three chapters may be the most useful for most XML tasks, as they deal with constructing web/network services with Python,



using SOAP, and the concept of XML switches. This last part is very well set out, with detailed commentary of the code (which as always, is available at [www.oreilly.com](http://www.oreilly.com)), as well as suggestions for how to build this into a more complicated application.

I'll use the word 'practical' again, because that sums up the tone of this book – it's about getting things done, and shows the processes in easy to understand chunks. There aren't great swathes of waffle, but there are plenty of examples.

Appendices cover installation of necessary tools and the APIs for SAX and DOM, as well as a guide to

MSXML and other Python tools available. Even the index is more comprehensive than the usual effort.

Over 340 pages of real, practical, useful info here, well worth reading.

## Linux Format VERDICT

A useful, practical guide to XML implementation

LinuxFormat **RATING**  
 **9/10**

# Web Security, Privacy & Commerce (2nd Edition)

Can one book sum up security issues for users and developers? **Steve Heaven** finds out.

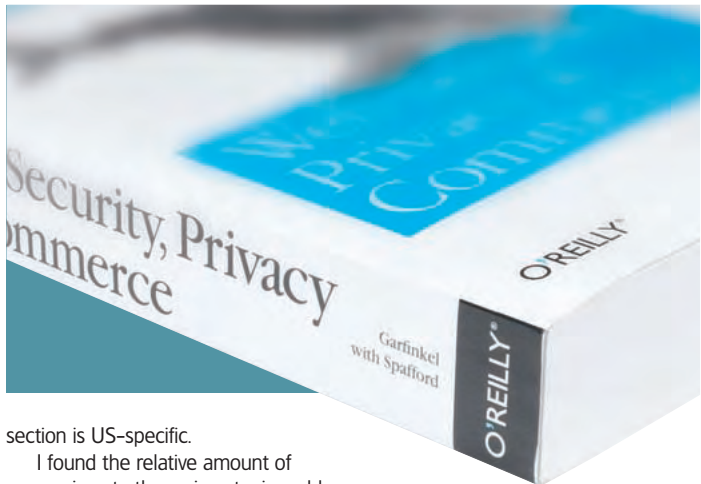
■ **AUTHORS** Simson Garfinkel  
with Gene Spafford  
■ **PUBLISHER** O'Reilly  
■ **ISBN** 0596000456  
■ **PRICE** £31.95

**A**t first glance this seems a useful book. Web security and privacy are very much in peoples' minds at the moment. The book lives up to the usual O'Reilly standards. It is well laid out, clearly written and appears well researched and authoritative. However on further reading I got the impression that the book was trying to be all things to all men, trying to cover too much material. The result of this was that none of it was covered in much depth. The book is subtitled "Security for Users, Administrators and ISPs". It would be a useful read for users, particularly those with little knowledge

of security and privacy issues. Administrators and ISPs, on the other hand, need much more in depth knowledge of the technical details than this book provides.

The book is in four sections "Web Technology", "Privacy & Security for Users", "Web Server Security" and "Security for Content Providers".

The first section explains how the Web works, DNS issues, etc., and covers cryptography and digital signatures. Section two has practical help on protecting personal privacy, including choosing a good password, blocking spam mail, etc. Section three covers matters of interest to ISPs and others running websites. This is the weakest section with most of the topics not covered in sufficient detail. The last section has useful information for those wishing to protect their published information. However much of the discussion of legal issues in this



section is US-specific.

I found the relative amount of space given to the various topics odd. For example, nearly as much space is given to discussing paper shredders as is to firewalls. For a book on Web and Internet security this is the wrong balance. Firewalls are covered in only two pages, and packet filtering and proxying is briefly mentioned but not techniques such as stateful inspection. In the section on writing secure CGI applications there is about half a page of guidance for Perl writers, but nearly ten pages for writers of PHP.

## Linux Format VERDICT

In general the book would be a useful read if you need an overall view of the topics covered. But if you need to actually implement privacy and security measures then you need something with more detail.

## LinuxFormat RATING

6/10

# XSL Essentials

Richard Drummond investigates a practical guide to doing useful things with XML documents.

■ **PUBLISHER** Wiley  
■ **AUTHOR** Mike Fitzgerald  
■ **PRICE** £33.50  
■ **ISBN** 0471416207

**X**SL Essentials is part of Wiley's XML series and covers the two XSL-derived languages, XSLT (XSL Transformations) and XSLFO (XSL Formatting Objects).

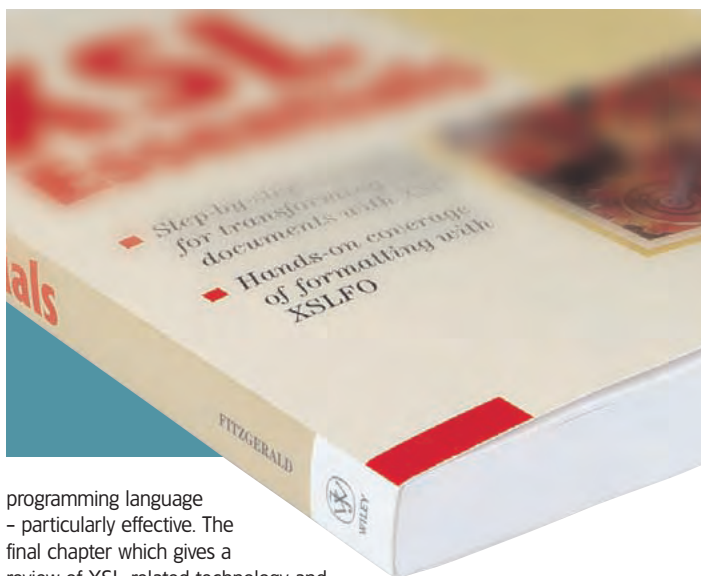
XSLT is the XML-based language for transforming XML documents into other formats, including XML and HTML, while XSLFO, is used for preparing XML documents for print. The former language takes up the lion's share of the book, with only two chapters devoted to XSLFO. The XSLFO specification is still unfinished, so this is an appropriate balance.

This is a practical book and right from the start gets the reader using XSL. It assumes little prior knowledge from the reader, but experience of at

least HTML would be beneficial. Chapter one introduces a simple XSL transform, and shows you how to apply it to an example document with the various XSL processors available.

The following chapters cover the basics of XSLT including how to transform XML and XHTML documents into text, HTML, XHTML and XML (using a different XML vocabulary). More advanced topics are gradually introduced in later chapters, including using XSLT to sort and number XML-formatted data, how to do conditional processing in XSLT and how to use variables.

The author adopts a clear and readable style throughout and the topics covered are illustrated well with plenty of example code (all of which is available on the CD-ROM that accompanies the book). I found the sections on conditional processing and variables and parameters in XSLT – where the author explains these by comparing XSLT to a traditional



programming language – particularly effective. The final chapter which gives a review of XSL-related technology and terminology is also useful.

XSL Essentials is an excellent introduction to XSL. However, I feel better use could have been made of the CD-ROM (for example, perhaps XSL processors such as GNOME's libxslt and Apache's Xalan could have been included as well) and more reference material would be a welcome addition. Also, the book's back jacket makes a claim that SOAP and B2B documents are covered, but

while this true, these topics are dealt with only very lightly. **LXF**

## Linux Format VERDICT

A clearly-written and practical introduction to all things XSL for the beginner.

## LinuxFormat RATING

8/10



# Roundup >>

Every month we compare tons of software, so you don't have to!



## Our selection at a glance

- The Saint
- Tripwire
- PortSentry
- OpenSSH
- Bastille
- Encrypted filesystems

## Security tools

**Charlie Stross** take a look at tools for keeping the script kiddies at bay and tightening up network security on Linux.

**T**hese days, security problems have become more of a threat to our freedom to enjoy a peaceful computing life than hardware failure or buggy software.

The usefulness of a computer is proportional to the data you can access from it, and there's no bigger source of data than the Internet – nor any bigger source of security

problems at the same time.

In this roundup we're going to look at security from the Linux point of view. This is fundamentally different from the Windows perspective on

security, for several reasons. On the one hand, it's almost impossible to create a viable virus that infects Linux machines – viruses have to replicate in order to spread, and Linux (with its compartmentalization of user IDs under which programs are executed) puts up natural barriers to the spread of viruses. But on the other hand, because Linux systems tend to run lots of server programs, they're vulnerable to flaws in network protocols. In the worst case, these flaws can allow an attacker to obtain a login shell running as root – in which case, you can lose control of your machine even more comprehensively than a Windows user who's had a copy of *Back Orifice* planted on their machine.

First, we're going to look at information sources on Linux security – books you need to read and mailing lists you need to subscribe to. Then we're going to examine some tools for securing access to your machine via the internet: *ssh* (for connecting to remote Linux systems securely); *PortSentry* (for detecting network attacks in progress); *tripwire* (for detecting intruders after they get in); hardening scripts (for making your system resistant to attackers); encrypted filesystems (for protecting your data); and *Saint* (for checking your system for network vulnerabilities).

## Read Me First!

And learn to *think* security

Security, as Bruce Schneier (CEO of Counterpane Systems) observed, is not a goal but a process. It's also a way of thinking. If you take a stock Linux server and connect it to the net it may be cracked within a matter of hours – the characteristics of the major linux distros are well known, and crackers have written automated tools to scan network addresses and identify machines with known security holes. Even if you apply all your vendor's updates and security patches, you may be vulnerable to a new crack that hasn't been publicized yet. The only way to achieve security is to know what you're doing and keep an eye out for stormy weather ahead.

If you know nothing about Linux security and only have time for one book, you should start with *Practical UNIX and Internet Security, 2nd Edition* by Gene Spafford and Simson Garfinkel (O'Reilly and Associates, ISBN 1-56592-148-8). It's a fat book, but it's accessible and it gives you a solid grounding in the basics – how the security mechanisms built into Linux (or other Unixes) work, what you can expect of them, and, more importantly, how your Linux system may come under attack when it's connected to the Internet, and how to deal with an intrusion. Like all books it is necessarily a bit out of date, and doesn't cover the latest issues (such as wireless security),

but it provides the vital background which anyone running a Linux box connected to the Internet needs to be aware of. Ignore it at your peril.

Going beyond books, you need to check your vendor's website for security alerts on a regular basis. If you run Red Hat, the URL to bookmark is <http://www.redhat.com/apps/support/errata/>. If you run SuSE you should go to <http://www.suse.co.uk/uk/support/download/updates/index.html>. For other vendors, check their websites. There is usually a mailing list you can join to be notified of bugfixes and security alerts; join it. You can also find weekly digests of security announcements from all

major vendors via *Linux Weekly News* (<http://lwn.net/>), but in event of a major security issue surfacing, you need to know about it within hours, not a week later (after you've been hacked).

Vendors take time to release patches. If you're in a real hurry, go to [www.securityfocus.com](http://www.securityfocus.com), and subscribe to the *BugTraq* mailing list – where new security vulnerabilities are notified; it's how the Linux distributors often first hear about problems. It won't always tell you how to fix a security hole, but at the very least it'll tell you which service is vulnerable so that you can temporarily disable it until a patch arrives. (Warning: *bugtraq* tends to be a busy mailing list).

# The SAINT

## Probing for vulnerability.

■ **VERSION** 3.2 ■ **URL:** <http://www.wwdsi.com/saint/index.html>

**Linux distributions tend to come** with a whole raft of network servers running on them. Different servers have different vulnerabilities, and out-of-date distributions are the number one source of machines that are open to attack. How do you tell if your machine has a hidden weakness?

*SAINT* (formerly known as *SATAN*) is the *Security Administrator's Integrated Network Tool*; it gathers as much information about remote hosts and networks as possible by examining network services (including *finger*, *NFS*, *NIS*, *ftp* and *tftp*, *rexed*, and *statd*) and checking for potential security flaws –

usually in the form of mis-configured network services; well known bugs in system or network utilities; or poor or ignorant policy decisions. It can then either report on this data or use a simple rule-based system to investigate any security problems. In exploratory mode, it will examine the avenues of trust and dependency between several hosts on a network and iterate further data collection runs over secondary hosts, to see if it's possible to compromise an entire network rather than just a single machine.

*SAINT* is distributed under a licence similar to the GPL, except that it forbids commercial resale of the tool. You can download it from <http://www.wwdsi.com/saint/>. RPMs are available for x86 Linux, but compiling it on other Unix systems is easy enough. When you run *SAINT*, you have a choice of modes – either as a command line tool that reports on the vulnerability of the target hosts, or (more usefully) with an interface presented via a web browser. This lets you configure *SAINT* interactively, choose target machines and tests to run, and view the results of a scan.

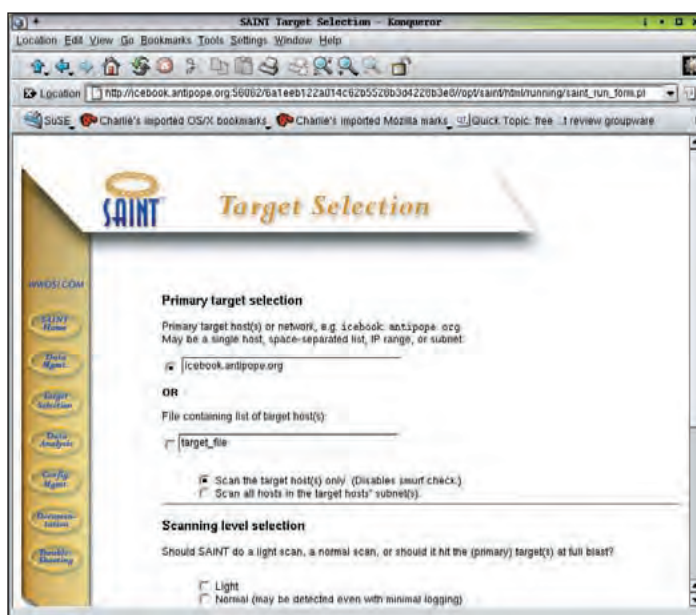
Rather than just probing your local machine, *SAINT* can identify all the hosts on a LAN and probe them in series. It can also locate gateways and probe hosts on the other side, such as firewalls or routers. You can tell *SAINT* to use *nmap* (for TCP/IP vulnerability probing) and *Samba* (for NetBIOS attacks), and look for signs of previously successful attacks (such as installed root kits and trojans), as well as vulnerable servers.

*SAINT* is fast, efficient, and very good indeed at identifying vulnerabilities; in particular, it highlights common security exploits and, if you shell out for the commercial version, helps you fix them. This is a must-have if you run a network of your own. One word of warning: because *SAINT* can explore neighbouring hosts, you run the risk of really annoying your neighbours if they don't know you're running it. A *SAINT* scan can be mistaken for an attack by crackers. Read the manual carefully before you turn it loose on anything except a single machine that you own!

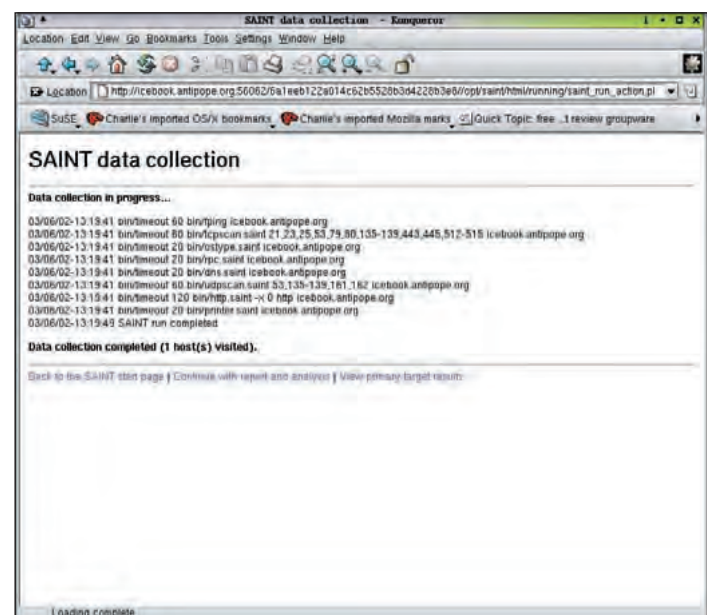
A second note of caution: the information *SAINT* reports is highly technical and won't make much sense to you unless you have already done the basic background reading to understand network security issues. This isn't a hand-holding solution to all your ills – it's a power tool, and like all power tools you need to understand what it's for and how to use it before you pick it up.



Saint uses a 100% web-based interface. The data analysis page shows a vulnerability, revealed in the results of a *SAINT*-scan of a local machine.



*SAINT*'s target selection page lets you pick hosts or entire networks to probe for vulnerabilities.



In data collection mode, *SAINT* shows you what it's doing, as it's doing it. It's fairly scary, if you know what it all means...

## LINUX Format VERDICT

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

The security vulnerability scanner *par excellence*, allows even relative novices to assess the vulnerability of their network.

## LINUX Format RATING

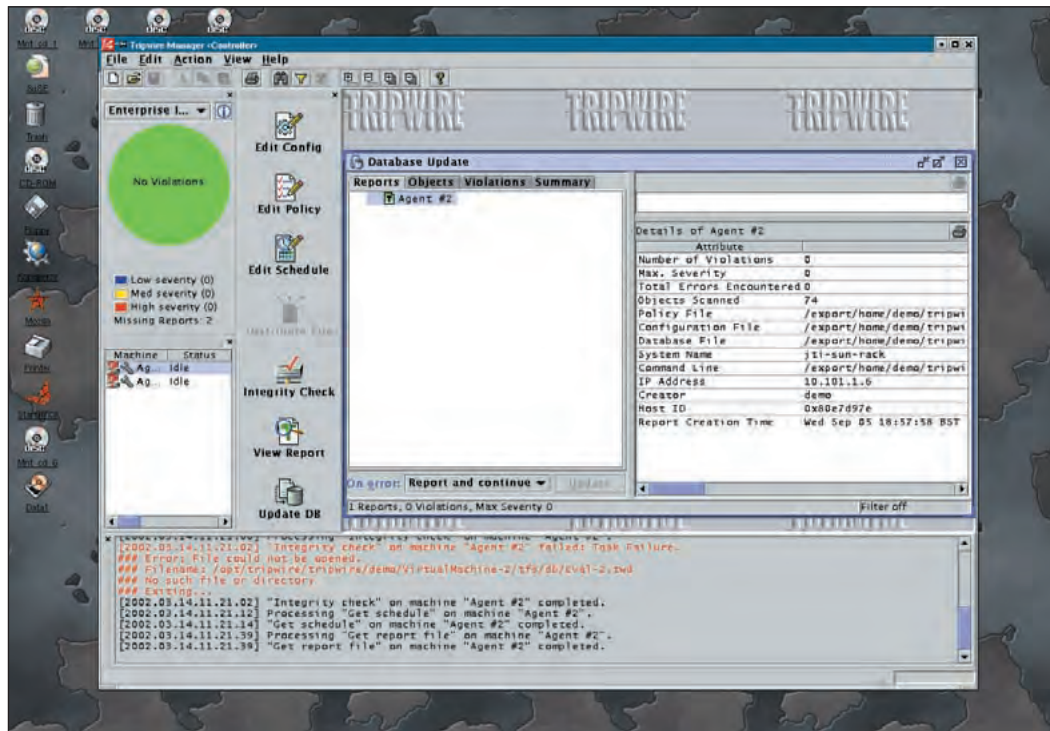
8/10



# Tripwire

The burglar alarm.

■ **VERSION** 2.3.1 ■ **WEB** <http://sourceforge.net/projects/tripwire/>



**Tripwire Manager**, a payware product from Tripwire Inc., lets you manage a whole bunch of remote tripwire servers on a network and visually track the state of files on your network – including spotting unauthorised changes. It's the equivalent of one of those centralized burglar alarm monitoring services.

**SAINT** can tell you if you have open ports, and **PortSentry** can station a guard behind them with a cosh, but how do you know if the guard's been overpowered and the burglars are rifling your safe? Or in other words a cracker has installed a root kit on your machine – corrupted programs (such as *ls*, *ps* and *top*) that won't display their home directory and password-gathering tools, and a bunch of items like a copy of *bash* with set-UID root permissions. With this kind of kit a cracker can make themselves at home – they've got a root shell to use whenever they please, and you can search for it high and low but your tools won't let you see it.

The answer is to install a tool like **tripwire**, from Tripwire Security ([www.tripwire.com](http://www.tripwire.com)), or the open source package (from <http://sourceforge.net/projects/tripwire/>).

The concept of **tripwire** is simple. When you install it – before you connect your computer to the network – **tripwire** scans the filesystem, creating a list of MD5 checksums (fingerprints) for every file it finds. You burn this onto a CD or write-protected

floppy, and it serves as a reference of the state of your system when you first set it up. Thereafter, every few hours or days, **tripwire** re-scans your filesystem and generates checksums, which it compares to the master copy. If the checksum for a file differs from the copy on file, then the file has been modified. By following directives in a configuration file that identifies critical security-related files, **tripwire** can warn you if, say, a cracker has replaced */bin/login* with a Trojan.

Like most of the other tools in this roundup, **tripwire** is resolutely non-graphical. You run it from a *crontab* file, typically daily, to check the filesystem against the master database. (You can also run **tripwire** interactively if you're suspicious about something.) Sooner or later you will make changes to config files in */etc*, or install binaries in */sbin*; when you do so, you can merge these into the master database (if you've mounted it on writeable media for that purpose) by running **tripwire** interactively in update mode. It prompts you with each file that has changed, asking you whether or not to accept the changes into the database.

One risk with **tripwire** is that a clueful cracker will see it's in place and disable it. A more subtle risk is that they'll subvert your machine so thoroughly that the device node your unwritable **tripwire** database is mounted on will point to somewhere else – but they can't stop you uploading a fresh **tripwire** binary, generating a database of checksums for your machine, then downloading it to a secure workstation and comparing it to a copy of the database that you keep in a safe.

## LINUX Format VERDICT

Installation	6/10
Documentation	8/10
Features	9/10
Ease of use	9/10

**Tripwire** will tell you if your security has failed and someone has broken in. Only trouble is, you need to know what all those files it's yelling about are for.

## LINUX Format RATING

9/10

# Port Sentry

Detecting probes and cutting them dead.

■ **VERSION** 1.1  
 ■ **WEB** <http://www.psonic.com/products/>

**PortSentry** (available along with companion products *HostSentry* and *LogSentry*, from Psionic Technologies: <http://www.psonic.com/products/>) is a tool for detecting unauthorised intrusion attempts. The package was developed by the Abacus Project, as part of a set of low-maintenance intrusion detection tools; an enhanced commercial version may be available later, but it is currently distributed under a free-for-non-redistribution licence.

Installation is relatively straightforward, but requires editing some files using a text editor, then running a *Makefile*; you should start by reading the *README.install* file carefully, and you need to have a vague idea how TCP wrappers work if you're going to install **PortSentry** on an existing system. (Debian packaged versions exist, as do RPMs.) Configuring **PortSentry** is harder: while the documentation is good, it assumes a knowledge of TCP/IP networking and some familiarity with routing and packet filtering commands. Be careful!

**PortSentry** is a network daemon that binds to a whole bunch of TCP/IP ports and monitors them. It can detect portscans (attempts to connect to consecutive ports to identify what services your machine is running); some stealth scans (portscans that are normally ignored by the operating system); and both TCP and UDP scans. It's smart enough not to mistake FTP connections for an attack, and it can log its status information via the *syslog* daemon (which can, if necessary, be directed to a secure logging host elsewhere on your network).

Among other things, **PortSentry** can be configured so that it not only logs attempts to scan; it adds a default route to the bit bucket for all packets from an attacker, or it uses your packet filtering

**Server log from a run of PortSentry. Note the bit at the end where it detects a scan and locks out the offending host!**

system (*ipfwadm* or *ipchains*) to reject packets.

I set up *PortSentry* on my gateway machine, *ssh'd* in to a colocated server, and tried to *telnet*, in succession, to the gateway machine on port 6000, then port 6001. Bang! The *ssh* session froze – I'd been locked out, as the following logfile shows:

```
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert:
PortSentry is now active and listening.
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert:
Connect from host:
raq981.uk2net.com/213.239.59.58
to TCP port: 4001
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert: Host
213.239.59.58 has been blocked via wrappers with string:
"ALL: 213.239.59.58"
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert: Host
213.239.59.58 has been blocked via dropped route using command:
"/sbin/route add -host 213.239.59.58 reject"
```

This sort of reactive defense is so good that it ought to be standard in all Linux distributions, out of the box – the problem is the restrictive licence on *PortSentry*, which prevents modification or redistribution without permission. If you can live with the licence terms and have a broadband connection, you're strongly advised to download *PortSentry* – if set up properly, it can warn you of attacks in progress and automatically reconfigure your system to reject all connection attempts by an attacker, locking them out.

The best feature is its ability to bar connection attempts by someone who seems to be port scanning you – don't install it until after you've run *SAINT*!

## LINUX Format VERDICT

Installation	5/10
Documentation	6/10
Features	7/10
Ease of use	10/10

Initially difficult to set up and requires some knowledge of networking, but it's the equivalent of stationing a guard behind your firewall.

## LINUX Format RATING

8/10

```
Mar 7 13:05:39 charlie17 portsentry[26988]: adminalert: Psionic PortSentry 1.1 is starting.
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced mode will monitor first 1024 ports
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced mode will manually exclude port: 113
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced mode will manually exclude port: 139
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 9
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 13
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 22
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 37
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 111
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 113
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 515
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: Advanced Stealth scan detection mode activated. Ignored TCP port: 139
Mar 7 13:05:39 charlie17 portsentry[26989]: adminalert: PortSentry is now active and listening.
Mar 7 13:05:42 charlie17 portsentry[26990]: adminalert: Psionic PortSentry 1.1 is starting.
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 1
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 7
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 9
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: ERROR: could not bind TCP socket: 9. Attempting to continue
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 11
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 15
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 70
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 79
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 80
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 109
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 110
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 111
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: ERROR: could not bind TCP socket: 111. Attempting to continue
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 119
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 138
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 139
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 143
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 512
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 513
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 514
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 515
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: ERROR: could not bind TCP socket: 515. Attempting to continue
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 540
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 635
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 1080
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 1524
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 2000
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 2001
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 4000
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 4001
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 5742
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 6000
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 6001
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 6667
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 12345
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 12346
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 20034
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 27665
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 30303
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 32771
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 32772
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 32773
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 32774
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 31337
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 40421
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 40425
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 49724
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: Going into listen mode on TCP port: 54320
Mar 7 13:05:42 charlie17 portsentry[26991]: adminalert: PortSentry is now active and listening.
Mar 7 13:13:22 charlie17 sshd[126998]: log: Connection from 192.168.1.14 port 33921
Mar 7 13:13:24 charlie17 sshd[126998]: log: Password authentication for charlie accepted.
Mar 7 13:13:26 charlie17 su[27001]: + ttyp2 charlie-root
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert: Connect from host: raq981.uk2net.com/213.239.59.58 to TCP port: 4001
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert: Host 213.239.59.58 has been blocked via wrappers with string: "ALL:
213.239.59.58"
Mar 7 13:13:43 charlie17 portsentry[26991]: attackalert: Host 213.239.59.58 has been blocked via dropped route using command:
"/sbin/route add -host 213.239.59.58 reject"
Mar 7 13:16:14 charlie17 sshd[127008]: log: Connection from 192.168.1.14 port 33923
Mar 7 13:16:19 charlie17 sshd[127008]: log: Password authentication for charlie accepted.
Mar 7 13:16:19 charlie17 sshd[127010]: log: executing remote command as user charlie
```



# Ssh

## It's a secret

■ **VERSION** 3.x ■ **WEB** <http://www.openssh.com/>

### Ssh should need no introduction.

If you work on remote Linux machines, you're probably using *telnet* to connect to them. Don't do that: install *ssh* and the *ssh* server instead. *Ssh* is the *secure shell* – when you point an *ssh* client at an *ssh* server, it provides an encryption layer for your terminal session. This is really important in the world of wireless networking, because anything you type in *telnet* is passed to the server unencrypted – including your passwords.

The original *ssh* kit is commercial software, but the OpenSSH project have released a free clone of it; you

can get it from [www.openssh.com](http://www.openssh.com). There are two components – the server and the client. You need to install the server (*sshd* – the *ssh daemon*) on any machine you intend to log into remotely, then use one of the client tools to connect.

In addition to the *ssh* client itself (which gives you a remote interactive shell on the server, with encryption) there's *scp*, *Secure Copy*; this lets you copy files from one machine to another, with encryption, by typing commands like:

```
scp myfile.txt
charlie@server:/tmp/
```

```
myfile-incoming.txt
```

or:

```
scp -r some_directory
charlie@server:/home/charlie/
some_directory/
```

Recent versions add *sftp* – a secure version of FTP that allows you to do interactive batch file transfers. The venerable *rsync* file synchronization tool can use *ssh* as a transport layer, too – as indeed can the X Window System.

If you work with a graphical desktop and don't like typing commands to copy files, never fear: there's an IOSlave for KDE that lets you use *ssh* to connect securely to a remote server. (If you're running KDE 2.x, go to <http://ich.bin.kein.hoschi.de/fish/> to download and install **kio\_fish**, which lets you manipulate files on *sshd*-equipped machines securely. KDE 3.0 will come with a different IOSlave, which uses *sftp* instead.) *Ssh*

should also be supported by the *GNOME* VFS layer before long.

One point to remember: *ssh* is cryptographic software and it relies on using RSA keys for authentication. Read the manual before you install an *ssh* server, so you know how to copy the public keys to the clients you will be logging in from.

### LINUX Format VERDICT

Installation	6/10
Documentation	7/10
Features	10/10
Ease of use	8/10

*Telnet* isn't secure. *Ssh* is. Use it, or lose your password to the first cracker on your network with a packet sniffer.

LINUX Format **RATING**  
 **10/10**

# Bastille

## Hardening scripts: storming the Bastille

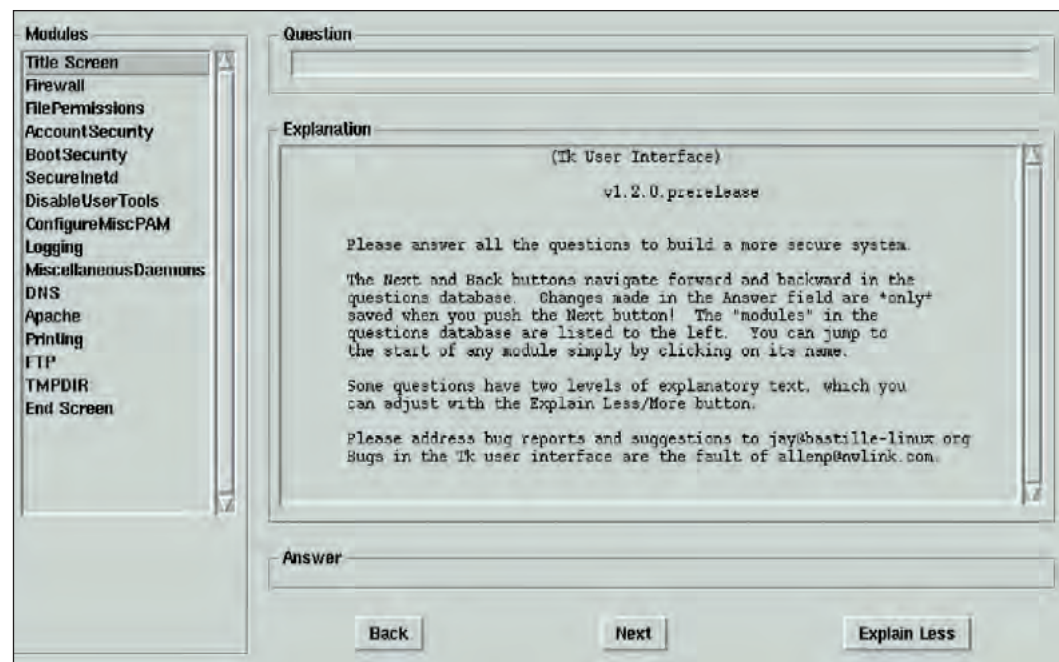
■ **VERSION** 1.3.0  
 ■ **WEB** <http://www.bastille-linux.org/>

### The first law of computer security

is that security is inversely proportional to convenience. It follows that companies who want to sell lots of copies of a software package try to make it convenient – and in doing so, sacrifice security. Any Linux system can be made arbitrarily secure, but it's inconvenient to use, so most commercial distributions (Red Hat, Mandrake, SuSE, and Caldera are all guilty) provide a default configuration that is easy to use – and insecure.

The *Bastille* hardening system (from <http://www.bastille-linux.org/>) is a set of scripts that attempt to “harden” a Linux distribution. It currently supports Red Hat and Mandrake systems, with support planned for Debian, SuSE, TurboLinux and HP-UX.

When you install *Bastille*, you get a bundle of goodies; a script (launched from early on in the *rc* scripts that run at boot time) that sets up a firewall using *ipchains* or *iptables*, and a graphical front-end (using either *Tk* or *Curses*) that lets you configure the features of the *Bastille* kit interactively (as an alternative to using the *AutomatedBastille* script to tighten security automatically).



**Bastille Linux** is a hardening system for Mandrake and Red Hat: here we see its graphical configuration manager prompting for the answers to some hard questions.

*Bastille* automates a bunch of common security tasks. It can keep an eye on temporary directories, where many applications write files in unsafe ways; it can disable all sorts of web- and email-related services (unhelpfully started by default on many Linux distributions), prevent anonymous downloads, enforce password ageing and security, disable the (old and insecure) *r*-protocols, and much more. This is all stuff that an experienced administrator can do by hand – but having it scripted for beginners, with online explanations of what it's trying to do (in the interactive version) is

useful. If you don't have a lot of experience of hardening a system, *Bastille* is a good place to start; the only *caveat* is that it tends to lag slightly behind new releases of each distribution, so that if you insist on always upgrading to the latest release you may lose the protection it confers.

The only real problem with *Bastille* is portability. Rather than aiming at the Linux FHS (Filesystem Hierarchy Standard), the developers have targeted specific distributions. This means that if you run, say, SuSE or Slackware and want it now, you're out of luck. Ditto Darwin/MacOS X users.

### LINUX Format VERDICT

Installation	9/10
Documentation	9/10
Features	8/10
Ease of use	9/10

If you use no other security measures, use *Bastille*. It'll install firewalling rules, tighten up your security, and, best of all, tell you what it's doing and why.

LINUX Format **RATING**  
 **9/10**





# HotPicks

The best new open source software on the planet!



**Maurice Kelly**

Busy coder, electronic engineer and Midnight Oil fan

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 coverdisc, but we've provided web links if you want to make sure you have the very latest version.

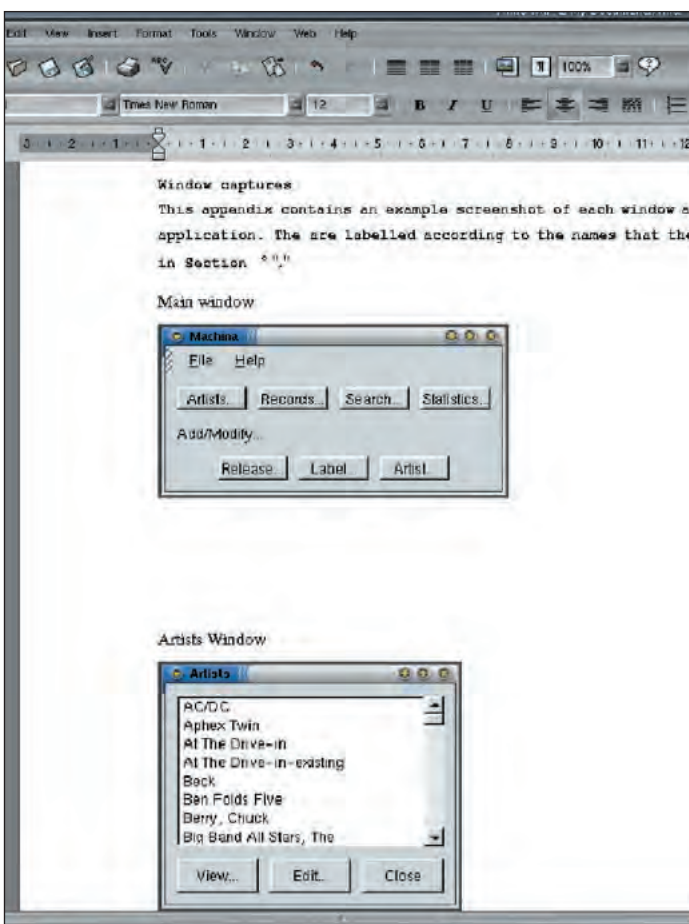
If you have any suggestions for things that we should cover, email us at [linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk)

## HotPicks at a glance

Abiword	38
Bomber Instinct	39
Acidlaunch	39
GTablature	40
Gentoo	41
PhpMyAdmin	41
Sylpheed	42

## HotPicks 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!



Abiword can handle Word documents but the results vary according to document complexity.

## WORD PROCESSOR

# AbiWord

■ VERSION 0.99.2 ■ WEB <http://www.abisource.com/>

Sun recently announced that its productivity suite *StarOffice* would only be made freely available to users of its own Solaris operating system. As the major competitor to Microsoft *Office* on Linux, this is obviously a cause for great dismay amongst many Linux users (and Windows users too, but who cares about them). As a result, people are thinking more about those applications which can be used instead of having to pay for *StarOffice*.

When it comes to word processors, *AbiWord* is one of the longest-running

and strongest contenders. It has been about for a while now and is slowly but surely becoming a top quality product as it nears the inevitable 1.0 release. Like *Gnumeric* (a HotPick a couple of issues back,) *AbiWord* has been a more than usable application for quite a while, consistently improving in terms of features and stability for release after release.

*AbiWord* comes in a couple of flavours for Linux – *GTK+* and *GNOME* versions. There isn't much difference between them, but the *GTK+* version means you can run *AbiWord* without

having to install the extra overhead of *GNOME*. In operation it is very much a Microsoft *Word* clone. It has a fairly full feature set – it will never have the number of features that the likes of *Word* contains, but most of those features are not really needed by the majority of those who seek to use a word processor.

We all know by now that one of the most important features of any word processor in today's environment is file compatibility with *Word*. This is especially the case for business users, where *Word* is a *de facto* standard. Thankfully *AbiWord* has support for the import of *Word* docs, but the more complex the document (and the more recent a version of *Word* it was created with) the less likely it is to load correctly. The *Word* file format will always inspire a game of continuous catch-up. It also has import support for some *WordPerfect* documents, plus the usual text and RTF documents.

We found stability to be excellent, and experienced not one crash in the entire time the application was in use. It also runs very quickly – loading times were impressive (especially compared to the behemoth that is *StarOffice*!). Configuration options aren't particularly deep, but there's enough there to set up *AbiWord* to your personal taste, and there's also the ability to save a number of preference setups.

*Abiword* offers a number of helpful extras to the basic word processor functions, such as symbol entry and autotext entries for commonly used pieces of text. As well as these there is also a plugin framework which allows for extensibility of the program if you're a keen programmer (or the ability to download other peoples' plugins if you are lazy.) While these hint at possible feature-creep in future versions of *Abiword*, at present they are just plain handy.

Overall, *AbiWord* is a top application – at present it might not be the best solution for someone dealing with complex *Word* documents on a daily basis (especially as it doesn't actually save out *Word* docs), but if you're in the market for a fast and reliable word processor for your own native Linux work, then you can't go far wrong with this one.

## MULTIPLAYER ARCADE GAME

# Bomber Instinct

■ **VERSION 0.8.8** ■ **WEB** [http://harpe.dptmaths.ens-cachan.fr/U/gesbert/index\\_en.html](http://harpe.dptmaths.ens-cachan.fr/U/gesbert/index_en.html)

Every LXF HotPicks we find one more SDL-based game for you to waste glorious amounts of time on. The SDL library has accelerated Linux games development, and has resulted in a rash of clones of classic old games. These programmers aren't interested in developing a *Metal Gear Solid* or *Grand Theft Auto 3* for Linux – they just want to bring back classics for which many people keep around antiquated hardware and software to play anyway.

*Bomber Instinct* is, of course, a clone of the venerable *Bomberman* games. You are presented with a top-down view of a "map" which is littered with obstacles and occasional pick-ups. The only way to get rid of said obstacles is to blow them to pieces, using the bombs of which you have an unlimited supply.

*Bomber Instinct* is a multiplayer-only game, so not only do the bombs destroy obstacles, but they also make mincemeat of your opponents. And if there's one thing that people everywhere love, it's the ability to violently blow up their mates – it doesn't matter if you do it in glorious 3D in the latest console smash, or if you do it



Crowd your mates round the keyboard for some *Bomberman*-style action.

crowded round a PC keyboard.

Like most SDL-based games, *Bomber Instinct* looks exceedingly slick. The game opens with some nausea-inducing swirly effects and some sort of game intro – you can safely ignore this with the **Esc** key. Once you get to the main menu you can appreciate the little touches, such as the bouncing pumpkin used to control the options. The in-game sound effects are nice, without being

overbearing, and there is some nice cheesy intro music – these sorts of games aren't made for fantastic musical scores.

There are a number of configuration options available, such as language, full-screen mode (which is nice but can make the game a little blocky on large screens), and controls for how the game plays. It is possible to define keys for up to eight players, but you would be hard pushed to get

that many successfully crowded round that average PC keyboard, and the maps aren't all that big.

There is an initial problem with this game in that the manual is actually in French, and so you are pretty much on your own when it comes to figuring out what you are doing. The first screen when you start the game is somewhat confusing, but it turns out to be character selection, which is quite nicely done.

The game itself features a number of levels that serve as the battlefield. Again, because of the language barrier, it is not immediately apparent what the aim of the game is, but I'm satisfied to abuse the opportunity to simply wreak havoc against some friends. To make things more interesting there are occasional collectibles which appear when rocks get blown up, and carefully placed teleports and one way squares make life slightly more complicated.

Like most of the re-hashed classics that are being churned out under the SDL libraries, *Bomber Instinct* is not going to win awards for being the best game of the year. It would be nice to have more the ability to play against multiple computer opponents, and possibly network play to reduce the need for so many fingers pounding one keyboard, but for the most part this game makes for some great, mindless entertainment.

## APPLICATION LAUNCHER

# Acidlaunch

■ **VERSION 0.5**

■ **WEB** <http://linuxgamers.net/index.php?category=acidlaunch>

Minimalist is probably the best way to describe my computer desktop. I dislike desktop icons and prefer to use quicklaunch icons, and, for less common apps, resort to menus. I've switched to the *Fluxbox* window manager, after reviewing it last month, and am more than pleased with its performance. However I want the ability to launch my favourite apps without having to go through the *Fluxbox* menus every time.

The solution is *acidlaunch*. This incredibly useful little app can be kicked off however you wish to, and positioned on your desktop. Its default

appearance is that of a small vertical bar with a double row of icons – each icon when clicked will cause an application to be started.

Configuration of *acidlaunch* is simple. It creates a ".acidlaunch" dir in your home directory, which contains the XML config file. In config.xml the main config options appear within the **<launcher>** tag, and include settings such as launcher orientation, button size, and the number of rows of buttons to be displayed.

*Acidlaunch* supports the concept of "tabs", allowing you to create multiple occurrences of the launch bar, which



There's not a lot to look at with *acidlaunch*, nonetheless it's a handy tool.

can be switched between to provide a larger range of apps whilst minimising desktop usage. As such, each group of application configurations are grouped within an XML tag called **<tab>**. Each tab can be named for convenience and they can be cycled through using the **A** and **Z** keys when the mouse pointer is over *acidlaunch*.

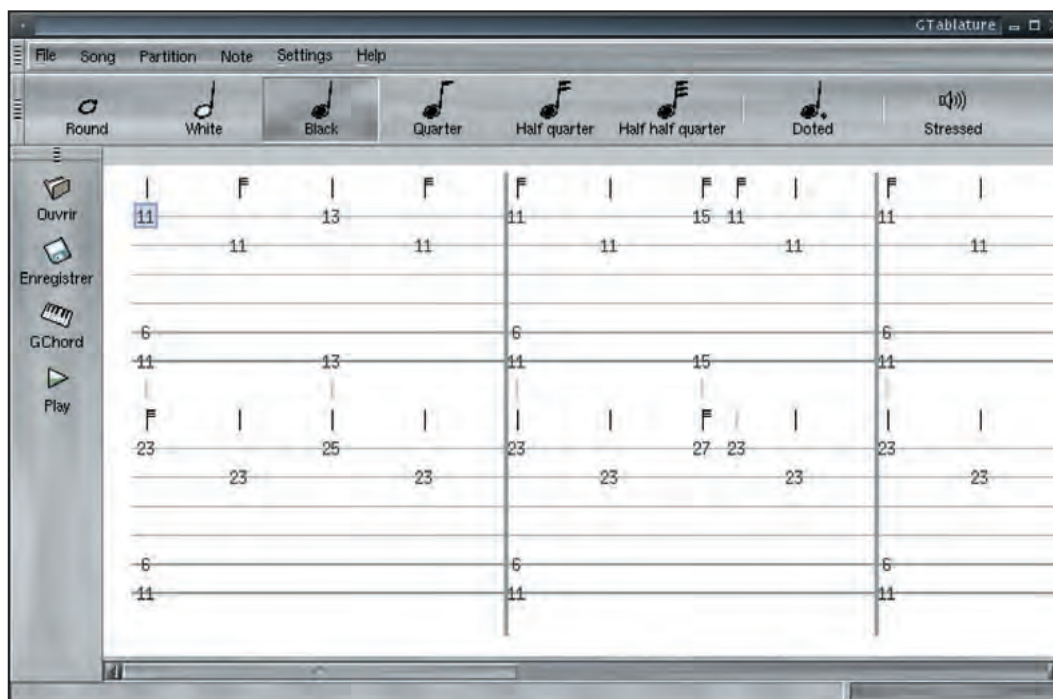
Within the **<tab>** tag appears the actual button configuration. The format requires the app name, a path to an appropriate icon, and the actual command line required to execute the application, thus:

```
<appicon>
```

```
<name>Eterm</name>
<icon>/usr/share/pixmaps/
gnome-eterm.png</icon>
<command>Eterm</command>
</appicon>
```

If you're a *Blackbox* (or *Fluxbox*!) user it is also possible to specify the tag **<withdrawn>true</withdrawn>** as part of your config file. This will cause *acidlaunch* to be sucked into the *Blackbox* slit, which allows for better integration with a *Blackbox*-based desktop. Throw in the ability to drag and drop files onto application icons to open the file, and you have a very well integrated launcher application.





Guitar tablature can look a bit boring if you have no use for it, but trust us – this is great!

## TAB NOTATION UTILITY

# GTablature

■ VERSION 0.1 ■ WEB <http://oomadness.tuxfamily.org/gtablature/en/>

At some stage – if you have ever tried to learn a song for guitar – it is highly likely that you will have come across guitar tablature. Tablature (or tab for short) is a means of representing the notes which are played on the guitar, generally using a system of numbers on a diagram of the guitar strings.

While there are musical notation systems out there for Linux, the author of *GTablature* has identified a niche for guitar tab software. *GTablature* is written using Python, with *GNOME* binding providing the user interface. As Python is a scripting language it doesn't need to be compiled on your system – simply extract the source tarball and run the file contained within. You will need a number of libraries installed, such as *PyGNOME* and *libGlade* for Python.

*GTablature* is pretty straightforward to use – the first time it is run a configuration wizard is displayed. Actually, one dialog is displayed to allow configuration of the MIDI playback software – the only other screen in the wizard is a place-holder for future configuration of printing options. Once you've chosen your MIDI software and finished the wizard you get deposited at a blank

canvas into which you can start entering your tab.

*GTablature* uses the concept of partitions to represent tabs – this means you can use a number of partitions in order to implement a single song. The partitions are displayed as six parallel lines to represent the strings, with an indicator at the appropriate points to mark out the bars of the song (the rhythm of the song can be changed in order to fit more or less notes per bar.)

Clicking the mouse on tab lines causes a small blue cursor to appear. This is your note entry box – it can be moved around using the arrow keys in order to change strings and move along the timeline. Typing in a number will cause a note corresponding to that played at the numbered fret of the guitar on the appropriate string. The note is played in real time so that you can hear exactly how it sounds. A side toolbar offers a "Play" button which will cause the entire tab to be played back to you via your MIDI player, allowing you to hear exactly how your tab is shaping up. It can also be played in a loop mode, which can be useful if used sparingly.

The main toolbar for the

application contains a number of note types which can be added to the tab. These range from notes lasting an entire bar to 16ths of a note – it's been a while since I've studied musical theory. This allows for fine placement of notes, causing more fluid and realistic sounding music. Once a note has been placed, you can place the cursor over it, and select its properties from the "Note" menu. It is possible to edit the pitch, duration and volume, which again provides for a more realistic sound that mimics true guitar playing rather than computer generated sounds.

Each of the partitions can be assigned a variety of MIDI instruments – you can even make your tab play as a set of drums if you really want.

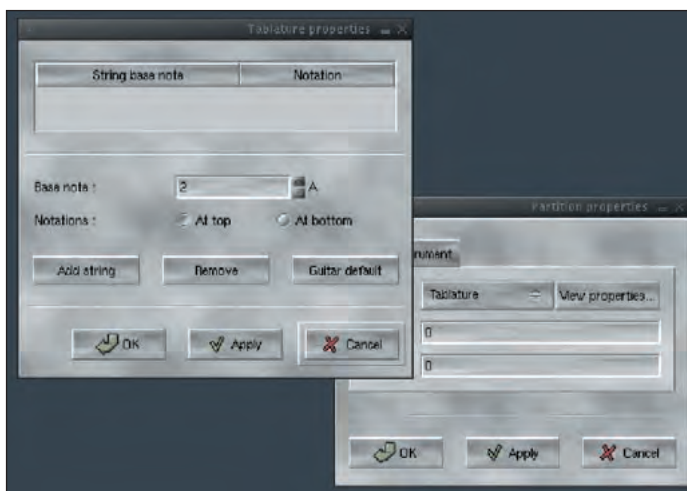
Whilst this has the potential to be a great feature, it is limited by the range of instruments you have available on your system. Some instruments don't sound too hot either, when played in this way, so some experimentation is required.

So why would a person want to use *GTablature*? For the users with a decent set of MIDI instruments (or, more likely, an output to a top-notch MIDI device) there is the potential to utilise existing tab as samples, or backing tracks for other songs. There is a MIDI export option in the File menu which could be used to distribute your compositions to other musicians.

It could also be potentially very useful for someone who is tabbing another artist's songs. *GTablature* could then be used as a record of the notes played as the tabber works them out on a guitar – the playback facility allows for verification that the tab you are creating is true to the original piece.

It also makes for a convenient way to share music with other people – when this software acquires printing facilities there is the potential to utilise a common file format for tabs to use for sharing them on the Internet. There are also plans afoot to integrate *GTablature* with *GChord* – a chord and scale repository – which should save time in placing complex chord structure on your tabs.

In the meantime it makes for a nice means of creating and utilising tabs, and is in general a very impressive effort for such a young program. The interface for creating tablature is reasonably intuitive if you know anything about tab, and very quickly becomes second nature to use. Its stability is reasonably good and it has a large number of features which many musicians could find very useful.



Config dialogs allows for a wide variety of tablature to be created.

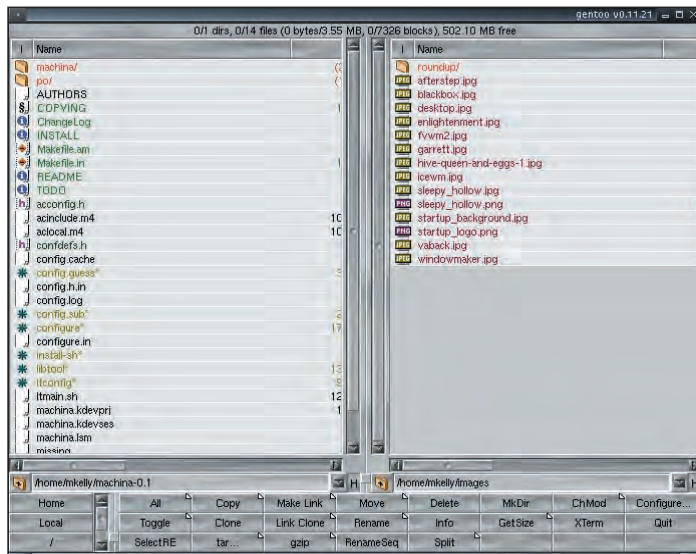
## FILE MANAGER

## Gentoo

■ VERSION 0.11.21 ■ WEB <http://www.obsession.se/gentoo/>

**B**ack in LXF 25, I thought I'd found a fantastic file browser in the form of *Endeavour Mark II*.

I raved about how it was fast, it didn't take over your desktop, and it did the job it was supposed to with minimal



The two-pane layout may be unorthodox, but it is useful.

frills. And it did. However, in the quest to find ever better software for the discerning LXF reader, another file browser has been brought to my attention – *Gentoo* (unrelated to the Linux distro called Gentoo).

*Gentoo* varies from most file browsers by being visually different, and a departure in terms of usage. At first glance it might appear as a slightly ugly rendition of the standard two-pane style of browser with a folder hierarchy on the left, and the files within the selected folder to the right. Not so – files and folders can both be found in the left-hand pane, and they can also be found in the pane on the right. Just below each pane you can find your main navigation options – an “up to higher level directory” button and a user-modifiable path which stores a list of recently visited folders.

The duplication of panes may seem like a complete waste of screen real estate, but you have to remember that *Gentoo* is not your standard file browser. Actions like copying and moving files are carried out with your source and destination directories both on display at once. This removes the needs for drag and drop and instead places emphasis on a more controlled style of file management.

Also breaking from the norm is the

complete lack of a menu at the top of the screen. This program does not miss any tricks. Aside from some basic operations which can be carried out using a right-click on a file or folder, just about all actions are carried out using the series of buttons at the bottom – the “button bank.” These can be re-defined to take advantage of a large number of built-in commands, or utilise the wide range of user commands that can be composed.

*Gentoo* is extremely configurable, and we're not talking about messing about with rc files here – everything is configurable from within the program itself. Not only that but the level of customisability allows you to make it the most personalised browser you can imagine. For example, most image types are pre-configured to open with the **view\_image** command, which defaults to using the *xv* program. To update all images to open with *Electric Eyes* instead requires only the modification of one user command. Simple, powerful, and less hassle than messing with a load of MIME types.

In short, *Gentoo* breaks the mould. It's efficient, practical and a pleasure to use. It's also so easy to customise that you'll soon find yourself using nothing else, and cursing people who don't have a copy installed on their box.

## DATABASE FRONT END

## phpMyadmin

■ VERSION 2.2.4 ■ WEB <http://phpwizard.net/projects/phpMyAdmin/>

**G**enerally I don't like picking Web-based apps for *HotPicks* – there are so many of them out there these days, that it's very hard to find anything unique or, at the very least, indispensable. One such app that I personally find indispensable is *phpMyAdmin*. It's been in use on my machines for nearly two years now, and I find it nothing less than brilliant.

*phpMyAdmin* is a PHP-based admin tool for taking care of a set of MySQL databases. There are command line and GUI admin tools for MySQL out there, but in some circumstances the only way to remotely administer your database will be via a web server running on the same machine, especially when you don't have the ability to connect to the database directly from a remote location. This is often the case with MySQL databases provided as part of a cheap web-hosting package.

Installation is simple – download the tarball, extract it to the appropriate

directory on your web server, and edit the configuration files so that they point to the appropriate database and have the correct username and password. Once that is done you just have to point your web browser at the appropriate directory and that's everything sorted. Well, not quite – you are advised to set up password protection on the directory, otherwise it may be possible for search engine spiders to traverse the links within *phpMyAdmin*. Some of the links are actually commands to perform tasks like dropping tables, so you don't want them executed by a spider!

When you browse to the tool you'll be presented with a frame-based layout – on the left is the list of databases, and on the right is the main work area. The amount of control offered over a database is fantastic. It is possible to do just about anything you might want to do with the databases, including destruction and creation. A text entry window is

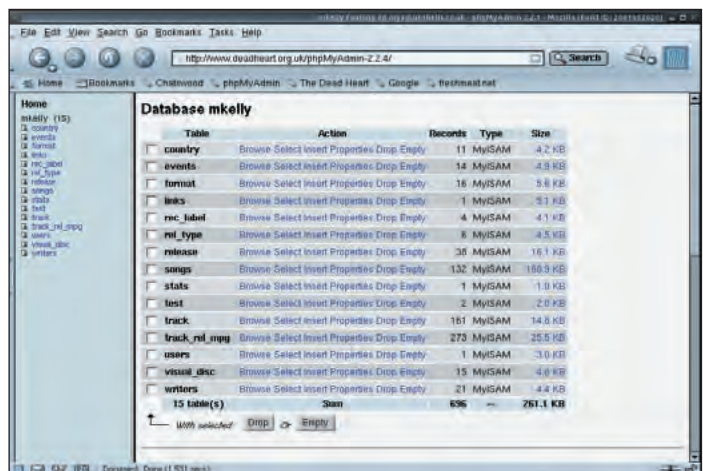
available for directly entering SQL queries, which is extremely useful when reloading previously dumped data. The dump options are numerous – with control over the output format, what data actually gets dumped and even the option to have the resultant data compressed using *zip* or *gzip*.

Browsing the databases can initially be confusing as there are a lot of options to take in. If you take a bit of time everything quickly becomes apparent, and there are many links directly to the MySQL online docs at

<http://www.mysql.com/>. The general layout is pretty much as effective as can be achieved with so much information to display.

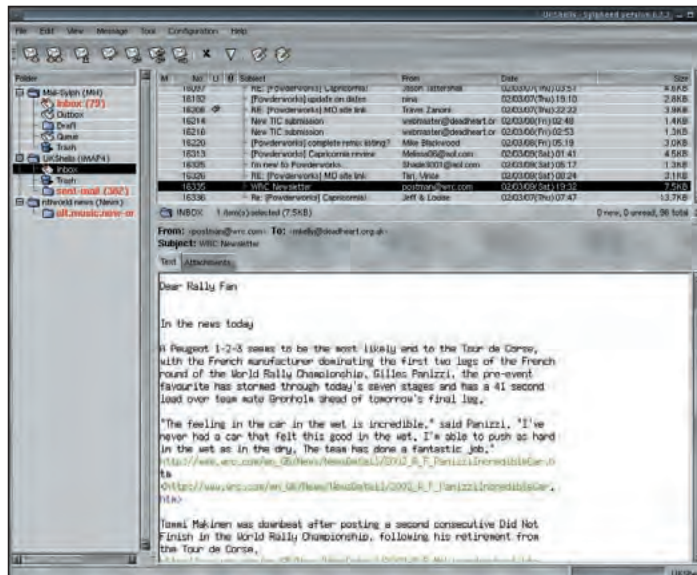
You can't get much more testament to the effectiveness of an application than by the fact that you don't realise you are using it.

*phpMyAdmin* is one of those apps that you'll find yourself using for a lot of your time, yet you'll never even realise. That's because it does the job quickly with minimal fuss, and genuinely makes life that little bit easier.



*phpMyAdmin* has to be one of the most useful web-based tools created.





The Sylpheed main window looks just like most other GUI mail clients these days...

## GRAPHICAL EMAIL CLIENT

# Sylpheed

■ **VERSION** 0.7.3 ■ **WEB** <http://sylpheed.good-day.net/>

**B**ack in LXF23 we had a roundup of some of the best mail clients out there for Linux.

Unfortunately we didn't have the space to include every mail client, and, in siphoning out the best, one was overlooked. This poor, neglected application is *Sylpheed*, and it is time that apologies were offered for its omission in the roundup.

*Sylpheed* is a graphical mail client which utilises the *GTK+* toolkit in order to run. On first execution it presents the user with a configuration dialog containing a wealth of options which can be set. You have the option to set up multiple mail accounts and can utilise a number of protocols including IMAP and POP3.

The interface is that of a bog standard GUI mail client – three pane layout featuring accounts and folders in a left-hand pane, and the message list and message display on the right-hand side. It may be boring, but it must work – otherwise why would mail clients keep getting made this way? You can separate the folder and message panes off if it helps you, but I can't quite figure out exactly how this is any better.

Indeed, in all honesty *Sylpheed* brings pretty much nothing new to the world of mail clients. Just about everything it can do, some other application can do in the same way.

That's an unfortunate fact when it comes to mail clients – email doesn't change that much, and so there's not an awful lot of reasons to produce innovative clients. But the reason

why *Sylpheed* is getting to be a *HotPick* this month is simply down to the fact that it does everything well.

Mail is displayed in a nice way – the interface is clear and messages are easy to navigate around and read. Multipart messages are displayed in a neat tabbed view which allows greater control over what



you do with attachments. Just about everything can be controlled using the keyboard, and *Sylpheed* even includes sets of key-bindings for a few other mail clients (including *Mutt*). It also supports threading and colourisation of messages, which make for a great time-saver when it comes to organising and reading posts from mailing lists.

A reasonable filtering system also helps when sorting out mailing list messages from your normal mail, and for getting rid of pesky spam. Filter setup is simple and can be performed based on a large number of pre-defined mail headers, or you are free to enter your own custom fields to discriminate upon. Each filter is restricted to two conditions, but this should be enough for all but the most advanced of filters you could conceive of. Your rules can be applied manually, or when *Sylpheed* picks up mail from your remote server.

If there's one reason why this program deserves to be this month's *Hottest Pick*, then it has to be the range of configuration options, which is very impressive. Each account can be configured individually for options such as account details, signatures, user-defined headers, and outgoing mail configuration. There is even the option to authenticate with a POP3 server before sending SMTP mail, which makes it easier to work with ISPs who have strict anti-spam measures. While the range of options don't surpass those on offer to users of software, such as *Mutt*, there really is a lot of control offered to the user here.

There's also a more general config settings dialog which offers loads more control over the picking up and sending of mail, as well as settings common to all accounts. Particularly useful is the ability to utilise an

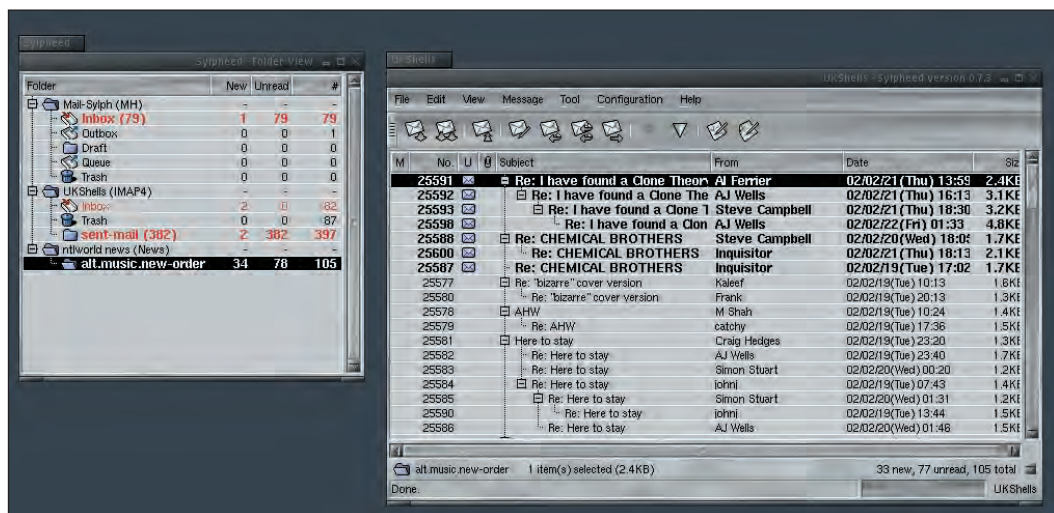
external editor – the built in editor is fine, but some people have their preferred software, and it is good to be able to choose it if you want to. It would be nice for new users to have context sensitive help in the configuration screens, but I guess you can't have your cake and eat it.

A built-in address book is included which makes it easy to use *Sylpheed* as your all-in-one mail solution. The address book is basic in appearance but stores a reasonable level of detail for each entry, and can handle vCards. A mail template facility is also available which could be useful if you send out a lot of standard mail messages.

As well as being a damn good mail client, *Sylpheed* also puts in a good stab at being a news client. It's general ease of use is not quite as good as that of dedicated newsreaders (such as the excellent *Pan*) but it still does a decent job for those who don't expect ultimate performance when news reading. The news interface is that same as that for mail, whereby newsgroups are listed in the folder window, with messages displayed at the right-hand side.

It's hard to find fault with *Sylpheed*. The lack of controls when reading messages in a separate window is just a minor quibble. It's a tidy application that does its job well. It's also pretty speedy at both POP3 and IMAP mail collection (even with some fairly full mailboxes), and no crashes were encountered the entire time I was using it. The addition of signature stripping would be a nice addition for future releases, as maybe would proper kill-filing and message scoring.

Sure, it has the graphical overhead that will not help to sway users of *Mutt* or *Pine*, but for anyone interested in a decent GUI mail client, you can't really go wrong with this program. [LXF](#)



...but for those who like to play about there are a number of ways you can make it act a bit differently.



# Welcome to Cooltown

cover feature



The astounding pace of technological change has seen computers invading every aspect of modern life. **Andy Channelle** looks at the next step in the quest for true pervasive computing, and the rôle Linux could play in Hewlett Packard's connection revolution.

Cooltown is 21st Century. Potentially it's the realisation of the ideas breathlessly predicted when the web was just a toddler and you could use fancy words like 'digitiserati' without everyone sniggering. But this project is more than just the clichéd fridge contacting the supermarket if you run out of milk: in cooltown everything is connected, and the human element is as important as the zeroes and ones flying through the ether from beacon to PDA. This is THE FUTURE as imagined by the inventors at Hewlett Packard's HP Labs. It takes the currently vogue concept of home networking out into the streets, so that – in the words of the marketers – 'everything has a web presence', from the cat to the kitchen sink. In cooltown, alarm clocks are smart enough to know whether the roads are icy or the



traffic is stacked up on the stretch of the M25 you have to brave each morning; your car will tell you if there's a problem with the engine (as well as booking you into the nearest garage to have it fixed, organising a taxi and letting work know you'll be late); and the settings for your workstation will be carried on your Sharp Zaurus, to be zapped onto the nearest available hotdesk as you enter the office.

If it all seems a little 2001: A Space Odyssey – the nightmare scenario of your fridge ordering in skimmed milk and cutting beer rations after a late night heart-to-heart with the bathroom scales: "You're getting love handles, Dave" – don't worry, Hewlett Packard promises this isn't the intention. The cooltown project is about making life easier and more joined-up. It's all about "never being lost in a strange city, never losing a receipt" and having the option of staying away from the office if you can't face the boss.

Jeff Morgan, who's been responsible for many of the ideas behind the project (including the Linux APIs), believes that mobility is the next big step in the evolution of the connected world, but a rethink on the way we do things is needed.

"This frontier will require a strong link between the physical and the virtual and a set of use models and application environments born out of the traditional desktop and PC environments of today." Morgan doesn't see faster connections as the dam holding back a flood of killer applications. "The connectivity issue is 2% of the problem," he says.

To take cooltown to the masses, hardware and software will have to work with the same easy grace as a digital watch or tumble dryer, but as with SMS messages on mobile phones, the cooltown 'killer app' is likely to be a surprise. "I really don't know where the killer app is going to come from (If I knew that, I would be a rich man). My best guess, based on the past, is that it will be some application that improves social interaction or information access, something that cuts across business and social boundaries. In social interaction, I include commerce since this is a big part of our society."

For many technology pundits, m-commerce was going to be the big thing of 2001 but people found current mobile devices too restrictive

(in both speed and functionality) for the whole concept to be viable. HP gets around this problem by utilising current technology in a new way. Your mobile phone/PDA becomes a universal remote control for other web-enabled devices, thus getting round their limitations. Of course there are going to be must-have gizmos, but the real revolution is in utilizing the devices already in the pockets and briefcases of millions. Working with what's available, Morgan says, is fundamental: "We didn't want to define another wireless standard but to utilise existing standards in support of the use models we have created for nomadic users." So your bleeding-edge Bluetooth-enabled phone is as valid in cooltown as your reliable old IrDA-equipped PDA.

### All join in!

A recent report by the Local Futures Group, a think tank funded by IBM, highlighted an increasing digital divide along both rich/poor and city/rural lines, and called for more action to bring digital services to the techno have-nots. In a vindication of the cooltown idea, the report concluded that mobile technologies are the best vehicle for bridging the gap, with emphasis on existing devices, such as mobile 'phones and digital television.

Inclusion is one of the project's watchwords; something Jeff Morgan is very passionate about. "The more people who use this technology, the cheaper it will be. The challenge is to get it into the hands of non-technical people, and the only way to do that is to provide value to them – enhance their lives. Allow them to have fun.

"One significant part of cooltown is the platform or infrastructure on which applications are built, services deployed, and within which devices produce and consume information. Like any other platform, it needs to be pervasive to realise the full benefit," he says. This was the reason for designing the platform around web and open source solutions. "From the start we made the barriers low by adopting a basic web infrastructure that was already pervasive. But we've extended the model to support new uses."

HP highlight their commitment to open source and Linux, so a project with the scale and ambition of cooltown was an ideal way to show their dedication, but there was more to the decision than mere posturing.

"In order to create a development community around the notions embodied in the cooltown vision, we needed to make the platform technology as accessible as possible. This led us down the open source route. Through open source and, where appropriate, Linux, we can deliver developer-accessible technology which will help us expand the community that is using, exploiting and extending cooltown."

This egalitarian model continues into the hardware arena with a range of client device classes designed to give everyone a chance to get involved. Services and uses will span a spectrum of client devices.

### Penguins in town

Hewlett Packard's commitment to open source development was cemented last year with the employment of Linux luminary Bruce Perens. His rôle appears to be that of advisor and evangelist, spurring on the development of Linux drivers for the company's printer and scanner ranges, and promoting the adoption of the GNU GPL on infrastructure projects such as cooltown.

Perens is confident that Linux is the ideal OS for the development of 'the mobile Internet'. In fact, he regards it as essential. "It's a very good OS for telecoms, and branches out to streaming media and sending video, where these things will happen over third generation wireless. Linux fits there very well. Another reason that it



Everything from the cat to the kitchen sink has a web presence.

**"It requires a strong link between the physical and the virtual and a set of use models from the traditional PCs of today."**

fits is that it's small and extremely portable. If you have a new computer architecture, one thing you can pretty certainly say is that Linux will run on it – not the case for any other OS."

One of Perens' perceived successes within HP is getting Linux drivers released for their most popular printers ("The printer division was already working on the solution to this when I came aboard," he says), and this has implications for cooltown. "The newer, higher-end printers, the LaserJets, have the *Chai* server software in them for doing some



# CoolTown



disconnected wireless services, and that is part of the cooltown initiative. So you can get a URL on a tag, scan it, and point to an HP printer and that printer will spit out the page of that URL." Moreover, the Chai server itself has been released as open source software and can be downloaded from <http://devnet.hp.com/community/chai>. "The reason we've open sourced it is that we want to have a platform-independent standard so that other people know how to use these special services of HP printers. And any successful standard within the past three years has an open source implementation behind it," Perens says.

HP have their eyes firmly fixed on the bottom line. "HP is embedding Linux and open source in products that make money, and that's something that I've been asking companies to do for years." The primary benefit of this outlook, says he, is that it has got a lot more people in business interested in Linux and open source than there were before.

"We're participating in a lot of the standards development. So you'll see HP's name on the standards. You'll know we're experts. And I think that [the mobile web] could be a very big part of our Linux business in the future," Perens says.

## Building blocks

The cooltown infrastructure is built around a number of elements – both software and hardware – each of



A mobile device could be used to tweak a presentation...

which is described in minute detail as part of the animated developer network that is evolving at [www.cooltown.com/dev](http://www.cooltown.com/dev). In addition to vision and technology, HP Labs believe that the future of cooltown will depend on 'a community of like-minded people who believe in open participation, investing in the web, and creating real solutions that add value to people's lives.' As such, much of the work already done is being set free; opening the gates on what is hoped will be a deluge of novel applications.

Wesley Chan, one of the coolbase project administrators, said the response to cooltown from open source development community had been very positive, but that more work was needed. "We do want more

people to build apps and help us add additional functionality," he said, thus HP want many people to "download and use the software, and eventually contribute back to it".

The architecture is divided into seven discrete parts.

### Coolbase appliance server:

At the heart of the entire project is the coolbase appliance server, which ties together the elements needed to create a cooltown application or service. Licensed under the GPL, coolbase includes software for 'enabling smart, connected web devices' and for representing people, places and things and their contextual relationships. Integrated into the main project are *CBserver* and *coolkit*. The former implements a very small

## Compliant clients

Your ticket to the brave new world

Hewlett Packard has defined a number of client device classes that range from simple information clipboards to powerful platforms.

### By Reference devices

This is a very simple interaction model based on the transfer and exchange of network references. By reference devices are the cheapest and simplest clients in cooltown and could be embedded



into watches, clothes, car keys and any number of things where portability and cost are paramount. HP's basic By Reference device is Taggy, a key fob sized transmitter/receiver that can participate in a cooltown environment.

### Content Carriers

These devices typically create and store some kind of content – digital cameras, basic PDAs and music players are prime examples. Cooltown provides a means for carriers to exchange (upload/download) content by means similar to the By Reference devices.



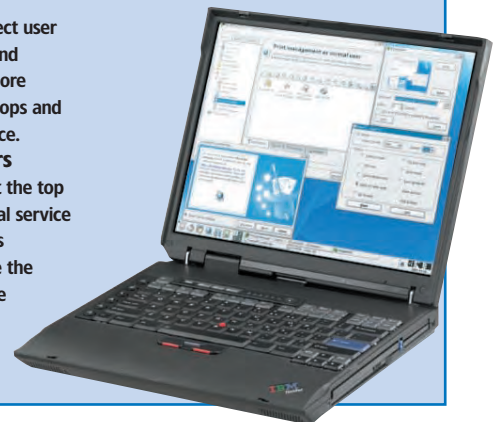
### Browsers

These devices include a screen and a means to render web content. They provide a rich environment for viewing information and supporting direct user manipulation and control – the more expensive palmtops and laptops, for instance.

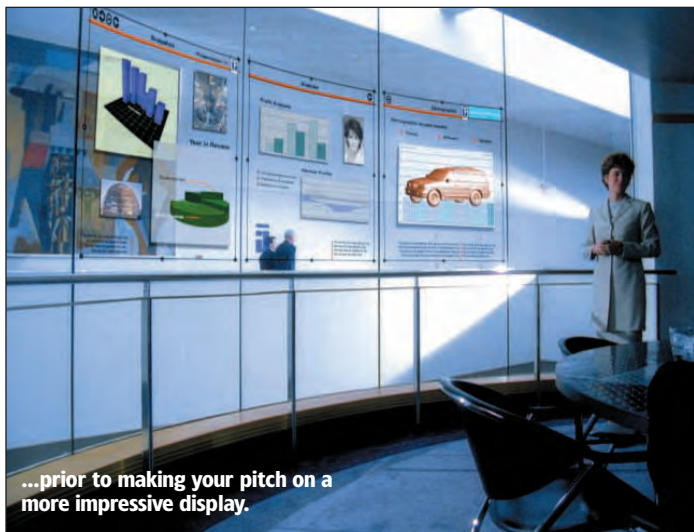
### Service Providers

These devices are at the top end. They have a local service platform capability. The services themselves are mobile (because the user is nomadic) and can include the user's environment and geographical data to provide personal services.

The above classifications can be combined, for instance your typical laptop provides all of the above capabilities, while something like Taggy just provides the By Reference support.







...prior to making your pitch on a more impressive display.

footprint, object-orientated web appliance server, while the latter gives developers all the tools for developing 'weblets' that run on it. While *CBserver* is designed specifically for embedded environments, it can be used in 'general purpose' environments such as Linux. The platform is currently regarded as stable and is available for Linux, WinCE and HP-UX.

#### Esquirt:

esquirt is described as a 'universal remote control for e-services' and is intended to circumvent the problems that most PDAs and mobile phones have when it comes to viewing media rich content: tiny, low resolution screens, short battery life, limited audio systems and poor networking abilities. esquirt is your go-between in the connected world.

For instance, you receive an SMS message from your partner pointing you to a short video postcard from Ibiza they've made to show you how much fun they're having without you. A mobile phone screen is going to have trouble showing the glory of an Ibiza sunset, so you beam a reference (the URL for the video) onto a nearby connected Widescreen TV – via a

reference to a cooltown-equipped photo printer (even charging the cost of the print to your phone bill) for a hard copy. esquirt also features facilities to receive information from cooltown beacons (see below) and is available for PalmOS, WinCE, Linux, Symbian/EPOC OS and Win32.

#### Web presence manager:

The Web Presence Manager (WPM) will handle the representations of people, places and things, or entities. According to HP, WPM 'enables the easy creation, management and hosting of cooltown web-presences. It is designed to support those interested in exploring the world of cooltown environments.' Essentially it is the bridge between the virtual and physical world, providing a method of representing any real life entity on the web with a couple of basic services. The first service holds a description of the entity, while the second contains references to other web entities that have a relationship with the first. So if you are in 'The Future Suite' at a hotel, the suite's web presence would have a directory entry pointing to yours and vice versa, and there would also be information describing the

**"HP suggests that total connection will lead to nothing less than a paradigm shift in the way that we access information."**

protocol such as Bluetooth or IrDA – and then watch your loved one getting loved-up in eye-popping colour. It's then a simple step on to select a still image from the video and send that

nature of your relationship to the suite, why you're there, how long you've been waiting, etc.

The current implementation of this architecture runs in a Servlet

## Cooltown beliefs

### Manifesto for a new world

When working on the cooltown concept, the 'inventors' at Hewlett Packard kept five thoughts uppermost in their minds.

#### Rampant diversity

For cooltown to be a success 'mobile and embedded information products, wireless and wired communication networks, and rich media content must be the norm.' This is often called pervasive, ubiquitous or anytime, anywhere computing, and relies upon the adoption of freely accessible and open standards.

#### The future network environment is the web

Obvious really, but proprietary formats – with their attendant licensing and 'retooling' costs – are far less desirable than the 'open, extensible, heterogeneous, standards-based network infrastructure' that we are already familiar with. The other benefit is that just about every computer on earth can understand the languages of the web.

#### Everything has a web presence

This is the central idea of the whole project, and like all good ideas, it's devastatingly simple. Cars, people,

buildings, doors, alarm clocks, shops and, yes, even fridges can all be connected to the 'Net. This allows the services you access to be tailored to your needs and also to be context aware.

#### Bridging the physical and online worlds

Online activities will be enhanced and enriched by access to real-world environments and artefacts, and vice versa. The scale and sensation of a geology exhibition at The Natural History Museum simply can't be transferred to a PC screen, while the physical space limits the amount of background information visitors can access. Cooltown unites bricks-and-mortar with the web to give users the best of both worlds.

#### Connected ecosystems of service providers

Douglas Adams' creation, Dirk Gently, would appreciate this idea: the interconnectedness of everything. HP suggests that total connection will lead to nothing less than a paradigm shift in the way we access information as 'as diverse services are woven effortlessly together.'

Container such as *Apache Tomcat*, or *HP Bluestone* and the web pages the end-user sees are developed using easily customisable JSP pages.

#### Beacons:

As the name suggests, beacons are designed to link locations to your cooltown client devices. They can be placed in a location, next to or even within an object, beaming a reference (typically just a URL) to that site or object's web presence. For instance, a beacon next to a painting by Goya could guide your palmtop to a site with a biography of the artist, links to other works and an online store where you could buy prints of the piece you've been admiring.

The beacon transmits its 'string' every three seconds and can also be used to receive device URLs which could point that beacon to your web presence. So if you were off out for a birthday meal with your partner, the restaurant could read that you are vegetarian and beam you a reference for their vegetarian specials, while your partner would get access to the whole specials board.

As well as releasing the embedded firmware code under the GPL, HP are



# CoolTown



Children will get the benefit that the Internet offers, in this case with an interactive Spanish lesson.



also allowing the beacon design and board layout to be used freely.

#### Taggy:

This is a prototype design that packs esquire and beacon receiver functions into a device the size of a key fob. The UI consists of a simple array of five buttons, which are used to set up and control the device. It communicates with the rest of the world over IR Ultra and also supports Object Exchange Protocol (OBEX) on Ultra. As well as releasing firmware code under the GPL, the hardware and board layout can be used freely. Devices such as

taggy are seen as a key technology in bringing cooltown to the masses and also to making it more pervasive – for everyone and everything to have a web presence, this type of device (one that can send and receive data) must be configurable, reliable and – above all else – cheap.

#### Baseboard:

Baseboard is an x86 embedded Linux system created specifically to provide an open source platform for designing 'web enabled entities'. The intention was to offer a reliable platform with no moving parts such as a hard disk and no graphic system, and it's hoped that the choice of an open solution built around off-the-shelf components will spur developers on to create the appliances needed to make cooltown a part of every day life.

The hardware platform consists of a Neptune single board computer from Adesta running at 233MHz sporting all the usual PC connectors and a 144MB solid-state disk. The system has been tested with a number of Linux distributions ("We used Linux because it was stable, widely available, and free. It was a natural choice for baseboard," says Wesley Chan); currently HP are using Red Hat 6.2 and have tailored most of their documentation to reflect this.

#### Secure web tunnel:

Security is obviously a concern when so much data is flying around, and this is where the secure web tunnel (SWT) comes in, providing transparent access to web resources that exist in 'mutually distrustful' environments, i.e. behind a firewall. The basic components of this system are SSL and an http proxy server, though the coolbase distribution has its SSL libraries removed to comply with licensing restrictions. Developers will need a valid SSL license to create a secure web tunnel.

## Big Brother

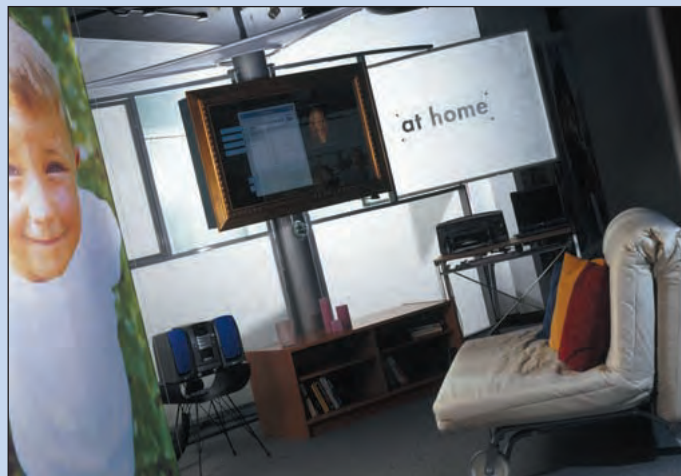
One of the scenarios HP constructs in its cooltown literature is the potential of the system for improving



A Spanish dictionary can be downloaded to a watch, offering translations of info from other cooltown devices in real-time.

## In darkest Surrey...

HP's £7 million cooltown centre in Woking, demonstrating how it could impact on our home, work and leisure time.



One of the most impressive demonstrations is the mirror, which is the centrepiece of the 'home' arena. Not only will it let you do your hair, but can also access the web, control other cooltown appliances installed in your house, surf the corporate intranet,

organise all incoming messages and, when not being used for its reflective qualities, display a series of digital images. It will also, of course, interface with cooltown-enabled printers.

When it comes to driving, cooltown has a number of tricks to improve your



journey. Biometric data allows you to be intimately linked to your car (no more theft), while GPS systems – linked to travel information – can route you around hold-ups on the road. The vehicle's web presence can also store service history, location

and engine status, just in case something goes wrong.

Advertising would be the primary use of cooltown when it comes to shopping, but users could also benefit from secure payments (via biometric sensors or the more traditional password



emergency services. During a house fire, for instance, the fire fighter's helmet visor could be a smart, connected device offering real-time updates on the situation and the condition of people trapped in the house. The waiting paramedic team could have ready access to the medical records of those in the house, and biometric data from the scene.

And though easy access to a user's data could, in this case, be a lifesaver, it's also where the biggest problem facing pervasive computing evangelists arises.

One potential – and extremely pressing – problem that could arise is that a personal user's primary device could be seen as an super ID card, containing not just name, address and National Insurance number but also bank and purchasing details, positional data and medical history. And this is something that the police (or a really dedicated cracker) could surreptitiously access as you walk down the street. It doesn't take a lawyer to work out that cooltown could be a civil liberties nightmare that makes the Big Brother of 1984 seem positively benign.

Would you like your genome and unexpurgated medical history trailing you around like a bad smell – for

insurance companies to discriminate against you? How about your credit record, to save you filling out forms?

And then there's the very nature of 'pervasive' computing – it's something that's everywhere. One concern expressed by a number of journalists who attended the opening of HP's cooltown centre in Woking was that the technology was yet another incursion of work into the home. "On one hand there's no need to commute," The Guardian's Michael Cross wrote, "on the other there is no escape from work" Cross paints a picture of a 'dysfunctional' environment where nobody communicates face-to-face, people are "unhinged by constant nagging emails" from appliances in the home, office and street, and we are forced into becoming semi-skilled technicians, trying to fix a fax machine instead of actually doing any work.

A computer crash is a real annoyance if you've just spent four hours crafting a begging letter to the bank, but would take on a whole new dimension if it prevented you from getting in through your front door, or decided to drive the wrong way up a motorway "because it's quicker."

If these difficulties can be addressed, cooltown could be the

## Weblinks

Cooltown home: [www.cooltown.com](http://www.cooltown.com)

Developer community: [www.cooltown.com/dev](http://www.cooltown.com/dev)

Chai development: <http://devnet.hp.com/community/chai>

Hewlett Packard and Linux:

[http://www.hp.com/products1/linux/linux\\_strategy.html](http://www.hp.com/products1/linux/linux_strategy.html)

European cooltown centre: <http://eu.cooltown.hp.com/home.htm>

**“Cross paints a picture of a dysfunctional environment, where nobody communicates face-to-face anymore.”**

idea that redefines work, commerce and our environment for the 21st Century. On the other hand, HP's vision could become the eight-track of the Internet age, leaving the pervasive computing crown to an established mobile operator such as Nokia – whose have a competing, though less ambitious, project on the drawing board.

Like every new advance pervasive computing has both benefits and drawbacks, but it seems likely that the great British (and American) public will happily forgo a little privacy in the quest for an easier life – until the cat flap crashes. **LXF**



authentication), notification of special offers and the help of virtual advisors.

Hot-desking has always been on the cusp of being the next big thing. Although there are still many problems to overcome, cooltown should at least make logging in and configuring a

workstation a little easier, as both these can be done over Bluetooth or IrDA from a PDA. It will also do away with the need to rewire the office every time you install a new PC or printer; everything is wireless.

It's in public spaces that cooltown



could have the biggest impact, but it will take some serious investment. The centre at Woking has a mocked up airport lounge where visitors can check up on the state of their flight, check-in using biometric sensors, iris- or fingerprint-recognition and print

documents over the air from a PDA. Of course, the plane would also be a cooltown device and could be appraised of your seating, food and alcoholic preferences, as well as other mundane details such as flight and destination data.

# LINUX format awards 2001 Winners

You nominated them, you voted for them, now you can find out who won the coveted title of 'Best of 2001'. As the drums roll, **Nick Veitch** gets ready to open the envelopes ...

**T**owards the end of last year, *Linux Format* decided it would be nice to honour the people who make Linux what it is with our own LXF awards. And, unlike other magazines where editorial teams sit around a big box of

doughnuts and decide who should get the plaudits, we decided the only fair way to accomplish this would be to allow you to vote. You nominated the contenders in each category, we published the voting form on [www.linuxformat.co.uk](http://www.linuxformat.co.uk), and then you

voted the winners. These are your awards, and they go out to the people you have chosen as the best of 2001. Give yourselves a pat on the back too, for taking the time to vote!

Several thousand valid votes were received, plus a few hundred that

were discounted for various reasons –not that we didn't like your answers, more to do with the fact that many of you for some reason seemed to think that they deserved more votes than others!

So, on to the ceremony!

## ★ Best Internet software

**NOMINEES:** Pan, Opera, Konqueror, Mozilla, Galeon

**This was a very closely fought** category. Early indicators were that *Opera* may take the prize, but when the main votes started rolling in, the eventual winner started pulling ahead. Without further ado, the name in the envelope is **Mozilla!**

It's been a year of great progress for the Mozilla team. Not only did

Netscape finally decide it was complete enough for the basis of *Netscape 6*, but Mozilla has spawned many other projects too: the excellent *Gecko* engine and it's standards compliance have made it incredibly popular, and though it didn't have a huge lead in the polls, at 26% of the votes cast, it was enough to win!



## ★ Best server software

**NOMINEES:** Apache, Cups, Samba, MySQL, Postfix

**In contrast to the last category,** this one produced a clear winner. So clear in fact, that it garnered more votes than all the others put together, and then some. That winner is, as you may have guessed, **Apache**.

From it's early start as 'a patchy' server, *Apache* has grown to be the real backbone of the worldwide web; the latest Netcraft surveys giving it over 60% of all active web servers. With *Apache 2.0* due out soon, 2002 will be an interesting year too!

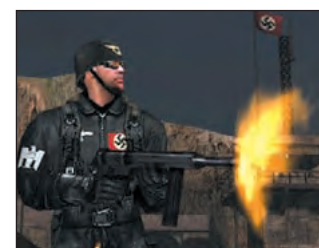


## ★ Best Game

**NOMINEES:** Loki, Return to Castle Wolfenstein, Nethack, tuxRacer, Exult

**This was one category we had no** idea about who would win. Would people vote on gaming excellence, on strict adherence to Free Software principles, on longstanding support for the Linux community? Whatever their reasons, the voters made their choice and elected **Return to Castle Wolfenstein** as the best Linux Game. It only just sneaked into the awards in time, so it might be a surprise choice.

While not officially supporting Linux versions, it have released



plenty of code before, spawning *Quake* for Linux, *Doom* for Linux, etc. This is the first game to feature a simultaneous release of the necessary Linux code.



## ★ Best Distribution

**NOMINEES:** Red Hat, SuSE, Slackware, Mandrake, Debian

**Now here's a category always sure** to provide some interesting voting. It was, no surprises, the most popular category, with only 10% of voters not registering a preference here.

This was a tough category to call, as there has been lots of progress in the last year. Red Hat finally produced a 'friendly' installer, SuSE added cool features and updated Yast, and Mandrake,



well the additions kept coming.

It is probably because of the continuing ease of use, cutting edge packages (some may say bleeding

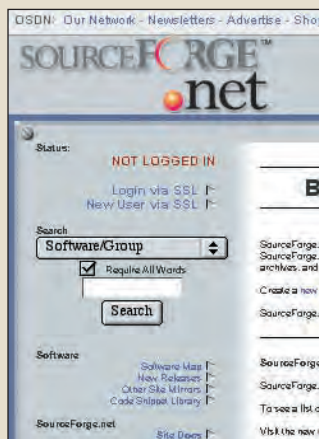
edge) and the development of fantastic tools like *DiskDrake*, that the readers chose **Mandrake**, who polled a total of 44% of the vote. Well done!

## ★ Best developer resource

**NOMINEES:** Loki, Sun, Red Hat, Sourceforge, O'Reilly

**This category attracted quite a** few votes, with few abstentions. The idea here was to find the organisation or service that provided the best support for developers.

Perhaps unsurprisingly, given the number of projects hosted there, **Sourceforge** gained almost 60% of the votes here. By basically providing free hosting, CVS and bugtracking services to open source projects, from the very big to the very small, Sourceforge provides an essential service to the Open Source community.



## ★ Embedded Linux award

**NOMINEES:** Cobalt Raq, Agenda, Montavista, Sharp, Lineo

**We told you your vote** counted, and this category certainly proved it. Although not everyone chose to make a preference in this category, those that did certainly made theirs count. After looking at the raw figures from the poll, there was little separating the top two contenders here – in fact, a mere four votes.



But, then the envigilator appeared on the scene, and processed the votes removing duplicate entries and other votes from untraceable email addresses (we did warn you we would check). Both candidates lost a few votes, but the result was perhaps more surprising – a tie! Both **Sharp** and **Cobalt**

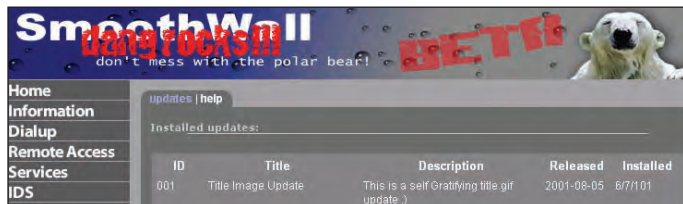
have the exact same number of votes, at 29.67% of the total each, so both will be on the receiving end of a well-deserved Linux Format award.



## ★ Best security tool

**NOMINEES:** Mandrake firewall, Xsentry, portsentry, LIDS, Smoothwall

**This category was open to a wide** field, everything from simple network security tools to full-blown security distributions. Some well known tools and projects were nominated here, and the vote was fairly evenly split between them.



The (potentially controversial) winner is **Smoothwall**, with 38% of the vote. Smoothwall has come a long way since we first featured it in *Linux Format*, and powers secure connections all over the world. A commercial version is also available.

## ★ Best Advocacy

**NOMINEES:** Bruce Perens, RM Stallman, 'Maddog' Hall, IBM, linuxnewbie.org

**The idea here was to reward the** person who had best furthered the cause of Linux/Open Source/Free Software. The nominations could really have been for almost anything, from a website to a company, and we duly received nominations for a wide range of obscure and not so obscure, portals, corporations and individuals.

The long list was whittled down to the five nominees by virtue of the number of nominations they received, but the shortlist is also a

general reflection on the diversity of nominations here.

Stallman perhaps was the odd-man out, as he doesn't actually advocate Linux at all, but he nevertheless does promote Free Software and accordingly got a large percentage of votes.

The outright winner, presumably due to the sheer scale of their involvement with Linux, was **IBM**, with a highly creditable 42% of the vote.

## ★ Individual Achievement

**NOMINEES:** Linux Torvalds, Alan Cox, Ben Collins, Mattias Ettrich, RM Stallman

**This category was supposed to** reward the individual who had made the greatest contribution towards Linux during 2001.

Obviously a certain **Torvalds** chap got nominated, but it was far from a forgone conclusion – the award was for achievements during that year – a year which had also seen Alan Cox doing some sterling work on the kernel, and chaps like Mattias Ettrich leading

the KDE coders to glory with successive, rapid improvements to that desktop environment.

Still the voters decided that Linus should get the prize, though his total of 51% of the vote belies the fact that there was some reasonable competition in this category.



# LinuxFormatAwards



## ★ Best Support Resource

**NOMINEES:** SuSE, Mandrake Expert, Linuxjunior.org, linuxquestions.org, Linux Documentation Project

**Getting hold of Linux is easy, but** often, getting good, reliable, accurate information on how to use it, or solve particular problems can be difficult. Thanks to the many people who nominated this magazine and its website, but we couldn't really give ourselves the award (and besides, we might have lost!), but a range of other worthy candidates emerged.

All the nominations were for web-based services, which I guess goes to show how connected our readers are, and how the Linux community is really a global, Internet based one. Some of the websites were independent sites, and two were supported by distro vendors,

SuSE and Mandrake. The Mandrake Expert scheme is certainly a novel one which has proved quite popular and polled very well here.

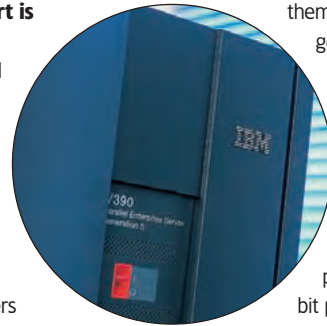
Nevertheless, the outright winner, with a commanding 55% of the vote, was the home of the HOWTO, the excellent **Linux Documentation Project**. If you've never visited [www.linuxdoc.org](http://www.linuxdoc.org) (shame on you), pop over to add your congratulations.



## ★ Hardware support

**NOMINEES:** Creative, Intel, IBM, Matrox, Nvidia

**Hardware support is** an interesting category. Designed to reward those who best support their hardware, it was a category which included some interesting nominations. Not many manufacturers have a good track record of Linux support, but a smattering of household names do make the extra effort, though perhaps not always in the way the community would like. Nvidia for example, do provide excellent drivers for their graphics cards, but they don't open source



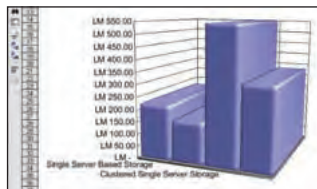
them. Nevertheless, they got into the shortlist. Some of the ones that didn't make it included chip manufacturers Intel and AMD, whose real support for Linux, particularly on 64-bit processors, is perhaps still a little invisible to the average user.

In the end it was **IBM** who took the glory here, mainly, if the nomination reasons reflect the vote as a whole, for their investment in developing Linux (with the help of others like SuSE) for the s/390.

## ★ Best Business Software

**NOMINEES:** Applixware, Evolution, GNOME Office, Koffice, OpenOffice

**Business software is a category** that could have potentially included all sorts of applications, but the nominations pretty much centred around office suites. There is certainly a lot of competition in this sector, with great progress being made on *OpenOffice*, *KOffice* and the loose conglomeration of packages



that make up GNOME Office. *Applixware* got a mention, as too did the surprisingly good *Evolution*.

But it seems when it comes to spreadsheets, word processing and presentations, the voters prefer **Open Office**, which polled a stunning 46% of the vote in what we expected to be a close-fought battle. A tremendous amount of progress has been made on this since early 2001, when it was barely useable, to today, where despite niggles, it clearly displays its competence. A hearty congratulations to the hundreds of developers!

## ★ Best Development tool

**NOMINEES:** Kxlix, jEdit, Quanta+, Kdevelop, Python

**In a community where you could** be forgiven for thinking that sometimes the developers outweigh the users, where the *lingua franca* is Makefiles and C++ classes, we had to have a category for best development tool.

The shortlist included commercial and non commercial tools, and covered a wide range of disciplines. Even a language managed to get onto the shortlist! *Kylix* has certainly made it's mark on the Linux world and just managed to nudge ahead of Python in the Polls, but both fell a few hundred votes short of the



eventual winner, **Kdevelop**.

Perhaps it is a sign of the dominance of KDE that this tool proved so popular, but you certainly can't take away from what is a well crafted IDE for Linux development.

## ★ Open Source Project of the Year

**NOMINEES:** Apache, Galeon, Samba, OpenOffice, KDE

**Finally we come to perhaps the** most prestigious award, the open source project of the year. The rules here are simple – which project was the best of 2001? Understandably, many of the nominees here were also nominated (and in some cases won) other categories in the awards.

*Apache*, as we have seen, stormed the Best Server software category. *OpenOffice*, through dint of hard work and amazing progress

topped the Office Software poll. *Samba* had a creditable performance, and perhaps suffers a little by being too good – it becomes invisible because it just works and people get on with using it.

*Galeon* has impressed many, coming from nowhere to become a very competent, clean little browser, and even managed to get nominated ahead of the project that spawned it, *Mozilla*.

When the votes were counted though, there could be only one winner. **KDE2** was revealed just before the beginning of 2001, but as we saw, that didn't signal a slowdown in the development process – far from it. Rapid successive versions kept up a high pace of development, not only in



the base system itself with new features such as antialiased text, faster startup and better internationalisation, but also in its attendant applications like *Konqueror*, *Kate*, *Kdevelop*, *Kugar*, etc. KDE is certainly a worthy winner of this award – the Linux users of the world salute you! **LXF**



# What on Earth is... LSB?

Imagine a world where all Linux distributions work in the same way.  
**Nick Veitch** raises the standard for a unified Linux platform.

## »» **LSB, is that some cunning three-letter acronym hiding the identity of some new web technology then...**

Not at all. LSB actually stands for Linux Standard Base, an organisation designed to increase the interoperability of the various flavours of Linux.

## »» **What standards are they a base for?**

The S stands for Standard, in the singular. The organisation does not administer open standards like those governing web protocols or whatever. The entire purpose of it is to create or outline a basic set of features that should be expected of any Linux distribution. This includes identifying and selecting some open standards, but the LSB people do not define these standards.

## »» **OK, what do they do?**

Working with other organisations and interest groups, including most of the major distribution vendors, they try to define a template for a basic Linux distribution.

## »» **So every Linux Distribution would be the same?**

No, the distributions would continue to be as unique as they are currently, more or less. The idea is not to make them all look the same, but to make them all functionally similar in certain respects.

One of the problems with Linux is that all the distros tend to do things differently, often for no better reason than that's the way they have always done things. These differences usually have no great significance, but they do mean that, for example,

things get put in different places, like init scripts, user accessible commands and so on. This can cause problems for people trying to distribute software with an install system, even rpm packages can require different versions to work on different platforms. There are guidelines for where things should go: the Filesystem Hierarchy Standard, part of the LSB. For more info on this specifically, you should check out <http://www.pathname.com/fhs>.

## »» **So the LSB specification is for developers?**

The LSB Specification is written for application developers and platform/operating system developers. For software developers, it will provide a set of rules which should allow any application they develop to run on any LSB compliant platform.

For the distribution developers, it provides a specification which, if followed, ensures that LSB compliant software will run on their Linux distro.

The specific benefit for the Linux user is obviously, that if they have an LSB distro, any LSB software should install and run on it.

## »» **So in the end though, it's just a list of what should go where?**

Not at all, the LSB is more than just a guideline for directory trees. In fact, that is only a (albeit important) part of it. The LSB specification attempts to give a complete binary interface definition for compliant platforms. This means that software should behave EXACTLY the same, no matter which LSB platform it is running on. It also goes much further than most 'development rules'. For example, if you were developing for Windows, say, you would

have a set of APIs, which merely give the the information on the system libraries and services you can build your application against. The LSB covers the OS environment to the binary level.

Thus, the LSB specification is really a family of specifications. There is a general one, which covers the common features, and specific ones for each of the processor platforms covered. Currently this includes only PC-style architecture, for 32-bit and 64-bit (Itanium) processors, but the plan is to include PPC and other processor types too.

## »» **But what does this really mean?**

For example, you may already have noticed that, in `/usr/lib`, there are probably thousands of files. If you look closely, you'll probably find that many of these files are symbolic links to others. So you might find:

```
libsomelib.so.1.0.1 -> libsomelib.so.1.8.0
libsomelib.so.1.2.0 -> libsomelib.so.1.8.0
libsomelib.so.1.2.2 -> libsomelib.so.1.8.0
libsomelib.so.1.4.0 -> libsomelib.so.1.8.0
libsomelib.so.1.7.1 -> libsomelib.so.1.8.0
```

The reason for this is that some files expect to find a particular version of a particular library. While most libraries *should* be backwardly compatible, it can't be guaranteed – so apart from clogging up your directory, it might not even work.

The LSB aims to define a minimal set of libraries as part of a base distribution, and give them specific/unique names so that, e.g. `libsomelib.lsb.1` could be relied upon to work with all applications that required it.

The advantage of this approach is that the other versions could still exist – you could still install the





# WhatOnEarthLSB

« very latest *libsomelib* if you wanted, but it wouldn't break lsb-compliant software built against *libsomelib.lsb*.

An LSB compliant installation would include a complete set of libraries that could be reliably used with identical results. The libraries currently identified include not only *libc*, but also common/useful libraries like *libcrypt*, *libpthread*, *libutil*, *libncurses*, *libz*, *libX11* and more.

## » Okay, so all common libraries for all languages will become part of the LSB specification?

No, and no. Only the reference implementation libraries will be released, as mentioned. A typical installation would have far more libraries than these, but these are the ones considered to be the bare essentials.

On the language front, only C/C++ are being supported. The kernel is written in C, as are nearly all of the commands and tools that make up the LSB. Yes, there may be people developing in FORTRAN, ALGOL68 or whatever, but these are not likely to form part of the LSB. This doesn't mean that you can't use such software on an LSB compliant platform, or that you shouldn't develop it – it just means that it falls outside the sphere of interest for the LSB.

## » But basically the LSB is all about files and libraries then?

No, as we touched on earlier, the LSB also considers certain programs or commands as essential for a base installation. Any LSB-compliant installation would include these commands. These are things that you, and any installation or system script, may take for granted and includes things like *grep*, *chmod*, *ls*, *ln*, *more*, *mount*, *umount*, *patch*, *diff*, *tar*, *passwd* and similar. A total of 123 common commands are currently included in the LSB specification.

The important thing here is that the commands simply must behave in exactly the same way. *E.g.* the LSB specified *tar* command only guarantees *--bzip2* as the switch to filter input/output through the bzip compression algorithms, not the *y* or *j* switches commonly in use (depending on which version of *tar* you have). While the commands themselves may offer additional options to the ones defined, the ones that are defined must always be present, so they can be relied upon.

## » Blimey. Okay then, the LSB covers file hierarchy, libraries and commands. Are we done now?

Nope. LSB also covers common services, like the shell (a reliable shell implementation is important for lots of things, not the least of which is install scripts). There are also notes covering package formats; users and groups; device specification (*i.e.* what devices must always be present in */dev*, like

*/null*, */zero* and */tty*); and a rather large section dealing with system initialization.

## » Woah there. Okay, package formats – what's the deal there?

Well, basically the standard that has been agreed upon is the RPM format, as defined by the Red Hat document at [www.rpm.org](http://www.rpm.org). This gives a reliable, standard method of package delivery

## » What about non-RPM systems such as Debian?

These can be and are still compliant. The only requirement is that RPM packages *can* be installed, not that they are the package system of choice. Debian has the *alien* tool, which should work okay with RPMs that follow an acknowledged standard.

## » Okay, what's this nonsense about Init scripts then. Surely these are all the same?

You'd be surprised. The idea here is that things like run-levels, the way cron tasks are set up and the way init scripts work are all standardised. For something like run-levels, this is more or less a confirmation of the systems that are already followed. For pretty much any distro I can think of, run level 1 is 'single user' mode, run level 5 is an X-enabled multi-user environment, and runlevel 6 is reboot.

The case with services is a bit more vague. Although most init scripts follow the usual conventions, like accepting only one argument which tells the service to stop, start, restart and so on, there are other things to be considered.

If you remove packages, the config files are often left behind (by design, in case you want to re-install), but some scripts behave badly when they are called but can't find the binaries to execute their service. So it is useful to define how they should behave.

## » What about clashes with names for scripts etc?

The LSB is leaving the assignment of namespaces to the Linux Assigned Names and Numbers Authority ([www.lanana.org](http://www.lanana.org)) to avoid conflicts.

## » You mentioned 32-bit PCs and the Itanium, what about non-Intel processors then?

The LSB is a co-operative effort by volunteers. So far, there have only been enough people with the right skills to work on the specifications already mentioned. Some work on other processors is underway, but the best way to find out the current status of such efforts is to visit the Linux Standard Base website ([www.linuxbase.org](http://www.linuxbase.org))

## » What LSB-conforming distros are there then?

Ah, well none really. The LSB test suites are still

under development, so it's a bit early days in that department. As the testing suite isn't finalised, nobody can say who passes what and why. Though to give you some idea, the current version of the LSB-FHS test suite (which tests the file hierarchy or directory structure) contains 243 individual tests, and when run on some now out of date distros, the results ranged from 5 failures (for SuSE Linux 7.0) to 44 (Caldera's OpenLinux eServer).

## » Does this mean that SuSE is best?

Not at all. The tests are not finalised, and in this case weren't performed on vanilla installs of the distribution, which may mean that extra software 'broke' the distros. Also, as the test-suites are not finished, the results are open to question.

Also, some things are not anything to do with the distros – if you as a user, for example, chose NOT to install python, then your distribution would not be LSB-compliant (which requires */usr/bin/python* as a reference for a python interpreter).

Also when you consider that, for example, the *bash* Shell isn't fully POSIX compliant, and therefore doesn't comply with the LSB, there is very little the distro maintainers can do to solve some of the problems at the moment.

## » What is this test suite then?

The test suite is a series of scripts which attempt to verify that all aspects of the LSB specification are met. These are split into different tests, which, for example, test the compliances of the File-hierarchy system, or that the LSB minimum spec commands are installed where they are supposed to be.

The test suite itself is available through CVS, but it is still very much a work in progress. You can check on current progress and find out how to get hold of the CVS version at [www.linuxbase.org/test](http://www.linuxbase.org/test)

## » So when will we see LSB badges on distributions then?

The test suite for x86 Linux was scheduled to be finished by now, but some work is still continuing. Depending on the final test suite, the rapidity with which changes can be applied to distributions, the prevailing wind direction, sun-spot activity and other factors, you'll probably see some LSB distros this year.

## » I notice that there is nothing about Internationalization in the LSB specification. Why is this not covered?

Internationalization issues are not covered by the LSB. Yet. They may be, they may not be. However, the Free Standards Group, which spawned LSB does have a project concerning Internationalization. They have produced a "Globalization Specification" called Li18nux which is available from the

www.freestandards.org website. This is now a release specification, and should cover any of your questions regarding language support and similar issues. It is highly likely this will be included in a later version of the LSB.

### » In the meantime, how can I test my software?

Well, if you just want a LSB compliant test installation, there is a sample implementation of the LSB available ([www.linuxbase.org/impl/](http://www.linuxbase.org/impl/)). You can use this as an environment for checking that your project will run okay and is LSB compliant, or just to see what the LSB compliant. There is a tarball for download that includes some RPMs and a Makefile for creating the LSB implementation. This will be installed to /tmp/live, and you can then 'enter' this environment with:

```
chroot /tmp/live
```

to test anything you like. Make sure you copy whatever you want to test to /tmp/live as well!

### » And if it all works then I'll be fine?

Well, possibly. The test suite itself, as I keep saying, is a work in progress, as is the sample implementation. It may be that they themselves do not fully conform to the LSB specification (and if you do notice any errors, be sure to report them to the relevant persons).

### » Blimey, that's a bit of a palaver

It isn't that much bother, but there are other simple tools you can use to check, for example, that your application or library conforms to the specification.

### » And if my software passes I can get some sort of badge or logo then?

Your software will also have to be audited before you receive a Certificate of Compliance, which you will be able to use in documentation and other related material. At present, no auditing process has yet been defined.

### » Isn't this more for commercial software then?

It has more apparent appeal for commercial developers. Your GPL project doesn't guarantee that the software works or is suitable for any purpose, but obviously commercial vendors are deemed to be more responsible for their products, particularly in the area of support.

That said, any software which does comply with the LSB, should work just as well as any other. It probably isn't going to be too much effort to develop software that complies to the standard, and if it means more people are actually going to be able to use it properly, that's just as much a benefit to open source developers as to anyone else.



### » This is all very well, but who actually decides what the standards are?

The contributors to the LSB. The contributors range from individuals to companies and other organisations, who take part either by actively participating in the specification or by sponsoring activities associated with the LSB.

### » What organisations are involved then?

On the company/organisation side of things, there is a lot of representation from the distribution developers. SuSE, Red Hat, Mandrake, Caldera, Debian, TurboLinux. There is also a lot of support from companies with an interest in Linux: VA Linux, The USENIX Association, Linuxcare, SGI and Compaq are also members, and IBM sponsors a lot of activity.

### » How can I get involved in the LSB?

The written specification for the Linux Standard Base is developed by volunteers, with varying degrees of available time and experience. There is a core team, to ensure some stability.

But you can contribute in a number of ways. If you have plenty of time and energy to devote, you could help one of the core team members with their tasks. For those whose time might be less flexible, you can always, for example, send patches and comments to the LSB forums and participate in the discussions on the various related mailing lists.

### » What mailing lists are there?

There are a variety, covering the different groups that make up the LSB effort.

- lsb-discuss@lists.linuxbase.org** – General lsb discussion
- lsb-spec@lists.linuxbase.org** – The written LSB specification
- lsb-test@lists.linuxbase.org** – About the LSB test suite.

**lsb-impl@lists.linuxbase.org**

– The LSB sample implementation

**lsb-freestandards-fhs-discuss**

**@lists.sourceforge.net**

– The file hierarchy discussion

**lsb-taskforce1@lists.sourceforge.net**

– Packaging discussion

These lists have archives available, but some are hosted on different list servers. To subscribe or check out the archives, follow the links from [www.linuxbase.org/lists.html](http://www.linuxbase.org/lists.html)

### » Are these open to everyone?

The lists are public, but as usual it's best to read the list first and try and get an idea about what things are actually discussed. There was a guy recently suggested that the FHS should include a condition that all configuration files should go in \$HOME/.settings, and received some polite and not so polite (but all constructive) replies. The major point here was that he hadn't really read the list and didn't understand that it's about standardising the hierarchy structure in the least obstructive way. The usual rules apply, try not to get too excited and take things too personally – flame wars just clog up the lists and stop important things from being discussed.

### » I have ideas for things not currently covered by the LSB

Well, there is a subcommittee for this too, LSB-Futures. The goal of most of the other working groups within the LSB is to perfect and polish the ideas, rules, software, etc. that they already have. Throwing completely new stuff in at this stage would, like any other software project, make a final and absolute release a bit of a moving target.

The mission of LSB-Futures is to consider ideas for additions to future versions of the LSB and how they might be implemented. This might be anything from specifying further services or libraries that might be included in the specification, to expanding the LSB to include *e.g.*, hardware driver modules, interpreted languages, and so on.


### » How do these things get chosen?

There is a process for this. The Futures group works mainly by assessing opinion amongst users and developers, to get a feel for which features are required or would be helpful for everyone.

### » What sort of things are on the list so far?

Currently under consideration are, for example, a collection of common image libraries like *libpng*, *libjpeg* and also the GNOME base. KDE is also under consideration.

### » And if I want to find out more?

What more could you need to know! There is plenty of information at [www.linuxbase.org](http://www.linuxbase.org) 



## Emulators



# Nintendo emulators

More ingenious emulators as **Simon Goodwin** wonders if **SNESsity** is the mother of invention.



**W**e reported on two dozen Nintendo emulators in *LXF21*, promising an update covering emulators for more capable Nintendo systems. Now we give a Linux perspective on Nintendo's adventures in the 1990s, emulating the Sega-bashing Super Nintendo, via the eccentric Virtual Boy, to the Nintendo 64 console, and beyond.

As usual I'll start by putting the emulated systems in perspective, picking up the Nintendo story where *LXF21* left off – at the end of the eight-bit era – and explaining how Nintendo tackled the moves to 16-, 32- and 64-bit gaming.

### SNES surfaces

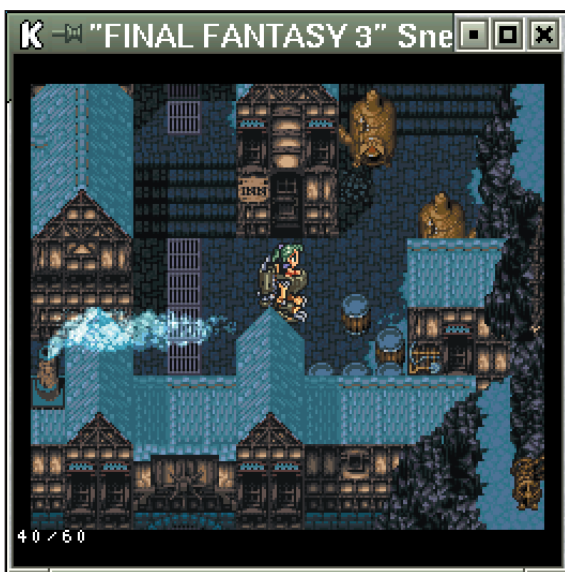
Nintendo dominated console gaming at the beginning of the 1990s, with the GameBoy and original Nintendo Entertainment System, or NES, but both had sound and graphics rooted in the previous decade, and eight-bit processors yet ten years older.

Sega's 16-bit MegaDrive was wowing home gamers, and Nintendo was slipping behind; we reported on Sega emulators in *LXF19*.

Nintendo's riposte was the SNES, short for Super Nintendo Entertainment System, which surfaced in 1991. Like Apple's GS and its emulators, tested in *LXF22*, Nintendo plumped for a 65C816 central processor. This is a 16-bit elaboration of the 6502 used in their best-selling NES consoles, with so few registers that even the x86 set looks generous by comparison. It buzzes along at around 3MHz; the exact speed depends on the memory type, but in any case CPU emulation is unlikely to be a bottleneck on usable Linux systems.

Apart from the custom graphics, the SNES hardware is modest, with 128KB of main memory and 64KB each for video and audio. Sony's SPC-700 chip handles sound, playing up to eight samples at a time, expanding them on the fly from four bit ADPCM to 16-bit stereo. This association between Nintendo and Sony became problematic later.

The SNES graphics array sets it apart from other consoles of its era, and poses challenges for emulators. Two PPU 'Picture Processing Unit' chips assemble each screen from four



Square's *Final Fantasy* series blended cinematic and role playing conventions on SNES.



Nintendo's 64 bit console, codenamed Project Reality.

HARDWARE PHOTOS COURTESY EDGE MAGAZINE

overlapping, smooth-scrolling images or 'planes', built from 8x8-pixel tiles. 256 colours are available at a time, organised as eight tables of 32 entries, each from a 15-bit palette.

Nintendo's famed mode 7 stepped towards modern 3D systems, allowing slices of the screen to be rotated and scaled in hardware according to vector and matrix settings. Some cartridges include co-processors to boost 3D capabilities. Eight DMA engines can transfer data to and from video RAM much faster than the processor could, with tight synchronisation to reprogram the display between one scan line and the next. Then 128 sprites can be mirrored vertically and horizontally and overlaid on the main 256 by 224 pixel window, with up to 32 on any scan line. All this means Linux SNES emulators deliver distinctive and stylish demos and games.

## Ultra NES

Nintendo struggled to follow-up this design, and ended up skipping a generation from the 16-bit SNES to the 64-bit product they introduced five years later. This joint venture with Unix workstation makers Silicon Graphics was initially labelled 'project reality', then renamed Ultra 64, and finally surfaced as the Nintendo 64, or N64, in 1996.

The SGI connection steered Nintendo away from proprietary planar technology to 3D graphics modelled on OpenGL. Mesa and other OpenGL clones mean that Nintendo and Linux 3D graphics have far more in common than they do with DirectX and Direct3D, the moving targets suffered by Windows developers.

The 64-bit MIPS R4300 processor is hard to emulate as conventional interpreting emulators can't hope to match its speed. In other respects the N64 hardware is within the resources of a modern Linux box – there's 4MB of dynamic RAM, 21-bit colour and 16-bit stereo sound. But the similar 3D graphics means many N64 system calls can be passed directly to Mesa on Linux, without the usual emulation overhead.

The N64 runs ROM cartridges, rather than optical media. This makes games quick to start and helps Nintendo protect their manufacturing monopoly, but limits the quality of the software as motion video and uncompressed sound do not make good use of expensive ROM space. In fact there is a working version of Linux for N64, as one of the links later explains, but it requires a hardware hack to make it run.

## CD no thanks

Nintendo did not deliberately skip both CD and 32-bit technology. They made two attempts to join those bandwagons, but deals with other firms fell through. In the meantime they

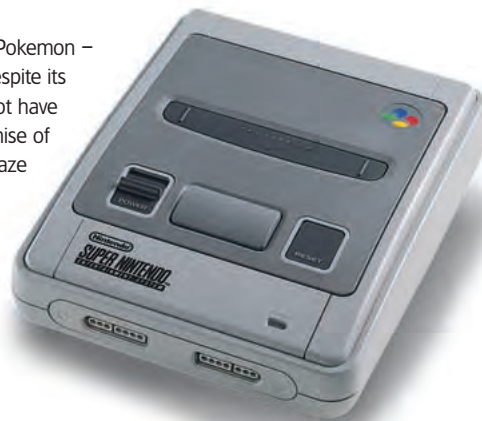


gambled on the Virtual Boy, a true-3D portable system, which vanished leaving little but an interesting emulator.

The unreleased Nintendo CD32s were joint ventures with Sony and Philips, who between them own the rights to the CD format. Nintendo courted Philips first, collaborating on the development of CDi home consoles and planning a CD add-on for the SNES to rival Sega's MegaCD. Meanwhile Sony were working on what became the PlayStation, another 32-bit CD-based console, and dug out a deal predating the SNES, when Nintendo promised that any CD-based console they did would be a joint venture with Sony. So the lawyers came in, Nintendo jumped camp to Sony, then jumped back, unwilling to hand over control of software production. Time passed; lawyers got substantially richer; the Philips CDi flopped, PlayStation conquered the world, and

Nintendo missed the CD32 boat.

If it had not been for the success of Pokemon – and continuing sales of the GameBoy, despite its hardware limitations – Nintendo might not have made it into the new century, as the demise of Sega demonstrates. But the Pokemon craze has bankrolled the development of two new and strong Nintendo systems – the Game Boy Advance, which belatedly puts them in the 32-bit game market – and the GameCube console.



**The humble SNES was the best-selling console of its era.**

## New cubes

The GameCube is based on an IBM PowerPC RISC processor clocked at 486 MHz, so there's little prospect of Linux systems, other than PPC ones, emulating it convincingly for a few years. Yet GameCube emulators actually preceded the real hardware, and helped Nintendo substantially, especially on PowerMacs where the processor and OpenGL graphics architecture map closely to the final GameCube hardware. Even now, *GCC* and *bash* are staples of the development environment.

The Dolphin x86 emulation still used by some GameCube developers relies on the cross-compilation of C code, so it's not a true emulator – more of an emulation library – but the OpenGL influence of Silicon Graphics continues to dominate the video API. With Sony's in-house Linux expertise spilling out to PlayStation 2 owners, and Linux on Dreamcast and N64 already, the convergence is clear; emulators are leading us forwards, as well as back to classic games.



**Fake64 is the only N64 emulator specially made for Linux**

## SNEmul

**T**he first option is *SNEmul* – I tried 0.61 and 0.73 for Linux, which are both fairly old, but I've not found a more recent version. Neither would work on SuSE 7.2 because they needed old libraries, but 0.73 worked on Debian 2.2. Unfortunately I can't include screen-grabs

as it uses DGA for direct video access, running as root with access to other program interfaces disabled till you quit.

*SNEmul* looks neat, with control window icons to call up a machine-code monitor, audio and video configuration and other setup panels, and to switch the size

of the main emulation screen window. It looks best in a 640 by 480 screen mode.

You must launch *SNEmul* from an X terminal as root. It uses *Allegro 3* for platform abstraction and the rather buggy *SEAL* librsor sound. It can save screens in PCX format, and save/load snapshots.





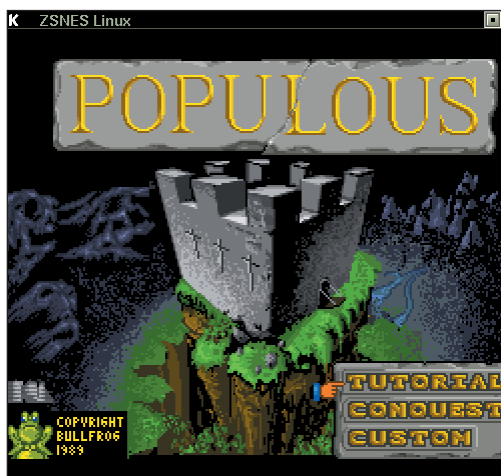
# Emulators

## « ZSnes – custom WIMP

**Z**Snes starts with five main menus offering game, config, cheat, network and misc options that can assign the non-game control keys, GUI style and eye candy effects. A file requester opens in the launch directory, and a short-cut menu allows quick access to previous selections.

The Esc key toggles between this GUI and the running ROM. Control keys can be assigned for each of four players, including the four main directions plus optional diagonals, select and start keys like those on the console, and normal and turbo variants of the six main gamepad buttons;

**Bullfrog's Populous runs well on 16 bit consoles but is a pain to control with keys or a joypad.**



a SNES mouse can be emulated or Super Scope or Lethal Enforcer add-ons.

At the start players can be assigned keyboard, joystick or gamepad control. Game Genie and similar cheat codes can be filed, searched and applied automatically or via menus after loading a game ROM. ROMs can be in any of ten formats, of which the 'SMC' emulator image is the most common.

Video can use a window or VGA-sized full screen, in colour or grey-scale, with three optional levels of scan-line separation. F1 calls up a 'misc options' menu that can dump the raw contents of the sound buffer or the screen, single-step the emulator a display frame at a time.

While the emulator runs function keys allow the state of the emulator to be saved and restored and you can switch the eight sound channels on and off with the F5 to F12 keys. ZSnes has good stereo, though it loses sync momentarily when unrelated programs are launched and during opaque window dragging. You can minimise by launching the emulator with 'nice —adjustment=-10 zsnes', giving the four emulator processes a priority boost.

### Compatibility

The RPM of ZSnes 1.31 beta installed smoothly on my K6/2-500 with SuSE Linux 7.2. Most of my SNES software worked fine, though some demo graphics were slightly scrambled, probably because

programmers relied on exact timings of the real SNES. *Guikuden 1* is the only standard SNES game documented as not working at all, though another half dozen show graphical emulation bugs, including *Final Fantasy 3* and *'Metal Combat*.

Some cartridges contain add-on chips that may cause problems. Graphics are garbled in games like *Street Fighter Alpha 2* and *Star Ocean* that use the SDD-1, and the SuperFX version of *Dirt Track* and *Winter Gold* are not correctly emulated.

ZSnes comes with a reasonable man page and command line help, plus extra text files in '/usr/doc'. The GUI is readily worked out by experiment. Command line options allow from zero to nine frames to be skipped between renderings, so that slow computers can run games in real time, although less smoothly. The frame rate in a 512 by 448 pixel window on my Matrox G400 was 30 to 40 frames per second. The auto frame skip option gives the best rate your system can manage by dropping frames as necessary. Other options adjust the display gamma, and cater for PAL and NTSC video formats.

ZSnes uses Intel MMX (or AMD equivalent) SIMD parallel processing instructions when available. It is mostly written in x86 assembly language, so it's not portable to other CPU families. The Linux version requires SDL 1.2, zlib for de-compression, and NASM 0.98 to assemble the GPL'd source.

## Nintendo 64s

**T**here are three N64 emulators for Linux – all quite limited at the moment. Your choices are *Fake64*, *Mupen64* and *TrueReality*.

*Fake64* is open source and the only N64 emulator developed specifically for Linux. It's timing is currently a bit wonky

but it runs some demos and incorporates a dynamic compiler and simple machine-level debugger.

*TrueReality* was initially developed by Austrian Niki Waibel, since joined by others who have extended and ported the code. It now runs on BSD and Solaris,

AmigaOS, MSDOS, OS/2, MacOS and Windows via DirectX, OpenGL or Glide. So far *TrueReality* runs demos but not commercial game ROMs. It's far from complete but relatively well-documented. You'll need GLIBC 2.2, SDL 1.2 and Mesa-4.0.1 to compile it on Linux.

## MuPen64

**M**uPen64 was written by Hacktarux in France. It was designed on Linux but it is not open source, though it may be later, once the planned plug-in system is working. This is a pity as *MuPen* is designed to be processor agnostic. Like *TrueReality*, it has been ported to BeOS and Windows, as well as various Linux setups.

*MuPen64* runs *Mario64*, but only in

wireframe, and various CPU demos. It has an optional dynamic compiler. It can load '.bin' and '.v64' game and demo image files, automatically unpacking them with 'zlib' if necessary.

Startup is slow as you must answer two French questions in the console window, typing numbers to chose the type of CPU and custom chip emulation you want. Diagnostic messages continue to spew

across the console as the emulator runs.

The Rotate demo runs nicely, as does Project Unreal's plasma. The Redbox Mandelbrot zoomer looks good though the speed hardly matches 64-bit standards. The fire demo works (roughly); it's many times faster when using the dynamic compiler than the interpreter, but the 64-bit R4300 is not easy to emulate on a typical 32-bit CPU, either way.

## Snes9x

**S**nes9x is another excellent SNES emulator, mainly written in C with some C++ and optional assembler CPU emulation. It is portable to non-Intel architectures, unlike ZSnes, and has been tested on big-endian as well as little-endian CPUs. I tested version 1.37 on the same SuSE/K6 set-up, and found it worked even better than ZSnes in some respects, though it lacks the configuration GUI. Relatively complete documentation is included but no man page.

Snes9x has many features in common with ZSnes, like adjustable speed and screen shots. It can freeze and restore games, emulate Game Genie and Action Replay cheat cartridges and add-ons like the SNES Mouse, SuperScope and Multi-player 5 widget. Argonaut's SuperFX co-processor is emulated, like the faster SA-1 main processor in some carts and the C4 graphics accelerator used by Capcom in *Megaman X2* and *X3*. DSP-1 is supported enough for *Mario Kart* but not for other games like *Pilotwings*, though ZSnes also struggles to emulate those, and *Metal Combat* and *Street Fighter Alpha 2* defeat both emulators.

The minimum host is a DX4/100 with 8-bit graphics and no sound, but you'll need a 16MB P200 or equivalent to run most games. Graphics can use any pixel depth in X windowed or full-screen modes; 15 or 16 bits give the best performance, though 8-bit is fast if you

don't care about transparency effects.

ROMs can be in '.fig', '.smc' (SuperMagiCom) or '.sfc' (SuperFamiCom) snapshot format, or split images

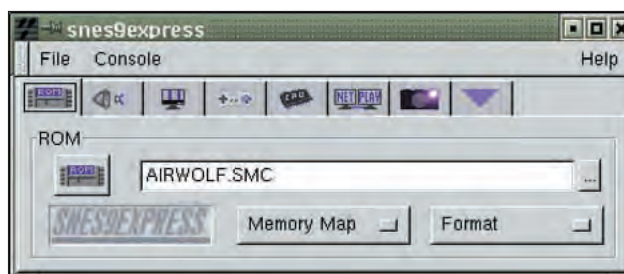
### Express control

Four joysticks or Sidewinder pads can be supported with the appropriate kernel drivers. The first five digit keys toggle four display layers and sprites on and off, followed by controller assignments with keys 6 and 7, then various graphics options on the remaining number keys. The '-' and '=' keys can speed up or slow down emulation by frame skipping.

Tab puts the emulator into turbo mode, the bracket keys toggle sound options, Scroll Lock pauses and Esc quits. Function keys are used to save and reload game states and saved positions. Frozen



Mario takes to the track in *Mario Kart* on Snes9x.



games are a lot bigger than the save files which would normally be written to CMOS RAM on a real SNES, but can be saved at any time rather than just when the game designers planned.

You can specify PAL or NTSC video mode on the command line, plus sound, joystick and video options similar to those for ZSnes. The '-is' switch interpolates sound samples, making low-frequency sounds smoother. It is an improvement but still rather prone to hiss compared with the curve-fitting done by ZSnes.

Snes9x starts in a small window using one pixel for each on the original system, and can manage 60 frames per second at that size. Scaling up to around 800 by 600 pixels drops updates to about one frame in 3, which is still quite playable.

Good video and audio options on most ROMs are '-y -sy -sc -r 7 -dfr -sy'. These deliver about 40 fps in a quad-sized window on my system. On a slow machine try '-j -ne -r 1 -dfr -mono' in an 8-bit video mode.

**Snes9x is set up and launched from this neat control panel.**

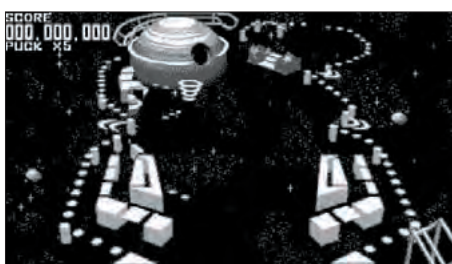
## Reality Boy

**R**eality Boy emulates an obscure 3D portable, the Virtual Boy, which used separate red displays for each eye, giving convincing 3D but in shades of red. It's more a technology demonstrator than a fully-developed product, which is probably why it never got a worldwide launch, but now you can try it on Linux.

The obscure hardware is emulated well

enough to run four games fully and half a dozen more to some extent. You can run it in mono or with separate displays for each eye, but that's missing the point.

The 3D effect can be ingeniously supported through the use of 3D glasses like those used for 3D films and printed novelties, or LCD shutters like the iGlasses once promoted by Escom.



Reality Boy's *Pinball* demo, in eye-preserving mono.

### And there's more!

As if the eight bit Nintendo emulators tested last year were not enough, we've found Goober, another GameBoy emulation, and put it on your coverdisc to complete your eight bit Nintendo collection. You'll also find Reality Boy, several Super Nintendo and N64 emulators, demo ROMs and documentation on the disc. There are already three Game Boy Advance emulators for Linux, all developing rapidly. We've run out of room to discuss them this month, but will return to test them and other Japanese console emulators in a future column. **LXF**

### Nintendo Links

Fake64 home:

<http://insomniac.ath.cx:81/~bluefyre>

Goober: <http://goobergbe.sourceforge.net>

MuPen64:

<http://mupen64.emulation64.com>

N64 demos: <http://www.dextrose.com>

N64 Linux:

<http://www.ix.de/ix/artikel/E/1997/04/036/>

Reality Boy:

<http://www.goodnet.com/~dh90791>

SNES demos: <http://www.cherryroms.com>

SNemul home: <http://snemul.free.fr>

Snes9X home: <http://www.snes9x.com>

TrueReality:

<http://www.emuhq.com/truereality/unix>

ZSnes source: <http://zsnes.sourceforge.net>

ZSnes home: <http://www.zsnes.com>



# 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 special forums? ([www.linuxformat.co.uk](http://www.linuxformat.co.uk)). Hope to hear from you soon!

**Nick Veitch**, Editor

## How code is represented

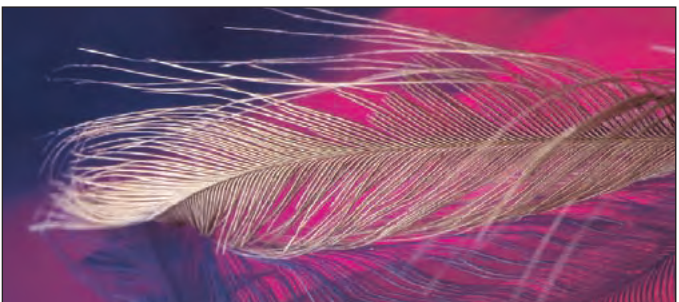
Including code in magazines can be tricky, but we hope our notation will help it become clear. When lines are too long for our columns, the remaining text appears on the next line in a solid blue box:

```
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.
```

## THIS MONTH...

### Apache >>

Configure Apache and start serving web pages, bind to non-reserved ports and then move on to name-based virtual hosting **p68**



### Perl

More on manserver – a client/server protocol for man pages – and a look at network protocols **p72**



### Java

Customize the way your classes are serialized as we delve deeper into Java's serialization mechanism and evolve some custom classes **p76**

### << FlashGo

Plug three USB portable data storage interfaces into your USB ports? No need with FlashGO! Now you can read all but the most obfuscated storage device **p84**

### Kylix

Following last month's examination of error handling we take a look at resource protection and raising exceptions using try...finally **p78**

### CGI

Introducing a short series on the Common Gateway Interface. Using Perl to provide dynamic web content and deal with HTML forms **p86**

## TIP OF THE MONTH!

**Have you ever needed to install** some software quickly – without having to muck about with building it – but the binary version is packaged in a format incompatible with your distro? If so, then what you need is the tool *alien*. *Alien* is a program that can convert between RPM, Debian, Stampede SLP, Slackware TGZ and Solaris pkg file formats. It tries to map the dependencies of a package appropriately in package formats that supply dependencies. *Alien* is still rather experimental, however, and can often get things wrong.

*Alien* ships with Debian, Red Hat and other distros. You can download it

## Alien package conversion

from <http://kitenet.net/programs/alien>. Depending on the type of packages you want to convert, you may need some additional tools. To manipulate RPMs, you also require the *rpm* command, and to handle Debian packages, you need *dpkg*. Non-Debian users can get hold of *dpkg* in the package *alien-extra*, from: <ftp://ykbsb2.yk.psu.edu/pub/alien/>

*Alien* is very simple to use. Specify what package format you want to convert to and the file path of the package you want to convert. *E.g.*, to convert *foobar-1.0.1-1.i386.rpm* into a Debian package, enter as root:

```
alien -d foobar-1.0.1-1.i386.rpm
```

This will create a file called *foobar\_1.0.1-2\_i386.deb* which can then be installed on a Debian system.

### Alien switches

<b>-d</b>	<b>--to-deb</b>	Build a debian package (default)
<b>-r</b>	<b>--to-rpm</b>	Build an RPM package.
<b>-t</b>	<b>--to-tgz</b>	Build a Slackware package
	<b>--to-slp</b>	Build a Stampede package
<b>-p</b>	<b>--to-pkg</b>	Build a Solaris package

## NETWORK CLIENTS

# Roll your own network clients

This month, **Charlie Stross** gets stuck into writing a network client in Perl and has fun with man pages – lots of fun.



Last month we took a look at how network servers work, what they do, and how to go about writing a simple one in Perl – most of the effort going, not into Perl code, but into defining a protocol (language) for the server to talk to its clients.

This month, we're going to look at how to go about writing networking protocol clients. We'll start with the client for the simple man page protocol (introduced last month), then look at some other aspects of client-side programming in Perl – including things like off-the-shelf libraries for writing web robots, parsing HTML, and doing FTP.

What does a network client look like in Perl? Here's about the simplest web client you can knock up using the modules in the standard Perl distribution:

```
use IO::Socket::INET;

my $remote = "www.linuxformat.co.uk";
# a well-known web server ;-)
my $http = 80;          # standard port for HTTP traffic
my $sock = IO::Socket::INET->new(PeerAddr => $remote,
                                PeerPort => $http,
                                Proto => 'tcp',
                                Type => SOCK_STREAM)
    or die "Couldn't connect to $remote:$http: $!";

print $sock "GET /\n\n";    # a minimalist HTTP request

print STDOUT $sock->getlines();
close $sock;
```

The **IO::Handle** module in the standard Perl 5.6 distribution provides a common interface to i/o tasks in Perl. A variety of subclasses such as **IO::File** and **IO::Socket** are derived from it; they understand all the methods of **IO::Handle**, but add additional methods for dealing with files and sockets respectively. In general the **IO** modules all use similar semantics; you treat an **IO** object as if it's a wrapper around a file handle. The usual Perl filehandle i/o commands (such as **print()**, **getc()**, **eof()** and **read()**) are all present, albeit provided as methods to call on the **IO::File** or **IO::Socket** object rather than as built-ins. Some of these core modules have sub-types in turn; for example, **IO::Socket** provides a generic interface to the Linux socket mechanism, but you need to use **IO::Socket::INET** to handle internet domain sockets (ones that send data over TCP/IP), in contrast to **IO::Socket::UNIX** (which builds UNIX domain sockets, which run very rapidly but are restricted to the local machine).

Note that there are some non-standard **IO::Handle**-derived modules that aren't in the standard distribution. **IO::Stringy** lets

you treat a scalar variable as if it's a file handle – great for splatting volumes of test into a handle that you're going to pass to some other subroutine. And **IO::Stty** lets you do things to the i/o settings on a serial TTY device. But we're not going to deal with these here.

It's possible to use lower level Perl commands to set up a socket connection; the **Socket.pm** module provides a low-level interface to the C **socket.h** standard library, and the basic socket system calls are provided as Perl built-in commands. But unless you want to deliberately build TCP/IP packets by hand to meet some specific aim (such as generating malformed packets that appear to originate from another machine) there's not much point to doing this. The basic socket connection provided by **IO::Socket::INET** lets you connect to a server on a remote host, read and write data, check for the end of a data stream, and set buffering.

## Protocols – client's eye view

Just being able to connect to a server isn't enough; the client needs to know how to get the server to do something useful. The usual solution, as we saw last month, is to use a protocol – a well-defined set of rules governing how a client and server should communicate. Protocols dictate the order in which a client and server exchange information, so that the session never enters a state in which the server is waiting to hear something from the client and the client is waiting to hear something from the server – a deadlock situation that can persist indefinitely. Protocols often provide some level of error recovery.

The example in the previous section is about the lowest-possible level of client for HTTP, the hypertext transport protocol used by the web. After connecting to an HTTP server, our client prints a message like this:

```
GET /
```

(followed by two carriage returns).

It then reads whatever the server emits and prints it to its standard output. The **GET** command in HTTP (or rather, the **GET** method) tells the server to return the object named after it – in this case **/**; the document root on the web server. If you execute this code, Linux Format's web server will reply:

```
200 OK
```

Then emit a stream of HTML (the default web page it's set to serve up as the document root).

HTTP is a simple, stateless protocol; that is, we can connect to a server, issue commands, and each command will be executed without reference to the previous one. Other TCP/IP protocols are not so simple.

An example of a stateful protocol is SMTP, Simple Mail Transport Protocol. The client can connect, tell the server to accept mail from person **X** and destined for person **Y**, and start sending the mail. The **MAIL** command that introduces the mail tells the server to enter a state in which it is accepting mail from



someone. This can be followed by **RCPT TO** command, telling the server who the mail is for. But **RCPT TO** can be followed by either another **RCPT TO** (introducing another recipient) or by **DATA** (introducing the message data). Keeping track of which state the client is in is essential. Other common stateful protocols include FTP (File Transfer Protocol), NNTP (NetNews Transfer Protocol), and IRC (Internet Relay Chat).

In Perl, the commonest way to handle the client side of a stateful protocol is to use a Perl module that provides an object-oriented interface to a session using that protocol. When a client program connects to, say, a mail server to send mail out, we can be pretty sure that its connection, and its state within that session, aren't dependent on any outside influences. By wrapping the connection up in an object, and providing methods to poke data into the object and read data out, we can build an object-oriented interface to a protocol that makes it easy to build network clients.

As it happens, there are Perl modules on CPAN that implement the client side of a whole bunch of network protocols. You can install a bunch of them with the following command:

```
perl -MCPAN -e 'install Bundle::libnet'
```

This bundle installs **Net::Cmd**, a collection of methods that can be inherited by descendants of **IO::Handle** and that provide tools used by command-based protocols (like SMTP or FTP). These in turn are used by **Net::FTP** (which, with its special-purpose subclasses, provides a client-side implementation of the FTP protocol); **Net::NNTP** (which provides the NNTP netnews transfer protocol client); **Net::POP3** (a client for the popular POP3 remote mailbox delivery protocol); **Net::SMTP** (which allows client-side sending of outgoing email); **Net::Time** (an ntp client); and utility modules such as **Net::Netrc** (which provides access to the .netrc file used by the *ftp* program to store login details for remote FTP servers) and **Net::Domain** (which attempts to verify the domain of the current host).

## Simple manserver protocol

Last month we introduced a simple sever that provides access to man pages over the network (by invoking *man* in response to various commands, and returning its output to the client). The manpage protocol is about as simple as it gets. The client opens a connection to the server on port 8888 and sends a request. Requests are a bunch of lines, each with a keyword and optional arguments. The server waits for the request to terminate (with an **END** command) then sends a reply and closes the connection. The server times out stale connections after 60 seconds.

To request a manpage, the client sends:

```
PAGE <manpagename> SECTION <section_number> END
```

The **<section\_number>** can be a number (**1-8**) or the keyword **ALL**, meaning send all the man pages named **<name>** – see below for how they're delivered.

The client can run an **apropos** search (equivalent to **man -k**) by sending:

```
SEARCH <regex> END
```

And can run a **whatis** search (equivalent to **man -f**) by sending:

```
DESCRIBE <manpagename> END
```

If the server parses the request and knows what to do, it sends:

```
200 - OK
```

Followed by the body of the manpage or search result.

If the request doesn't make sense, the server sends:

```
500 - Bad Request
```

And closes the connection.

Now, we could use the tools in **Net::Cmd** to implement proper protocol-dependent behaviour, but the manserver protocol is stateless; we just issue commands and read whatever

comes back. So **Net::Cmd** is overkill. Instead, we want a command – called, say, **manc** (for **man client**) – that accepts the same basic arguments as **man**: that is, a man page name, optional section, and optional **-f** and **-k** arguments. Depending on its arguments it should then open a connection to the man server (specified in a simple configuration file in /etc – /etc/manc.conf) and send the request. It should then print whatever comes back, or a suitable error message.

## manc client program

The *manc* client is designed as a replacement for the *man* command. Instead of invoking external commands (notably *troff*) to format a man page stored on the local filesystem, *manc* parses its arguments, then opens a socket connection to the *manserver* program (in last month's column) and sends a protocol request.

Here are the highlights (line numbers not required by Perl): First, we set up the modules *manc* uses. In this case, **Pod::Text** is used to format the help text (actually a man page embedded in the program), **Getopt::Long** is used for parsing command line arguments, and **IO::Socket::INET** is used to talk to the server:

```
013: use strict;
```

```
014:
```

```
015: use Getopt::Long;
```

```
016: use IO::Socket::INET;
```

```
017: use Pod::Text;
```

**use strict** isn't an external module but a compiler pragma – it turns on strict variable initialisation checking.

```
019: # ----- main program variables
```

```
020: my ($man) = ""; # flag: used to print man page
```

```
021: # ----- set up config variables - default
```

```
022:
```

```
023: my ($server, $port) = "";
```

```
024:
```

```
025: if (exists $ENV{MANSERVER}) {
```

```
026: $server = $ENV{MANSERVER};
```

```
027: } else {
```

```
028: $server = "localhost";
```

```
029: }
```

```
030:
```

```
031: if (exists $ENV{MANPORT}) {
```

```
032: $port = $ENV{MANPORT};
```

```
033: } else {
```

```
034: $port = 8888;
```

```
035: }
```

The bit above lets us configure the *manc* program via two environment variables – **MANSERVER** and **MANPORT**. These default to **localhost** and **8888** respectively and indicate the server the program will try to connect to. (We could have built a configuration file mechanism in, using a module like **ConfigReader::DirectiveStyle** to do the heavy lifting, but for the purpose of a tutorial program this seemed like overkill. We demonstrated a program that used **ConfigReader** in LXF 24.)

Our *manc* client implements the basic *man* features – lookup man page by section and name, search the man descriptions for a string (**apropos**), and retrieve the one-line description of a man page name (**whatis**). The next part of the program uses **Getopt::Long** to work out what sort of command line we're processing. The picture is complicated by the possibility that *manc* has been fed a mangled command line (say, a combination of the **-k** (**apropos**) and **-w** (**whatis**) flags).

```
044: # -----
```

```
first, get options for -s, -k, -w flags
```

```
045:
```

```
046: my $section = ""; # manpage section >>
```

## LinuxFormatTutorialPerl

```

047: my $page      = ""; # man page name, if given
048: my $apropos   = ""; # apropos string, if given
049: my $whatis    = ""; # whatis string, if given
050: my $help      = 0 ; # flag; print help if > 0
051:
052: GetOptions(
053:     "section=s"      => \$section,
054:     "s=s"            => \$section,
055:     "k=s"            => \$apropos,
056:     "w=s"            => \$whatis,
057:     "help"           => \$help);
058:
059: # -----
        now get manpage and section from ARGV
060:
061: if (scalar(@ARGV) == 2) {
062:     # presumably we have arguments of the form man
    <sect> <page>
063:     ($section, $page) = (@ARGV);
064: } elsif (scalar(@ARGV) == 1) {
065:     # presumably we have arguments of the form man
    <page>
066:     $page = $ARGV[0];
067: }
068:
069: # -----
        check for confused arguments
070:
071: my $sanity = 0;
072: if ( $apropos ) { $sanity++ }
073: if ( $whatis ) { $sanity++ }
074: if ( $page ) { $sanity++ }
075: print "sanity: $sanity\n";
076: if ( ( $sanity != 1 ) or ( $help > 0 ) ) {
077:     # the user entered an invalid mixture of arguments or
    asked for help
078:     do_help();
079:     exit;
080: }

```

Lines 52-58 are a call to **GetOptions**. **GetOptions** (imported from **Getopt::Long**) scans the command line and if it finds a known option it puts any parameter to that option in the appropriately named variable.

Lines 61-67 then check for **section/page** arguments. *man* traditionally takes either one argument (a man page name) or two arguments (section number, page name), hence the significance of looking for the number of arguments specified after we've finished processing the options.

At lines 71-80 we check for valid combinations of query. **\$sanity** is a flag that indicates a sane request; if we've got parameters relating to different types of query, something's gone wrong and it's time to give the user some help. If not, we have more or less appropriate parameters, so it's time to move on:

```

082: if ( $apropos ) { print STDOUT @{ do_apropos($server
, $port, $apropos) } }
083: if ( $whatis ) { print STDOUT @{ do_whatis($server,
$port, $whatis) } }
084: if ( $page ) { print STDOUT @{ do_man($server, $port,
$page, $section) } }
085: exit;

```

That's all the main program!

Actually, it isn't. But depending on whether we're processing an **apropos**, **whatis**, or **man** query, we need to do different things. To avoid excessively complex **if()** constructs, we farm the

details out to different subroutines, each of which returns an **arrayref** pointing to an array of lines of text containing the answer sent by the manserver program. This, when printed, provides the output for *man*.

Three subroutines (**do\_apropos**, **do\_whatis** and **do\_man**) create a manserver protocol request, as a string. They then feed the request to a single routine, **talk\_to\_server()**, and return whatever **talk\_to\_server()** gives them.

```

116: sub talk_to_server {
117:     my ($server, $port, $request) = @_ ;
118:     my @response = ();
119:     eval {
120:         my $alarm_timeout = 60;
121:         local %SIG;
122:         $SIG{ALRM} = sub { die "Error: timeout\n"; };
123:         alarm $alarm_timeout;
124:         my $socket = new IO::Socket::INET(PeerAddr =>
$server,
125:                                     PeerPort => $port,
126:                                     Proto  => 'tcp',
127:                                     Type   => SOCK_STREAM)
        or die "Couldn't connect to $server:$port: $!\n";
128:         print $socket $request;
129:         @response = $socket->getlines();
130:         close $socket;
131:         alarm 0;
132:     };
133:     if ($?) { return [ "Error: timeout: $@\n" ] }
134:     @response = @response[2..$#response];
        # discard first two lines
135:     return \@response;
136: }
137: }

```

**talk\_to\_server()** is the subroutine that actually does the dialogue with the manserver program. It's wrapped in an **eval()** block, from lines 119 to 133 – this allows us to set **SIGALRM** inside the block and do something sensible if the alarm times out and causes an exception. If we remove all the signal-handling stuff (which was explained in *Linux Format* 26) what we're left with is:

```

my $socket = new IO::Socket::INET(PeerAddr => $server,
                                PeerPort => $port,
                                Proto  => 'tcp',
                                Type   => SOCK_STREAM)
        or die "Couldn't connect to $server:$port: $!\n";
print $socket $request;
@response = $socket->getlines();
close $socket;

```

**\$socket** is an **IO::Socket::INET** object – a wrapper around a filehandle that is bound to a socket connected to **\$server** on port **\$port**. We simply print a message to it (to send a request), then read from it to see what the program at the other end sends us back. (**getlines()** reads all the lines from a filehandle until **eof()** is true.)

Note line 135:

```

135:     @response = @response[2..$#response];
        # discard first two lines

```

The manserver returns two things that aren't part of the response – a line saying "hello, I'm a manserver" when the client first connects, and a protocol response saying **200 - ok** or **500 - not ok** (or words to that effect) in response to the request. Line 135 naïvely and crudely throws this information away. If we wanted to do the job properly we wouldn't do that – we'd make some attempt to get the status line, then parse it, and only then



call `getlines()` (if necessary) to read the man page text.

```
139: sub do_help {
140:   my $parser = Pod::Text->new();
141:   $parser->parse_from_file($0, "-");
142: }
```

`do_help()` is a boilerplate help routine; the program has an `END` marker at the end of the Perl code, followed by its man page in POD documentation format. `do_help()` feeds the file `$0` (that is, the currently executing Perl script's source file) to `Pod::Text` and tells it to parse the stream, emitting text on the file handle `-` (a Unixy traditional shorthand notation meaning whatever file handle is bound to `STDOUT`).

## What next?

This is a pretty basic man page client. For starters, there's no option to return man page source files (that is, *troff* macro files using the *man* macro set). For seconds, there's no local caching mechanism; ideally *man* would maintain a small (of the order of 1MB) cache of files, and look there before going out over the net to pester a documentation server. And all good clients should set their exit status sensibly when they exit, in case they're invoked within a shell script.

We could get elaborate by adding extra functionality on the

server side, too. It's not hard to use one of the many html-to-text filters to provide access to the Linux HOWTO and Mini-HOWTO documentation, which lives in a set location (according to the Filesystem Hierarchy Standard).

Monitoring the response codes from the server is a must. For example, as written, this manserver client will not be able to deal intelligently with a server-side error. If the server hangs, the alarm call should enable the client to exit gracefully – but there's no way for it to deal with the server returning an error code indicating that there's no such man page or some other error has occurred.

Probably the worst issue with this client (and the most subtle) is that it is vulnerable to attack. Perl doesn't succumb to buffer overruns – it has dynamically resizable strings – but it is possible for an attacker who has subverted a server to force-feed the client a long string, eventually eating up all available memory on the client machine. A solution for this would be to replace `getlines()` with something more controllable (such as by calling `read()` to fill a buffer until `eof()` becomes true or a hard limit is exceeded).

However, this example illustrates the basics of exchanging data between a client and a server. Have fun, and remember: man pages are not the only fruit! [LXF](#)

## Spidering the web

### Data handling and recursive descendant parsers

Pulling stuff off a server is all very well, but it begs the most important question you need to ask when writing a network client: what do you do with the data when you've got it?

Processing data retrieved from a network server raises a bunch of questions. In general, there are about four types of data you're likely to be concerned with. Firstly, there's free-form text with no organising structure; objects pulled in from an FTP server often fall into this category. Secondly, there is regular structured data. Email or usenet message headers fall into this category, as do CSV files and a few similar things: you can make assumptions about them having a regular field-based layout with certain items stored in each field. Thirdly, there is nested data. The body of an email message containing MIME attachments may consist of a MIME wrapper around a bunch of included MIME sections. Similarly, an HTML file contains nested tagged text structured in accordance with a DTD. To handle nested data almost always requires a fairly sophisticated parser that recognizes the containing structure and builds a tree of elements. And finally there's binary data – which may or may not be amenable to processing in Perl (`pack()` and `unpack()` are your can-opening friends, here), and which often has an internal structure that imposes problems qualitatively equivalent to one of the other three types of data.

When messing around with network clients, we are mostly concerned with types two and three – regularly structured messages (such as email or usenet message headers, or HTTP headers) and nested structures (such as email attachments or HTML and XML files).

If you want to dig around inside data retrieved from a network server, you first need to know what you're dealing with. It may be that you're dealing with a mixture of data types. For example, if you establish an HTTP connection to a web server with keep-alive, you may receive multiple files via the same connection,

encapsulated within separate MIME-encoded messages. To deal with this sort of bundle, you need a two-stage process: first, to identify the separate message components (by parsing the HTTP response messages and separating out and decoding the attachments), and secondly to deal with individual components (for example, by parsing the HTML in a file into a tree representing a document's structure, which can then be searched for specific keywords, either identified as attributes of a META tag, or as plain text).

The difficulty of this job should not be underestimated. Any Perl developer working on this sort of project would do well to lay their hands on a copy of *Data Munging in Perl* by David Cross (ISBN: 1-930110-00-6, published by Manning: see [www.manning.com/cross](http://www.manning.com/cross)). This book is concerned with one of Perl's core tasks – taking raw data from a source (such as a server), manipulating or parsing it, and processing it into a final form. It includes a lot of valuable insight into processing nested formats like HTML and XML, and an introduction to `Parse::RecDescent`, the standard Perl module for building recursive-descendant parsers (which can cope with nested expressions).

The world wide web is a special case. Perl has a rich grab-bag of tools for serving up data under *Apache* and other web servers – and an equally rich client-side grab-bag of tools. The biggest and most powerful Swiss Army chainsaw in your toolbox is the CPAN **Bundle::LWP** package – LWP is short for “Lib-WWW Perl”, and you can install it by typing:

```
perl -MCPAN -e 'install Bundle::LWP;'
```

LWP contains two types of tool; gadgets for retrieving web pages from a remote server, and tools for parsing HTML. The first set of classes, grouped under `HTTP::Request`, treat an HTTP request as an object; you can set the method to use (`GET`, `PUT`, `POST`, or `HEAD`, as defined in the HTTP RFC's), the uri (universal resource indicator) denoting the object to retrieve, and additional headers. You execute an

`HTTP::Request` by passing the object to a User Agent (such as `LWP::UserAgent`), which creates the network connection to the server and handles communication-related aspects of the transaction, such as coping with timeouts and making use of proxy servers. When activated, the User Agent returns an `HTTP::Response` object, which provides access to a response code, HTTP response headers, and data returned from the server. For example:

```
use LWP::UserAgent;
# create a user agent
$ua = LWP::UserAgent->new;
$ua->agent("Test/0.1 ");

# Create a request
my $req = HTTP::Request->new(GET =>
'http://www.perl.com/index.html');

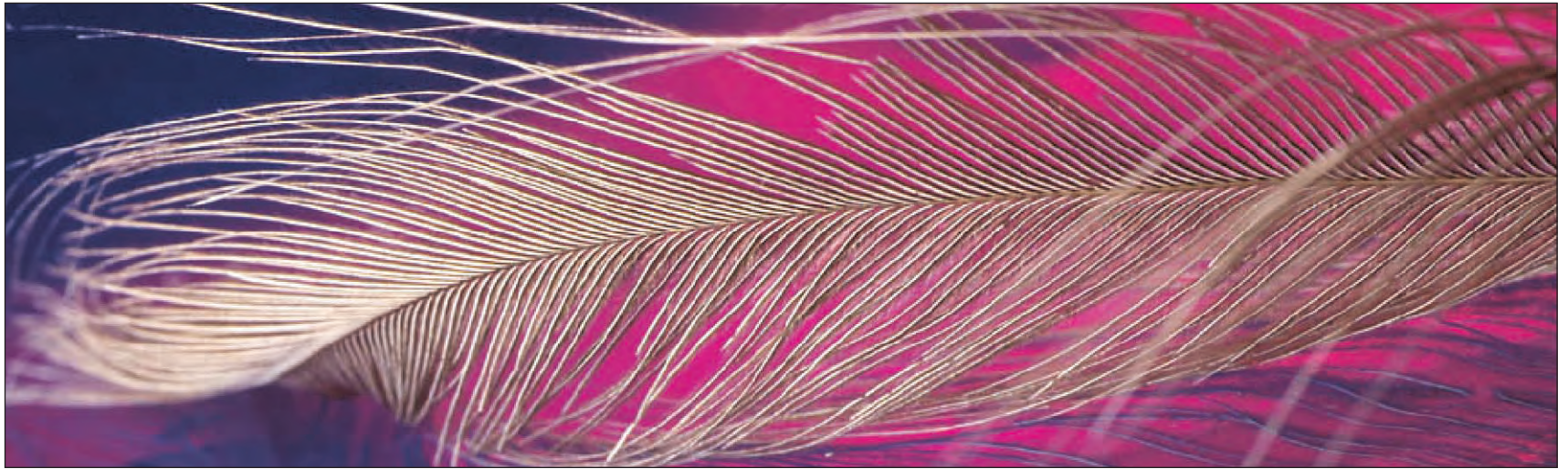
# Throw HTTP::Request at LWP::UserAgent and
return HTTP::Response
my $res = $ua->request($req);

# Check the outcome of the response
if ($res->is_success) {
    print $res->content;
} else {
    print "Bad luck this time!\n";
}

# get contents

my $html = $res->content;
```

What you do with the contents of the HTTP response once you've retrieved it is somewhat more difficult; a good starting point is Gisle Aas's module `HTML::Parser`, which can be found on CPAN and which provides a toolkit for building tools that parse HTML files and extract specified data from them – for example, all the link addresses, or all the text enclosed in `<H3>...</H3>` tags. We'll be looking at this in detail another tutorial.



THE WORLD WIDE WEBSERVER

# Apache webserver

**Chris Brown** demystifies the Apache configuration file and demonstrates name-based virtual hosting

Last month we looked at several ways of installing *Apache*: from your Linux distro CDs, from a binary RPM, and from source. Those of you persistent enough to make it to the end of that article will remember that we left our intrepid web server, *httpd*, up and running with a basic, default configuration. In this month's tutorial we'll examine the architecture of a web server in a bit more detail, and start to get to grips with *Apache*'s configuration file.

## The architecture of a web server

**Figure 1** shows the overall architecture of the *Apache* server. When the server is started, usually from a script run at boot time, it reads the `httpd.conf` file. This specifies, among many other things, the **DocumentRoot** directory, which is where *Apache* expects to find the HTML files it will serve. It can also specify the port on which *Apache* will listen for connections (port 80 by default), the user identity the server will run under, where it will write its log files, and a bunch of other stuff. The server will then settle down to wait for client connections. If it's the primary server for Microsoft's website, it probably won't have to wait long. (Although if I had to guess, I doubt Microsoft run their websites on *Apache*.) If, on the other hand, it's the website of the Carshalton Pregnant Mothers' Leapfrog Team, it may be waiting er, expectantly, for quite a while.

When a browser connects to the server it sends a request for a specific file, using a protocol called HTTP (Hyper Text Transfer Protocol). The server will look up the name of the file starting in the **DocumentRoot** directory. If it finds it, it will construct an HTTP response header, which it will send back to the browser prior to sending the contents of the requested file. Depending on how it's configured, it will then close the connection and wait for the next client.

## HTTP and HTML

To illustrate how all this works, we're going to try a little experiment. I'm hoping you'll follow along, in which case you should start by making sure your server is actually running. You'll also need to find which directory your server's config file is in.

That will depend on where your installation put it. Mine is in `/usr/local/apache2/conf`, but if you did a standard Red Hat install, for example, it will be in `/etc/httpd/conf`. One thing's for sure, it will be called `httpd.conf`, so if all else fails you can try:

```
find / -name httpd.conf
```

Now look in the config file to see where the **DocumentRoot** directory is. No, no ... don't scan it by eye, let *grep* do the work:

```
cd /usr/local/apache2/conf
```

```
grep '^DocumentRoot' httpd.conf
```

```
DocumentRoot "/usr/local/apache2/htdocs"
```

Of course, yours may be in a different place.

With the **DocumentRoot** defined as above, if I ask the server for the file `greeting.html`, it will look for `/usr/local/apache2/htdocs/greeting.html`.

There's a tradition amongst Linux developers, that grew up in the early days of Unix, when you found menus in restaurants, icons in churches, and mice in kitchen cupboards eating the soap, that our first programming effort should print "Hello world!". Whilst writing our first HTML file probably doesn't count as programming, it's a good tradition to continue. Let's do it: Here's the contents of `greeting.html`:

```
<HTML>
<BODY>
  Hello <B> bold </B> world!
</BODY>
</HTML>
```

The things in angle brackets are called tags, and they're defined by the Hyper Text Markup Language (HTML). When the file is received by the browser, it interprets these tags to determine the style and layout of the text. The **<B>** and **</B>** tags, for example, tell the browser to display the intervening text in a bold font. With this file in the **DocumentRoot** directory we can have our browser request it from the server, as shown in **figure 2**. Notice the effect of the **<B>** tag on the way the text is rendered.

Well, we probably won't get a job as a web designer on the basis of one HTML tag, but we're on our way – we've got the underlying mechanics working, and that's a good start.



## What's really going on?

We can get a more detailed picture of what's going on here by conducting the same dialogue using *telnet* as the client, instead of a browser. *Telnet* is usually used to connect to the *telnet* server on a remote machine, to obtain a command prompt login on that machine. However, if it's given a specific port number to use, it will connect to that port instead, and simply ferry lines of text back and forth between your terminal and the server. This makes it a useful tool to hold a conversation with any server that uses a text-based protocol. **Figure 3** shows a *telnet* dialogue with an *Apache* server which is essentially equivalent with the dialogue which occurred behind the scenes when we used the browser.

The **GET** command we see in this dialog is part of the HTTP protocol – a request for a specific file. The blank line that follows it is important – it marks the end of the request. The next block of lines are the HTTP response header, which is constructed by the server. The last line of this header (the one specifying the content type) tells the browser to interpret what follows as an HTML document and not, for example, a GIF or JPEG image. There then follows the contents of the file *greeting.html*, complete with HTML tags. Of course, *telnet* is blissfully unaware of the significance of either the HTTP header or the HTML tags, and just displays whatever it receives from the server as text. Finally, notice that *telnet* has reported that the connection has been closed by the web server.

Let's recap a couple of important points: The HTTP protocol is used to request files from the server and return them to the client. The files can be of several types, such as JPEG or GIF images, or text with embedded formatting tags. HTML is the mark-up notation used for such tags. It's not the server that knows how to interpret these tags, it's the browser.

## Apache's filesystem

The **ServerRoot** directive in the config file specifies the top-level directory of that piece of the file containing the files *Apache* uses at run time. There are a few directories under here worthy of note, including the *bin* directory, which contains several useful scripts and support programs. One of these, the script *apachectl*, provides a convenient interface to starting, stopping and restarting *Apache*, and checking the syntax of the config file. It doesn't do anything you can't do manually at the command line, but it hides a few details. Also in this directory you will find the executable of the *Apache* server itself.

By the way, (this is a true story) years ago I had a Unix machine whose disk was slowly and mysteriously filling up (we had a total of 10 MB in those days, split over 4 drives). It turned out that a colleague was under the mistaken impression

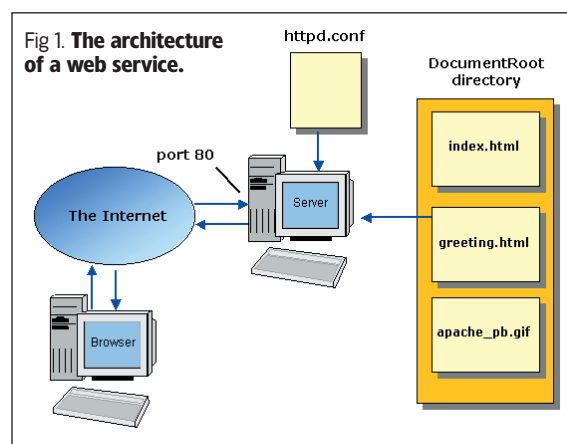


Fig 1. The architecture of a web service.

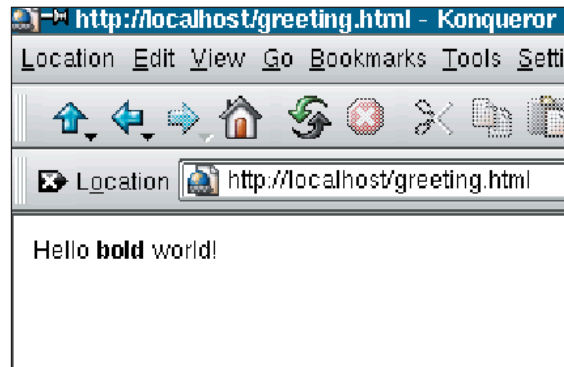


Fig 2. Displaying the greeting file in a browser.

that to delete files you moved them into */bin*. Once I had explained that it was bin-as-in-binary, not bin-as-in-rubbish, the problem went away.

Also underneath the **ServerRoot** directory you'll find the logs directory, where the runtime logs are kept, and the *conf* directory, where the configuration files live. *Apache* used to have three config files called *httpd.conf*, *access.conf* and *srm.conf*, but everything is merged into *httpd.conf* these days.

You'll also find the *cgi-bin* directory, a place where CGI scripts can be placed. (CGI scripting is one of several technologies for generating dynamic content on the server), and the *htdocs* directory, where the actual web content can be stored. Typically, however, since web content is controlled by the individual users who own the sites rather than a central web administrator, the content is not usually all gathered under this one place.

## Squaring up to the config file

*Apache*'s config file can be a bit intimidating at first, mainly because it's so long. However, most of it consists of nice helpful comments, and many of the settings have sensible defaults which means the server will work fine without them.

Another aside – also a true story. I once read an article in a popular programming magazine which claimed that the difficulty in understanding a program was proportional to the square of its length, so that a 2000 line program was four times as hard to understand as a 1000 line program – a not unreasonable claim, but the writer went on to suggest that since the inclusion of comments in the code made the programs longer, the way to make them easier to understand was to remove the comments. To this day, I do not know whether he was serious or not.

Anyway, back to the config file. Let's start by taking a look at some of the directives which control operation of the server as a whole. We've already talked about **ServerRoot** and **DocumentRoot**.

The **PidFile** directive specifies the name of the file (relative to the **ServerRoot**) into which *Apache* will write its process ID when it starts. This is used, for example, by the *apachectl* script to find and terminate the server.

In order to serve multiple clients at the same time, *Apache* services each client in a separate process. When it starts up, it pre-creates a number of these processes to save time later, and continues to maintain a pool of spare servers as more clients connect. There is a bunch of directives with names like **StartServers** and **MinSpareServers** which control this. Unless you're running a very busy server, and have some way to measure its performance under heavy load, you have no real basis for trying to tune these numbers and you'd be well advised to leave them alone and fiddle with something more interesting.

The **Listen** directive can be used to control which network



# LinuxFormatTutorialApache

◀ interfaces and ports the server will listen on. (The **Port** directive performs a similar function.) The standard port number for a web server is 80. There are a couple of reasons why you might want to specify an alternative port. First, if you're familiar with Linux networking you may know that only processes running with super-user privileges can listen on ports below 1024 – these are the so-called (and confusingly named) reserved ports. If you want to try *Apache* out and you don't have super-user access to the machine, you'll need to configure it to use a non-reserved port. 8080 is a conventional choice of port for this purpose; it's also used for caching web servers. Secondly, if you're going to set up the secure sockets layer (SSL), you'll need to have the server listen at port 443.

## Time for another experiment

First, try connecting to the server at port 8080 by entering the URL `http://localhost:8080` into your browser. It will probably complain that it can't connect, because there's no server listening at that port. Now, edit `httpd.conf` and add the two lines:

```
Listen 80
Listen 8080
```

You should delete any **Port** directives that may be present. To put the change into effect, you'll need to restart *Apache*. We'll use the `apachectl` script in the `bin` directory (under the server root) to do this, like so:

```
cd /usr/local/apache2/bin
./apachectl stop
./apachectl start
```

Now, if you point your browser at `http://localhost:8080`, it should work. You can also verify that the server is now listening on the new port with the command:

```
netstat -t
```

which tells you which tcp ports are listening for connections. You should see (among others) endpoints with names `http` and `webcache`, which are the names defined in the `/etc/services` file for ports 80 and 8080 respectively.

security model. Every process that runs under Linux runs with the identity of a particular user, and a particular group. (It is useful to think of it running on behalf of that user and group.) Every file in Linux is owned by a particular user and group. Each file has an "access mode" which allows or denies various types of access to processes running with (a) the identity of the file's owner, or (b) with the group identity of the file's group owner, or (c) neither of the above. There's one special user, conventionally called the super-user, or root, who can access anything. In addition to restrictions imposed by the file-based permissions, there are a few other operations that Linux doesn't allow unless the process requesting them is running with root's permissions. In particular, only a process running as root can create a TCP endpoint to listen for connections on a port number less than 1024 (and remember, web servers listen on port 80).

Now, this means that *Apache* needs to start running as root, to create the endpoint. However, it's not a good idea to let the server continue to run with root privileges, particularly if the site is set up to allow server-side scripts, that might get up to all kinds of mischief. So, once the endpoint is established, the server creates child processes which change their user and group identity so they are no longer running as root.

The identities used by the server are determined by the **User** and **Group** directives in the config file. For our next experiment, we'll create a new user and group especially for *Apache* to run as. First, find out who *Apache* currently thinks it is, by looking at the identity of the running processes. A command like

```
ps u -C httpd | head
```

should do the trick. You should see that the first (parent) `httpd` process is running as root, but its many children are all running as somebody else – probably the user "nobody", but this depends on how your default *Apache* installation is set up. Now create a new user and group:

```
adduser webuser
```

(On some systems this creates both a user and a group with the same name, on others, you might need to put the group in manually, with `addgroup`.) Now edit the config file, adding or changing the **User** and **Group** directives to specify 'webuser' as both. Stop and restart the server, as before. Now re-run the `ps` command. This time, you should see that all the child processes are running as 'webuser'. We'll see how we can use this new user identity to limit access to the site in the section on virtual hosting, below.

In addition to this basic piece of configuration, there are several mechanisms in *Apache* to limit access to the site, or parts of the site, based on where the browser is running, or on a password controlled login against a database of user accounts held by the server. We'll talk about those next month.

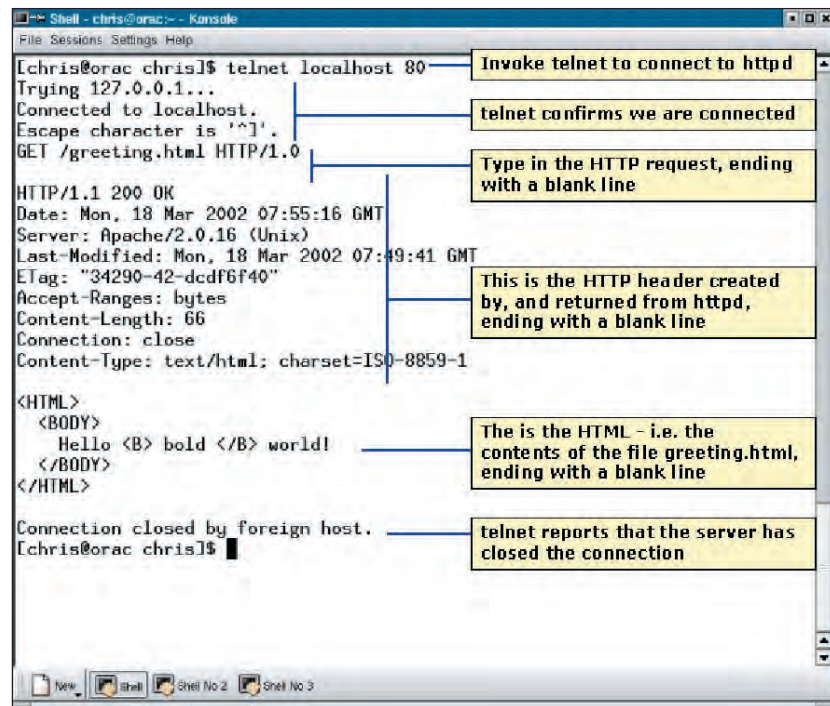
## Virtual Hosting

Some *Apache* servers sit on corporate intranets and serve a single site, accessed via a single host name. However, web servers operated by ISPs generally want to host many sites on one machine. Let's suppose that, in addition to hosting the site for the Carshalton Pregnant Mother's Leapfrog Team at `www.leapfrog.co.uk`, we'd also like to run sites for the Hillman Minx Owners' Society at `www.minx.org` and the Institute of Sumo Wrestlers Glee Club at `www.sumosing.com`.

Note to our younger readers: The Hillman Minx was a car very popular in the 1950s which, even to my childish eye of the time, was lacking in style. Not the kind of thing James Bond would drive. If you would like to see a picture of one, visit `www.charm.net/~pdbragg/whatsahillman`.

Serving more than one site off a single server requires the co-

Fig 3. *Telnet* session to web server.





operation of DNS (so that the URLs for all the sites reach your machine) and a trick in *Apache* called virtual hosting.

## IP-based Virtual Hosting

The oldest, simplest and least satisfactory way of doing virtual hosting is to establish multiple IP addresses for one machine. This is called IP-based virtual hosting. It's easy to attach more than one IP address to a single network interface in Linux, using the *ifconfig* command, and it's easy to set up virtual hosts in the *httpd.conf* file to serve different content depending on which IP address the browser used to connect with. The trouble is, in these days of rapidly dwindling stocks of unassigned IP addresses, the world needs a way of squandering multiple addresses on a single machine about as much as it needs a hole in the head. Consequently, IP-based virtual hosting is strongly deprecated by the Internet community, and we won't consider it further.

## Name-based Virtual Hosting

Name-based virtual hosting distinguishes between multiple virtual sites based on the site's name, not on its IP address. **Figure 4** shows how it works. DNS is configured so that the names for all the sites you're hosting point to the same IP address – your machine.

For virtual hosting to work, the browser has to support version 1.1 of the HTTP protocol. One of the features defined in this version is that the browser sends a **HOST** line following the **GET** request, to identify the name of the host it's fetching the file from. *Apache* uses this to serve different content depending on which host name is used.

Name-based virtual hosting is a much more satisfactory solution because it uses only one IP address. The down side is that it relies on the browser to pass the server name as part of the **GET** request. This is a standard part of HTTP/1.1, but there are a few old browsers around that don't support it. Tough.

## Configuring Virtual Hosting

As you'll have guessed, virtual hosting is configured by entries in *httpd.conf*, which will look something like this:

```
NameVirtualHost 192.168.1.76

<VirtualHost 192.168.1.76>
    ServerName www.leapfrog.co.uk
    DocumentRoot /home/leapfrog/html
</VirtualHost>

<VirtualHost 192.168.1.76>
    ServerName www.minx.org
    DocumentRoot /home/minx/html
</VirtualHost>

<VirtualHost 192.168.1.76>
    ServerName www.sumosing.com
    DocumentRoot /home/sumosing/html
</VirtualHost>
```

There are a few things we need to understand about this. First, the **NameVirtualHost** directive says that requests arriving at this IP address will be resolved using the named virtual hosts mechanism. Second, the rather XML-like tags **<VirtualHost>** and **</VirtualHost>** define what is called a container. Within the container, directives apply only to that virtual host. Within each of our containers, we define the name of the site, and a new **DocumentRoot** which tells the server where the content for that specific site is to be found. Note that because these directory names start with a slash, they are taken to be full pathnames.

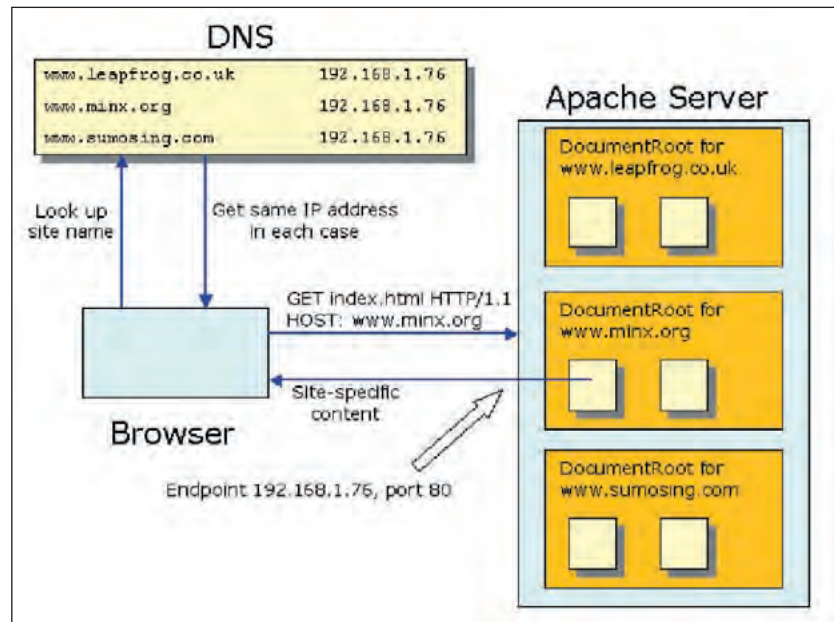


Fig 4. Name-based virtual hosting.

The server does not look in the **ServerRoot** directory.

Implicit in the directory names I chose here is a specific administrative organisation; namely, that the three sites are maintained by three users *leapfrog*, *minx*, and *sumosing*, who each have home directories under */home* and who each put their html files into subdirectories called *html*. It is common when setting up *Apache* to move the site contents out of the **ServerRoot** filespace, into user directories.

If you'd like to do a final experiment this week, you can set virtual hosting up fairly easily. First, you can fake the DNS entries by putting entries into your */etc/hosts* file, something like this:

```
192.168.1.76 www.minx.org
192.168.1.76 www.leapfrog.co.uk
192.168.1.76 www.sumosing.com
```

Now create the three users:

```
adduser minx
adduser leapfrog
adduser sumosing
```

Now log on as each user in turn, create a subdirectory called *html*, and in that directory place a file called *index.html* with some content specific to that site.

Add the **NameVirtualHost** directive and the **<VirtualHost>** containers to the *httpd.conf* file. Restart *Apache*.

Now we need to be a bit careful with the file permissions. Remember that *Apache* is running with the user and group identity 'webuser'. The idea is to give *Apache* read-only access to the html files, to give the user, such as *minx*, read/write permission, and nothing else. Here's how to set the ownerships and permissions, using *minx* as an example:

Permissions	Owner	Group	File
rw-r-x---	minx	webuser	/home/minx
rw-r-x---	minx	webuser	/home/minx/html
rw-r-----	minx	webuser	/home/minx/html/index.html

Now if you point your browser at <http://www.minx.org/index.html>, you should see the content that you defined for that site. Notice that we're not really using the group *webuser* as a group – it only has one member. We're using it so that we can independently set access permissions on the files for two separate entities – the site's owner, such as *minx*, and the *Apache* server.

Well, I'll leave you in peace now to get that lot debugged!

# NEXT MONTH

We'll take a look at how to limit and control access to a web site, and how to log the hits it receives. If you like all this syntax-rich config stuff, you'll love next month.

## CONTROLLING SERIALIZATION

# Speaking Java

Find out how to customize the way your classes are serialized as **Richard Drummond** delves deeper into Java's serialization mechanism.



As we saw last time, Java's default serialization algorithm serializes all the fields of a class that are non-transient, instance fields. So, one way to control which fields of a class are serialized is to flag the ones you don't wish to be serialized with the **transient** modifier. This is fine as far as it goes, but it lacks finesse.

Another way to take control is to explicitly declare which fields are to be serialized. We do this by creating a static field called **serialPersistentFields**, which is a reference to an array of **ObjectStreamField** objects, which describes the fields that are to be serialized (the constructor for the **ObjectStreamField** class takes a field's identifier as a **String** and class as a **Class**). The order of fields here is significant: the fields will be written and read in the order that they are specified in the array.

The following example should help to make this more clear. The class **MyUser1** models the user of some software system and contains the self-explanatory fields **userName**, **password** and **realName**. When an object of this class is serialized, for security reasons, we don't want the **password** field to be written out using the default serialization algorithm because it will be visible to anybody with a hex editor.

```
class MyUser1 implements Serializable
{
    String userName;
    String password;
    String realName;

    // declare serializable fields
    private static final ObjectStreamField[] serialPersistentFields =
    {
        new ObjectStreamField( "realName", String.class ),
        new ObjectStreamField( "userName", String.class ),
    };

    // rest of the class . . .
}
```

Objects of **MyUser1** class can still be serialized as per normal with **ObjectOutputStream.writeObject()** and **ObjectInputStream.readObject()**. This time, when the serialization mechanism inspects the class, it sees the field **serialPersistentFields** and uses the fields it finds there – rather than interrogating the structure of the class itself.

The above doesn't seem to offer any real benefit over simply flagging fields as **transient** – other than letting you specify the order in which fields are serialized. However, the fields specified in **serialPersistentFields** are the fields that will externally represent the state of an object of your class. They don't actually have to exist internally in the class, as we will see later.

## The missing piece?

The above example solves part of the problem, but how do we serialize the field **password**?

Well, we can override the default serialization algorithm by implementing the following instance methods in our class to handle serialization and deserialization, respectively:

```
private void writeObject(java.io.ObjectOutputStream out )
    throws IOException;

private void readObject(java.io.ObjectInputStream in ) throws
    IOException, ClassNotFoundException;
```

When these methods exist in a class – with these exact method signatures – the serialization mechanism will use them in preference to the default algorithm. Additionally, the class **ObjectOutputStream** supplies the method **defaultWriteObject()** and **ObjectInputStream** supplies **defaultReadObject()** which can be called by the above methods to perform the standard serialization algorithm. This means we can, for example, override **writeObject()**, call **defaultWriteObject** to write the serializable fields as per normal and then implement whatever custom serialization we need (typically with calls to the **writeX** methods in **ObjectOutputStream**).

Have a look at this example. This is another attempt at modeling a user, but this time the **password** field is serialized. We implement **writeObject()** and **readObject()** to override the default mechanism, but invoke **defaultWriteObject()** and **defaultReadObject()**, respectively, to perform the default serialization, in this case, of the fields **userName** and **realName**.

The custom handling of **password** is performed by calls to **readObject()** and **writeObject()**, but here we wrap the **String** with calls to some hypothetical methods **encrypt()** and **decrypt()** to ensure that the external representation of the password field is encrypted.

```
class MyUser2 implements Serializable
{
    String userName;
    transient String password; // don't serialize by default
    String realName;

    private void writeObject(java.io.ObjectOutputStream out )
        throws IOException {
        out.defaultWriteObject();
        out.writeObject( encrypt( password ) );
    }

    private void readObject(java.io.ObjectInputStream in ) throws
        IOException, ClassNotFoundException {
        in.defaultReadObject();
        password = decrypt( (String) in.readObject() );
    }
}
```

This solution seems to solve our problems, but it does have one shortcoming. Overriding the serialization mechanism in this way breaks the serialization mechanism's versioning system (which we talked about briefly last time), so problems



will occur if we wish to evolve this class at all.

We can take things one step further by using the `Serializable` Fields API. This allows us exert even more control over how the serialization mechanism serializes our class.

## The Serializable Fields API

When writing the object in our private `writeObject()` method, we can retrieve a reference to the buffer that the serialization mechanism uses for storing the fields prior to writing them by invoking the method `putFields()` on the `ObjectOutputStream()` in question. We can then populate this buffer with our own data to be serialized and write it out with a call to `writeFields()`.

The `putFields()` method returns the cache as a reference to an object of `ObjectOutputStream.PutField()` class. This class provides us with access to the named fields that will be written to the stream. We can fill in the values of the various fields with its `put( name, value )` methods. Here **name** is the field's identifier and **value** is the data to be stored in that field. This method is overloaded to cover storing the values of various primitive types and of objects in the buffer. The fields named in `put()` must exist: that is, they must be declared in the `serialPersistentFields` array if you are explicitly declaring the serializable fields, or they must be the default serializable fields of your class.

To look at the flip side of the coin, deserializing is similar.

`ObjectInputStream.getFields()` returns a reference to an object of `ObjectInputStream.GetField` class containing the fields read in from the stream. We can access the values stored here with a series of `get()` calls and store them in our instance fields.

We reimplement the `writeObject()` and `readObject()` methods of our example class as follows:

```
private void writeObject(java.io.ObjectOutputStream out )
throws IOException {
    ObjectOutputStream.PutField fields = out.putFields();
    fields.put( "userName", userName );
    fields.put( "realName", realName );
    fields.put( "password", encrypt( password ) );
    out.writeFields();
}

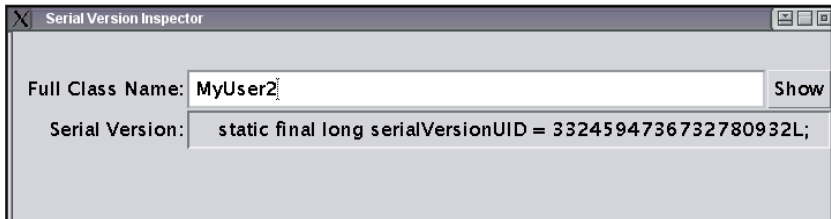
private void readObject(java.io.ObjectInputStream in ) throws
IOException, ClassNotFoundException {
    ObjectInputStream.GetField fields = in.readFields();
    userName = (String) fields.get( "userName", "" );
    realName = (String) fields.get( "realName", "" );
    password = decrypt( (String) fields.get( "password", "" ) );
}
```

This solution does support the versioning mechanism.

## Evolving a custom class

What happens when wish to evolve a class that performs custom serialization? With the `Serializable` Fields API, we can still rely on the serialization mechanism's built-in versioning. That is, we can maintain compatibility by explicitly declaring the serial version ID.

We can do more, however. Since we have full control over the serialization, we can even maintain compatibility when we change the fields of class. Suppose that, for some bizarre reason, we wished to modify our example class such that instead of storing the user's real name in a field called **realName** we want to instead store a user's **forname** and **surname** separately in the fields **forname** and **surname**. Changing the fields in a class will change its default serial API, so to retain compatibility we have to specify the serial ID from the previous version (obtained with the **serialver** command). However, since we now have an extra field to be serialized, this still won't work.



What we can do however is to retain the old external representation of the class, and just use the new fields internally. We can declare the old fields in the `serialPersistentFields` API and map between the old and new fields in the `readObject()` and `writeObject()` methods like this:

```
class MyUser4 implements Serializable
{
    String userName;
    String password;
    String forname; // changed from original class
    String surname; // changed from original class

    // these are the fields that will be stored externally
    private static final ObjectStreamField[] serialPersistentFields =
    {
        new ObjectStreamField( "userName", String.class ),
        new ObjectStreamField( "realName", String.class ),
        new ObjectStreamField( "password", String.class )
    }

    // SUID obtained from original class
    static final long serialVersionUID = 9030593813711490592L;

    private void writeObject(java.io.ObjectOutputStream out )
    throws IOException {
        ObjectOutputStream.PutField fields = out.putFields();
        fields.put( "userName", username );
        fields.put( "realName", forname + " " + surname );
        fields.put( "password", encrypt( password ) );
        out.writeFields();
    }

    private void readObject(java.io.ObjectInputStream in ) throws
    IOException, ClassNotFoundException {
        ObjectInputStream.GetField fields = in.readFields();
        userName = (String) fields.get( "userName", "" );
        String tmp = (String) fields.get( "realName", "" );
        forname = getForname( tmp );
        surname = getSurname( tmp );
        password = decrypt( (String) fields.get( "password", "" ) );
    }
}
```

Notice that now when serialize, we concatenate the internal **forname** and **surname** fields and write them externally as the **realname** field. When deserializing, we do the opposite. We read in the the value of **realname** and split it into **forname** and **surname** with the hypothetical methods `getForname()` and `getSurname()`, respectively. [LXF](#)

The **serialver** tool can be used to obtain the serial version ID of a compiled class.

## Serialization reference

For more information on serialization, see the Object Serialization section of Sun's Java web pages at <http://java.sun.com/j2se/1.3/docs/guide/serialization/>. This contains the serialization specification, a FAQ and lots of useful examples.



## EXCEPTION HANDLING

# Raising exceptions

## PART 9 Brian Long continues to take exceptions.

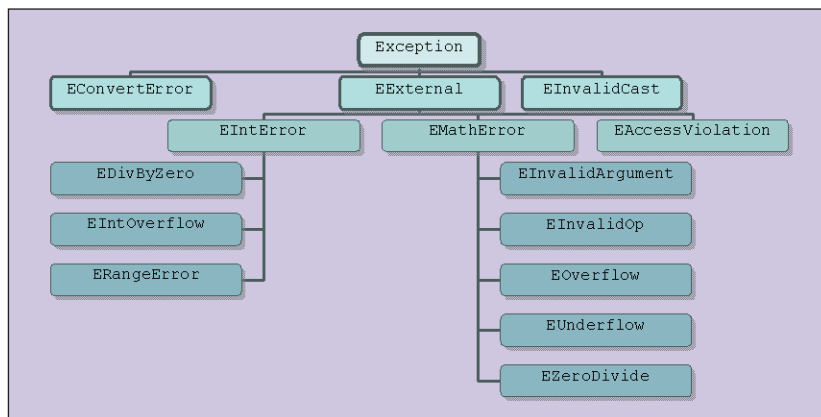
Last month we saw how the **try..except** statement could be used to catch exceptions from a specified block of code. This month we will see how to broaden the scope of an exception handler and how to raise exceptions of our own. We will also look at resource protection, a necessary consequence of Object Pascal's support of exceptions.

### Exception Handling Statements

We have seen that any exceptions emanating from the **try** part of the **try..except** statement could be picked up by appropriate exception handlers in its **except** part, where you can have as many exception handlers as you like. You can incorporate an optional **else** section as well for other unspecified exceptions.

```
try
  //code that may cause exceptions
except
  on ExceptionType1 do
    // ExceptionType1 handler
  on ExceptionType2 do
    // ExceptionType2 handler
  //and so on
  else
    //handle any other exception
```

Figure 1: A branch of the exception class hierarchy.



end

Any given exception handler will react to exceptions from the specified exception class or from classes inherited from it. So the first handler in the code above would handle exceptions that are either of type **ExceptionType1** or inherited from this type.

This might sound no big deal until you realise that many exceptions are organised in branches of the **CLX** class hierarchy. **Figure 1** shows a small portion of the exception class hierarchy (the top-level class **Exception** inherits directly from **TObject**). Looking back at the exceptions we saw last time you can see that **EConvertError** and **EInvalidCast** are immediate descendants of **Exception**. On the other hand **EAccessViolation** inherits from **EExternal**, which is inherited from **Exception**, and **EDivByZero** inherits from **EIntError**, which itself also inherits from **EExternal**.

In fact you can see that **EIntError** is the base class for all integer exceptions (including the integer divide by zero exception, **EDivByZero**) and **EMathError** is the base class for all floating point (fp) exceptions (including the fp divide by zero exception, **EZeroDivide**). Thus you can write exception handlers that are more generic than for just handling an individual error. You could write an exception handler to, say, handle all integer arithmetic exceptions that might occur in a block of code like this example:

```
try
  //code that performs integer arithmetic
except
  //handle all integer arithmetic errors
  on EIntError do
    ShowMessage(
      'An unexpected error occurred during some ' +
      'integer arithmetic. Sorry about that!')
end;
```

To handle one particular exception in a branch, but to handle the rest generically, the specific handler goes first, as in:

```
try
  //code that performs integer arithmetic
except
  on EDivByZero do
    ShowMessage('A divide by zero error occurred. Oops!');
  //handle all other integer arithmetic errors
  on EIntError do
```



```
ShowMessage(
  'An unexpected error occurred during some ' +
  'integer arithmetic. Sorry about that!')
end;
```

In fact, you can take this a bit further and catch any exception that may occur in a block of code by writing a handler for the **Exception** class. This is generally considered bad practice unless you are catching all exceptions to perform some action and then chain back to the previous error handler (the same applies to an else clause in the except part of a **try..except** statement).

```
try
  //some code statements
except
  on E: Exception do
    begin
      LogException(E); //fictitious logging routine
    raise
    end
end;
```

If you have no need to access the exception object, you can use an abbreviated version of the **try..except** statement that has no exception handlers at all. For example, you might need to set some variable or property to **False** if an error of any description occurs, but don't want to affect the natural exception handling:

```
try
  //some code statements
except
  //set some flag variable to False
  raise
end
```

*NB* external hardware errors (propagated to *Kyl* applications through signals and represented by exception classes inherited from **EExternal**) are not well handled in *Kyl1* (they tend to cause the app to hang). As a consequence our advice is to install *Kyl 2 OE* to get proper recoverable signal support in *CLX* apps.

## App-wide exception handlers

It is common to write blocks of code and then write error handling for the errors that may occur in that code block. But sometimes you want to change the default response to certain kinds of exceptions throughout the entire app – to alter the implementation of the default application-wide exception handler.

Fortunately this is a straightforward proposition as the helpful **Application** object has an event that fits the bill. The **OnException** event provides an opportunity for the developer to *replace* the default exception handling code that normally catches any unhandled exception.

In the normal course of events *Kyl* event handlers have no obligation to anything in particular. They can do as little or as much as the programmer wants and it makes no odds to the operation of the application. This scheme is sometimes called *contractless event programming*, as you are not required to do any specific tasks if you make an event handler.

Occasionally you bump into an event that breaks this general trend, such as the **Application** object's **OnException** event. As soon as you set up an event handler for this event, the normal response to unhandled exceptions (the simple message box) is disabled. If you make an event handler for this event, then you are expected to perform some sensible handling of any exception that comes along to ensure the app behaves in an orderly way.

The default response to an unhandled exception is to display its description in a message box (see **figure 2**), so you could do much the same in your replacement application-wide exception handler. The **Application** object uses its own **ShowException**

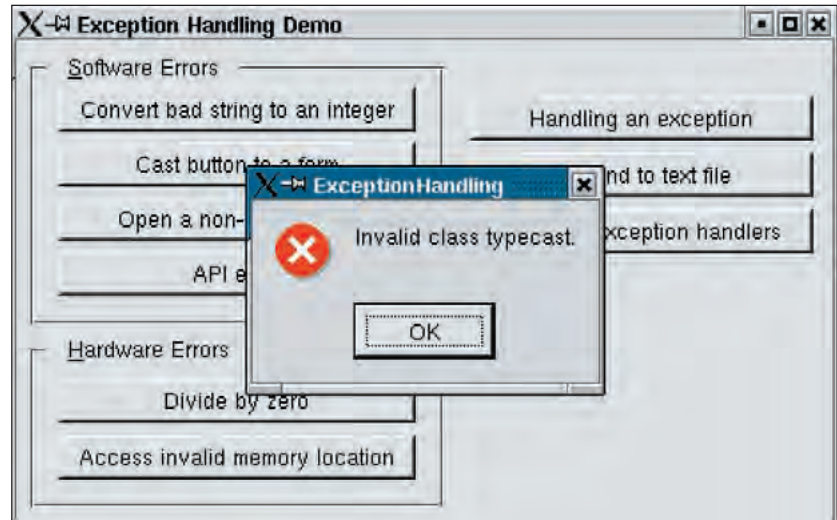


Figure 3: The event that acts as an application-wide exception handler.

method to achieve this but you can use any mechanism you like to display the message (and perhaps display a little more info).

Before we get onto such details we should look at how to set up the event handler. Unfortunately the **Application** object does not show up at design-time and so we will need to manually set up a method and turn it into an event handler.

The online help (shown in **figure 3**) confirms that the event is a property of type **TExceptionEvent**, which represents a procedure method with two parameters. The first parameter, **Sender**, will typically be the **Application** object and the other parameter, **E**, is the unhandled exception.

The code below (from the *ExceptionHandling.dpr* project on the disc) shows the method **AppExceptionHandler** being set up as this event handler. The method is implemented and then assigned to the **Application's OnException** property in the main form's **OnCreate** event handler. All **AppExceptionHandler** does in this case is to replicate the default behaviour by passing the unhandled exception to a call to **Application.ShowException**.

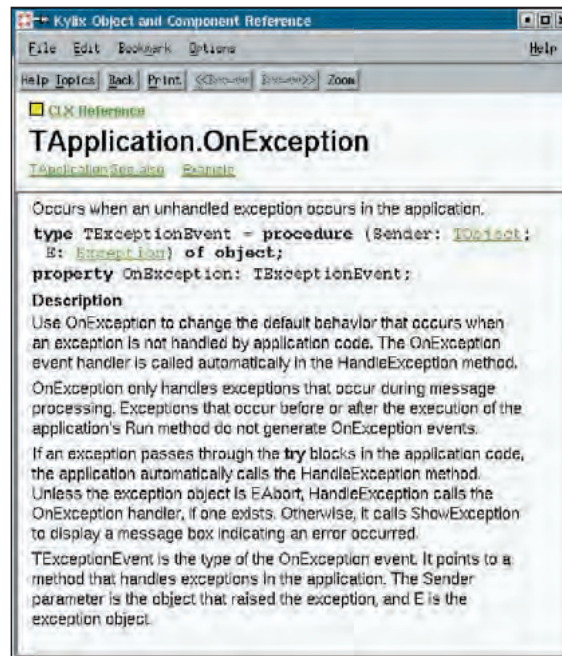
```
type
  TForm1 = class(TForm)
  ...
  procedure FormCreate(Sender: TObject);
  ...
  public
    procedure AppExceptionHandler(Sender: TObject; E:
      Exception);
  end;
  ...
  procedure TForm1.AppExceptionHandler(Sender: TObject; E:
    Exception);
  begin
    Application.ShowException(E)
  end;
  ...
  procedure TForm1.FormCreate(Sender: TObject);
  begin
    Application.OnException := AppExceptionHandler;
  end;
```

Of course you could use a different message box if you like. **ShowException** is implemented using a call to **Application.MessageBox** but you could equally use a member of the **MessageDlg** or **ShowMessage** families. In addition you could provide more information than the default message.

Normally the exception's **Message** property is displayed. This is descriptive, but whilst developing an application it might be



Figure 2: The normal application response to an unhandled exception.



◀ handy if the message also told you its exception class. If you actually need to handle the exception you will need to know this information anyway, so the default handler may as well display it.

Every *Kylix* object has a **ClassName** method that returns the class name as a string (it's defined in **TObject**). The following displays the message and class name, as you can see in **fig 4**:

```
procedure TForm1.AppExceptionHandler(Sender: TObject; E:
Exception);
begin
  MessageDlg(
    Format('An %s occurred: '#10#10'""s"', [E.ClassName,
    E.Message]),
    mtError, [mbOK], 0)
end;
```

Another common job for the application-wide exception handler is to isolate certain classes of exceptions for special-case handling. *E.g.* you might want to prevent your users from seeing:

Access Violation at address XXXXX, accessing address YYYYY.

You could do this by trapping **EAccessViolation** exceptions and displaying a less alarming message as shown in **fig 5**, thus:

```
procedure TForm1.AppExceptionHandler(Sender: TObject; E:
Exception);
begin
  if E is EAccessViolation then
    MessageDlg('A small technical hitch has occurred. '#10 +
```

Figure 4: More information in the default exception handler.

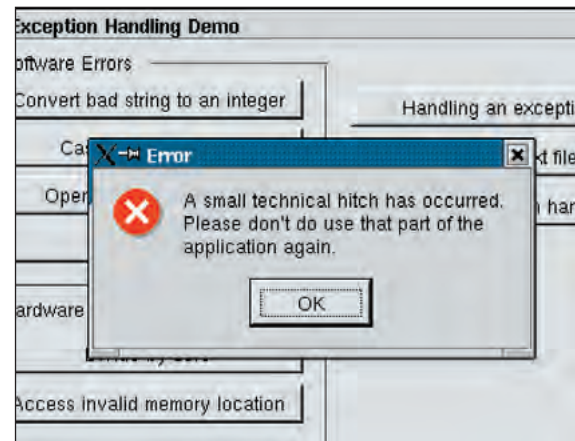
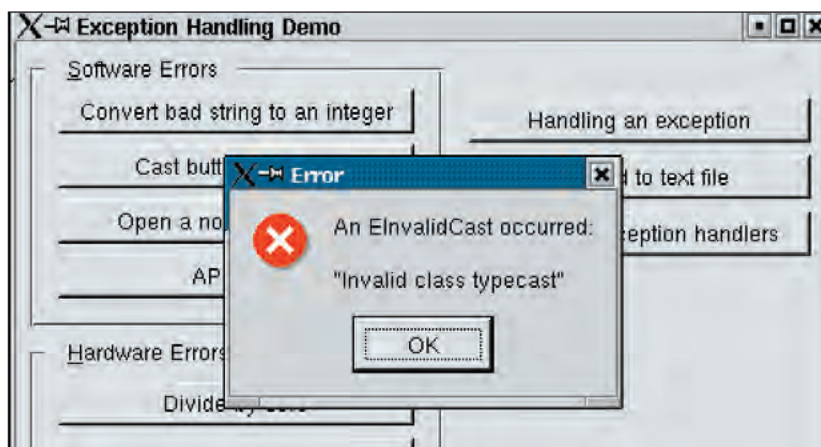


Figure 5: A different response to any Access Violation across the application.

```
'Please don't do use that part of the application again,'
mtError, [mbOK], 0)
else
  MessageDlg(
    Format('An %s occurred: '#10#10'""s"', [E.ClassName,
    E.Message]),
    mtError, [mbOK], 0)
end;
```

A more likely rôle of this type of event handler would be to log unhandled exceptions experienced by the app in some sort of log file (such as the one shown in **fig. 6**). This might give support or QA people a chance to see what type of errors are occurring most often, so the developers could be notified of some work to be done. The following code contains the additions needed to get simplistic error logging support (note the conditional compilation to cater for Linux and Windows file names, in the interest of supporting cross-platform compilation with *Kylix* and *Delphi*).

```
procedure TForm1.AppExceptionHandler(Sender: TObject; E:
Exception);
var
  Log: TextFile;
const
  {IFDEF LINUX}
  LogFile = '/tmp/CLX_app_error_log.txt';
  {ENDIF}
  {IFDEF MSWINDOWS}
  LogFile = 'C:\Temp\CLX_error_log.txt';
  {ENDIF}
begin
  if E is EAccessViolation then
    MessageDlg('A small technical hitch has occurred. '#10 +
    'Please don't do use that part of the application again,'
    mtError, [mbOK], 0)
  else
    //Default response to unhandled exceptions
    //Application.ShowException(E)

    //Custom response to unhandled exceptions
    MessageDlg(
      Format('An %s occurred: '#10#10'""s"', [E.ClassName,
      E.Message]),
      mtError, [mbOK], 0);
    AssignFile(Log, LogFile);
    if FileExists(LogFile) then
      Append(Log)
    else
```



```

Rewrite(Log);
WriteLn(Log, E.ClassName, ' ', E.Message, '): ',
DateTimeToStr(Now));
CloseFile(Log);
end;

```

## Raising Exceptions

We now know the ins and outs of handling exceptions, which is great, but what about generating exceptions of our own to indicate problems? If you think about it, exceptions occur in *Kyl* when you execute statements and such statements often involve calling RTL routines, VCL routines or object methods.

Last month we saw that exceptions take the responsibility of checking for errors away from the programmer, and that is true for the most part. However, when implementing all those commonly used library routines, at some point some checking has to be done. These library routines check their input or behaviour, and when a problem is detected they raise an exception. The benefit of this is that they do the checking and your normal processing code uses exception handling.

When implementing your own reusable routines it would be prudent to represent any issues by raising your own exceptions rather than returning error values. As an example, let's say you wanted to write a routine that does some calculations on some numerical input. The routine is designed to take an arbitrary number of floating point values but has some specific design constraints. Let's say that to operate successfully it must have at least two values passed to it and all values must be non-zero.

Using traditional coding you might implement the routine like this; returning special values (-1 and -2 in this case) to indicate failure. The function expects what's called an *open array* of doubles to be passed in. This is an array of double-precision floating point numbers, but with no dictated size (the array can be as large or as small as you like). When calling the function you can pass any existing array variable that is defined to contain Double elements or you can pass an open array without using a variable at all. An open array is made by surrounding a comma-separated list of expressions in square brackets.

```

function CalcFloats(Floats: array of Double): Double;
var
  I: Integer;
begin
  if Length(Floats) < 2 then
  begin
    Result := -1;
    Exit;
  end;
  Result := 1;
  for I := Low(Floats) to High(Floats) do
  begin
    if Floats[I] = 0 then
    begin
      Result := -2;
      Exit;
    end;
    Result := Result * Floats[I]
  end
end;

procedure TForm1.Button9Click(Sender: TObject);
var
  Radius, Res: Double;
begin
  Radius := 10; //this should be read from the user

```

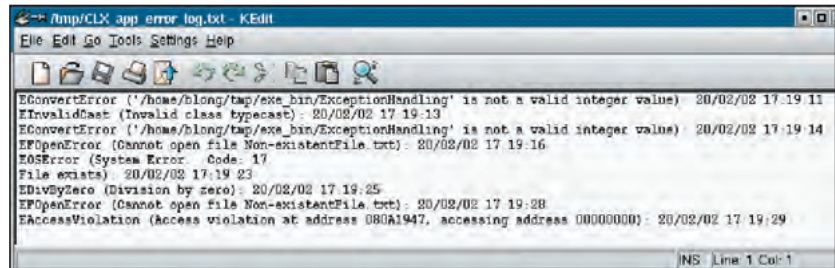


Figure 6: A simple exception log file.

```

Res := CalcFloats([Pi, Radius, Radius]);
if Res < 0 then
  ShowMessage('There was an error, code %g', [Res])
else
  ShowMessage('The answer was %g', [Res])
end;

```

You can see some code that uses the routine, passing three double expressions to **CalcFloats**. It is obliged to check for error values being returned before using the calculated return value.

Contrast the following version of the routine that raises exceptions to represent input problems. The code that uses the function's result has no need to check for an error. If one occurs during the call to **CalcFloats** the execution path will immediately jump to the closest error handle, which by default is the app error handler that automatically displays the error message.

```

function CalcFloats(Floats: array of Double): Double;
var
  I: Integer;
begin
  if Length(Floats) < 2 then
    raise EMathError.Create('Too few arguments');
  Result := 1;
  for I := Low(Floats) to High(Floats) do
  begin
    if Floats[I] = 0 then
      raise EMathError.Create('Arguments must be non-zero');
    Result := Result * Floats[I]
  end
end;

procedure TForm1.Button9Click(Sender: TObject);
var
  Radius, Res: Double;
begin
  Radius := 10; //this should be read from the user
  Res := CalcFloats([Pi, Radius, Radius]);
  ShowMessage('The answer was %g', [Res])
end;

```

In this particular case a predefined error class (**EMathError**) was used but you can also define your own exception class types by inheriting from existing exception classes. This allows you to design your own mini-exception hierarchy to perform specific handling of your own personal exceptions. Below we use custom exceptions to represent each problem that can be reported.

```

type
  ELXError = class(Exception);
  ELXMathError = class(ELXError);
  ELXZeroArg = class(ELXError);
  ELXTooFewArgs = class(ELXError);
...
function CalcFloats(Floats: array of Double): Double;
var
  I: Integer;
begin

```

**ORDER  
KYLIX 2  
NOW!**

You can purchase *Kyl* 2 from Borland at [www.borland.com/kyl](http://www.borland.com/kyl) or telephone your order on 0800 454065. Fax orders to 0800 454066. Current pricing and order information can be found on the Borland website.



```

<< if Length(Floats) < 2 then
    raise ELXFTooFewArgs.Create('Too few arguments');
Result := 1;
for I := Low(Floats) to High(Floats) do
begin
    if Floats[I] = 0 then
        raise ELXFZeroArg.Create('Arguments must be non-zero');
    Result := Result * Floats[I]
    end
end;

```

Figure 7 shows how you can trap any custom mathematical error coming from the **CalcFloats** call:

```

procedure TForm1.Button9Click(Sender: TObject);
var
    Radius, Res: Double;
begin
    Radius := 10; //this should be read from the user
    try
        Res := CalcFloats([Pi, Radius, Radius]);
        ShowMessage('The answer was %g', [Res])
    except
        on E: ELXFMathError do
            ShowMessage(E.Message)
        end
    end;
end;

```

One last thing to mention about raising exceptions is the **CLX Abort** procedure, which simply raises an **EAbort** exception. The application default exception handler completely ignores **EAbort**, meaning it never produces a message box and does not get delivered to the **Application** object's **OnException** event.

This fact is why **EAbort** is sometimes referred to as a "silent" exception. You can use a call to **Abort** to break the natural execution path. If no **try..except** statement catches it the **Application** object will pick it up and kill it in the same way it does for other unhandled exceptions, but without any intrusive dialog. So **Abort** can effectively stop an event handler in its tracks and "reset" the app to wait for the next event to handle.

The *Kylx* debugger defaults to ignoring **EAbort**, even if it intercepts all others. You can specify any number of exceptions that you want the debugger to ignore on the Language Exceptions page of its options dialog (Tools | Debugger Options...). Just press **Add...** and type in the name of an exception class and it will be added to a list of ignored exceptions (see figure 8).

## The Problem With Exceptions

The one problem of exceptions is with resource protection. Their very nature causes execution to divert from its natural path, hunting down an exception handler. The problem arises when the code already executed has performed some operation that needs to be undone, but the code to undo it is skipped because of an exception. Some obvious scenarios where this may be a problem:

- opening a file on disk, processing its, then closing the file
- allocating some memory, using that memory, then freeing it
- changing the mouse cursor to an hourglass, performing a

time-consuming operation, then changing it back to an arrow

If an exception occurs in the in-between part (processing the file contents, using the memory or in the time-consuming operation, from the examples above) then the important last parts (closing the file, freeing the memory, or restoring the mouse cursor) will be skipped. There are many cases where this kind of problem can arise and so a general purpose solution is required for it. All languages that support exceptions offer a solution to the problem although the one offered by the ANSI standard C++ language is rather more involved than Object Pascal's.

So the resource protection issue arises where some form of resource (be it a file, cursor, memory block, dynamically created component) is set up, used by some code and then tidied up (closed or freed or whatever) in a sequential series of statements that will execute. This is shown in the following pseudocode:

```
resource allocation
```

```
code that uses the resource
```

```
resource deallocation
```

If an exception occurs in the code between the allocation and deallocation then the deallocation will not occur. The Object Pascal solution to this problem is another compound statement dedicated to the task in hand. The **try..finally** statement is used to guarantee the execution of important code:

```
resource allocation
```

```
try
```

```
code that uses the resource
```

```
finally
```

```
resource deallocation
```

```
end
```

The resource allocation goes first. If you fail to allocate the resource (for example there is not enough memory available to satisfy the memory request, or a file could not be opened) then there is no point trying to deallocate the resource.

Once execution hits the **try** you are guaranteed that the finally part will execute, regardless of whether any exceptions occur or not. This is why a **try..except** statement cannot validly overcome this problem; it is designed to execution special code *only* if an exception occurs whereas a **try..finally** statement executes some special code *all the time*.

Some programmers occasionally "fix" this problem by catching and killing any exception that occurs, as in the following code block. This is frowned upon as it is considered bad practice to kill all exceptions arbitrarily. You should handle exceptions you are anticipating or can foresee potentially occurring, which means the problem should be solved with the **try..finally** statement.

```
//Not a good way of dealing with resource protection issues
```

```
resource allocation
```

```
try
```

```
code that uses the resource
```

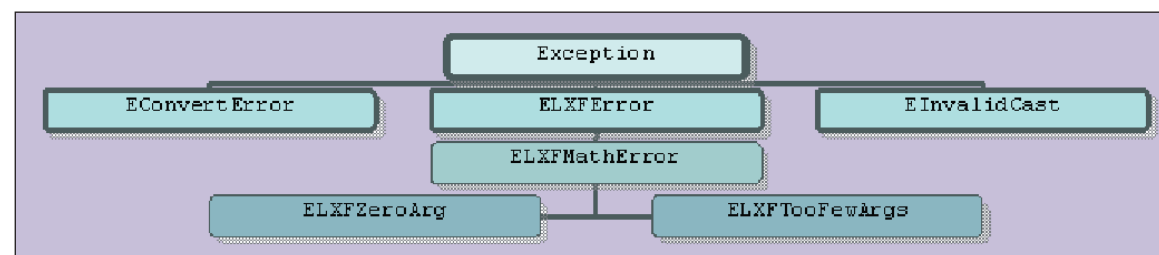
```
except
```

```
end;
```

```
resource deallocation
```

With this in mind, the code from last month that appended a line to a text file should now be written as follows. If **AssignFile**

Figure 7: A family of custom exceptions.





generates an exception the **try..finally** that closes the file will not be executed at all, but the error handling will pick the error up. If an exception occurs whilst trying to set the file to **append** mode (inside the try part of the **try..finally**) the **finally** part will ensure the file is closed before the exception makes it way to the **EInOutError** exception handler.

```
if dlgSave.Execute then
try
  AssignFile(TF, dlgSave.FileName);
try
  Append(TF);
  WriteLn(TF, 'Hello world');
finally
  CloseFile(TF)
end
except
  on E: EInOutError do
    case E.ErrorCode of
      ENOENT: ShowMessage('"%s" cannot be located'
, [dlgSave.FileName]);
      EACCES: ShowMessage('You do not have permission to
overwrite this file');
    else
      if E.ErrorCode < 100 then
        ShowMessage('I/O Error'#10#9'Error code: %d'#10#9 +
'Error msg: %s', [E.ErrorCode,
SysErrorMessage(E.ErrorCode)])
      else
        ShowMessage('I/O Error'#10#9'Error code: %d'#10#9 +
'Error msg: %s', [E.ErrorCode, E.Message]);
      end;
    end
end
```

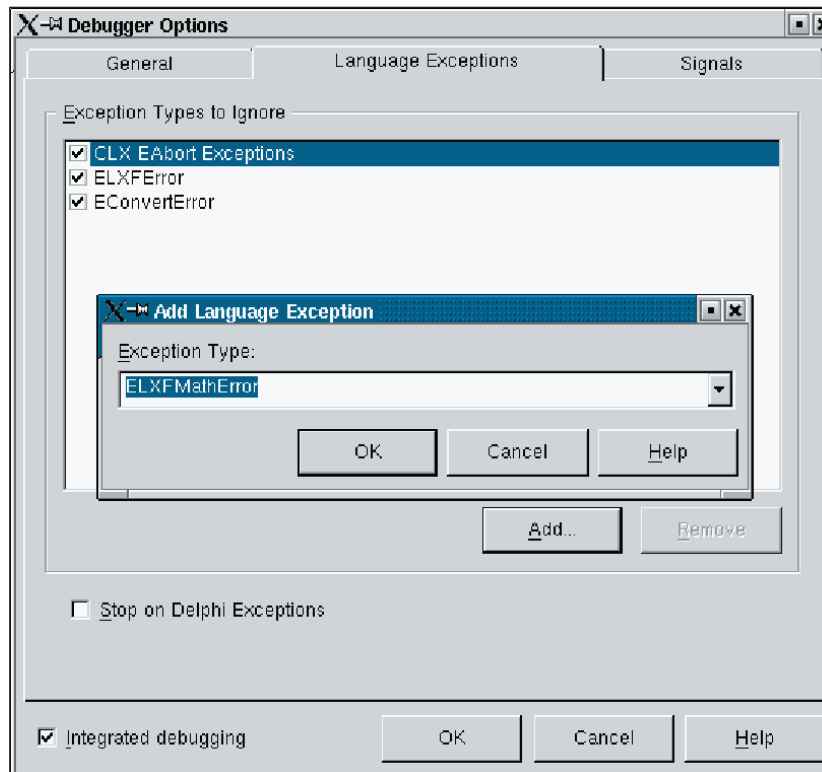
## Abbreviated Component Creation

If you need to use a component or object for a short period of time, such as in an event handler, you should use a **try..finally** statement to ensure it gets destroyed. This is commonly done with **TIniFile** objects (used to store application settings in simple text files). If you only have a couple of settings to make there is no need to keep the **TIniFile** object in existence – create it, use it and then destroy it. This can be wrapped up in a reusable routine:

```
procedure WriteIniSetting(const IniFile, Section, Ident, Value:
String); overload;
var
  Ini: TIniFile;
begin
  Ini := TIniFile.Create(IniFile);
  Ini.WriteString(Section, Ident, Value);
  Ini.UpdateFile;
  Ini.Free
end;
```

It is very common to abbreviate this type of code using a **with** statement, dispensing with the local variable, like this:

```
procedure WriteIniSetting(const IniFile, Section, Ident, Value:
String); overload;
begin
  with TIniFile.Create(IniFile) do
  try
    WriteString(Section, Ident, Value);
    UpdateFile
  finally
    Free
  end
end;
```



And with dynamically created forms to be used as dialogs:

```
frmDialog := TfrmDialog.Create(Application);
//Set checkbox to reflect window visibility
frmDialog.CheckBox1.Checked := frmWindow.Visible;
if frmDialog.ShowModal = mrOk then
  frmWindow.Visible := frmDialog.CheckBox1.Checked;
frmDialog.Free;
frmDialog := nil
```

This needs resource protection to ensure that the call to **Free** is made, and can also be abbreviated using a **with** statement:

```
with TfrmDialog.Create(Application) do
try
  //Set checkbox to reflect window visibility
  CheckBox1.Checked := frmWindow.Visible;
  if ShowModal = mrOk then
    frmWindow.Visible := CheckBox1.Checked
  finally
    frmDialog.Free
  end
```

## Miscellaneous

Last month we noted that the translation routine **StrToInt** generates an **EConvertError** exception if you pass it a string that does not represent a 32-bit integer. The same is also true of **StrToInt64** (if the string doesn't represent an **Int64**, a 64-bit integer). So **StrToIntDef('hello', 0)** returns **0**.

The exception can be avoided by using a related routine, **StrToIntDef** (or **StrToInt64Def**), which takes a default value to return in case the string is bad. *Kylx 2* introduces more of these exception-free translation routine equivalents:

Translation routine	Exception-free equivalent
StrToBool	StrToBoolDef
StrToFloat	StrToFloatDef
StrToCurr	StrToCurrDef
StrToDate	StrToDateDef
StrToTime	StrToTimeDef
StrToDateTime	StrToDateTimeDef

LXF

Figure 8: Telling the debugger to ignore some exceptions.

## About Brian Long

Brian Long is a UK-based freelance trainer and problem solver for Borland's *Kylx*, *Delphi* and *C++Builder* packages. His Web site is at [www.blong.com](http://www.blong.com) and he can be emailed at [brian@blong.com](mailto:brian@blong.com)

## NEXT MONTH

Next month we'll look at some simple graphics programming using the **CLX TCanvas** class. In the meantime, if there is something about *Kylx Open Edition* you want to see covered here, drop us an email and we'll try our best to incorporate it into a future instalment.

## SNEAKERWARE STORAGE

# FlashGO! and USB storage for Linux

**Chris Bidmead** opens his USB port to a host of portable devices and different flash storage media.

One of the things that puts people off using Linux is the supposed unavailability of drivers for the newer devices like cameras, dictating machines and MP3 players. I'm always inclined to turn the question the other way round, and ask if you really want to buy a peripheral that only works with a particular operating system. I'm glad to report that the world seems to be moving in the right direction now. Instead of eccentric peripherals that each require special drivers, which may or may not be available for your favorite operating system, manufacturers are beginning to build devices that play to established cross-platform standards.

One such standard, "the USB-Storage class", covers devices that connect by USB and present themselves to the host system as drives. The downside from the point of view of the Free Software community is that some details of the standard are only available under non-disclosure. However, a group of Linux developers has bitten the bullet and created a project (<http://www2.one-eyed-alien.net/~mdharm/linux-usb/>), maintained

by Matthew Dharm. Over the past year the fruits of this project have been appearing in standard Linux distributions.

USB Storage very broadly solves the problem of software incompatibility when attaching many USB devices, but there's a further hurdle. Manufacturers have muddied the water by creating a number of different physical standards around flash memory technology. For example, as I write this I have in front of me a Sony camera, an Olympus camera and a Panasonic MP3 player. All of them store their data on removable flash memory, but on three different and physically incompatible formats called respectively Memory Stick, Smart Media and Secure Digital. And there's also the Compact Flash standard used by Kodak cameras and many PDAs.

The solution is a very useful piece of hardware from Imation called FlashGO! ([www.imationltd.co.uk](http://www.imationltd.co.uk)). FlashGO! is an adapter system that manages all of these different flash formats as a single USB connectable device. So with a FlashGO! and Matthew Dharm's *usb-storage* driver the Linux user is theoretically able to cope with the whole range of flash formats.

Unfortunately I have to add the rider "theoretically" because it turns out that the manufacturers have further surprises up their sleeves. FlashGO! plus *usb-storage* enables any of these motley formats to be presented to your Linux system as a regular SCSI drive. An attached SCSI drive doesn't have to be formatted to Linux's native ext2 standard – Linux understands DOS, Mac, ISO-9660 and variety of other filesystems. But a manufacturer may choose to mess you around at this level, either with a totally different filesystem format, or a standard format like DOS, but tricked up with some special encryption. Alternatively you may get a perfectly manageable filesystem but be presented with individual files that use some off-the-wall proprietary format that open source software has no access to (See box *Lock out and Lock in*).

## Modules you need

Matthew Dharm's *usb-storage.o* module makes use of the standard Linux USB module base. This consists of *usbcore.o* and

```
[root@artist bidmead]# lsmod | grep usb
usb-uhci      20672      0 (unused)
usbcore      47248      1 [usb-uhci]
[root@artist bidmead]#

Nov 28 16:40:02 artist kernel: hub.c: USB new device connect on bus1/1, assigned device number 5
Nov 28 16:40:02 artist kernel: usb.c: USB device 5 (vend/prod 0x718/0xb000) is not claimed by any active driver.
Nov 28 16:40:03 artist kernel: Initializing USB Mass Storage driver...
Nov 28 16:40:03 artist kernel: usb.c: registered new driver usb-storage
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 3, frame# 490
Nov 28 16:40:03 artist kernel: scsi: SCSI emulation for USB Mass Storage devices
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 2, frame# 546
Nov 28 16:40:03 artist kernel: Vendor: Model: Rev:
Nov 28 16:40:03 artist kernel: Type: Direct-Access ANSI SCSI revision: 02
Nov 28 16:40:03 artist kernel: attached scsi removable disk sda at scsi, channel 0, id 0, lun 0
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 3, frame# 614
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 3, frame# 628
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 3, frame# 642
Nov 28 16:40:03 artist kernel: sda: READ COMMAND failed.
Nov 28 16:40:03 artist kernel: sda: status = 1, message = 00, host = 0, driver = 00
Nov 28 16:40:03 artist kernel: sda: extended sense code = 2
Nov 28 16:40:03 artist kernel: sda: block size assumed to be 512 bytes, disk size 1GB.
Nov 28 16:40:03 artist kernel: usb-uhci.c: interrupt, status 3, frame# 657
Nov 28 16:40:03 artist kernel: sda: test WPI failed, assume Write Protected
Nov 28 16:40:03 artist kernel: sda: I/O error: dev 08:00, sector 0
Nov 28 16:40:03 artist kernel: unable to read partition table
Nov 28 16:40:03 artist kernel: USB Mass Storage support registered.
```

Blue window: the USB modules already running. White window: the FlashGo! device being attached as */dev/sda*.

## Lock out and lock in

### Proprietary evils explained

Why would any sane manufacturer want to create proprietary filesystems or data types that defeat the purpose of USB Storage? There are two answers: lock-in and lock-out. Lock-in means that any data you create gets stored in the manufacturer's own proprietary format, so you need to stick with that manufacturer to maintain compatibility.

An example of this is the proprietary format Sony uses to store the sound files in its range of dictating machines. FlashGO! and *usb-storage* can access these files quite happily, but without Sony's own software (written only for Windows) you can't play or edit them.

Lock-out means that a manufacturer is happy to sell you data, like a music file,

but doesn't trust you with it. The data may still be in a universally manageable format like MP3, but the filesystem it resides on will have some tricky feature that only gives you conditional access. Sony has a version of Memory Stick, fancifully called *Magic Bridge*, that fits this description. And as its name implies, Secure Digital is another

format that employs lock-out.

Lock-in and lock-out are essentially weapons in what manufacturers see as a war against their customers. And not just Linux users – they put users of every kind at a disadvantage. But with plenty of devices available that don't use lock-in or lock-out, happily it's not a war you have to fight.



## Approx memory card prices

A rough guide compiled from various sources

128MB except where otherwise stated

CompactFlash	£50
MicroDrive	£169 (340MB)
SmartMedia	£50
MultiMedia	£40 (64MB)
Memory Stick	£120

either `usb-uhci.o` or `usb-ohci.o` depending on whether your hardware uses the "Universal" or the "Open" Host Controller Interface. You'll also require `ide-scsi` and the set of associated `scsi` modules.

The good news is that if you're using any of the current mainstream Linux distributions, all this will be set up for you already. When you plug in your USB device the required modules should load automatically. The FlashGO! gives you visual feedback about the connection by lighting up its red LED, and you can check that the modules are loaded by running

**lsmod | grep usb** (if you're not root you may need the full path to `lsmod`).

Depending on your Linux distribution you may also find that the new drive device is also mounted automatically. If it isn't you'll need to mount it manually, and to do that you'll need to know what SCSI device name it has acquired, and how it is partitioned. The following two screen shots walk you through this part of the process.

You won't necessarily need FlashGO! if your device uses the USB-Storage standard when connected directly. For example, although FlashGO! works perfectly as a means of connecting the Memory Stick module from my Sony SDC-S75 digital camera, simply plugging in the camera directly through its built in USB port makes it possible to mount the Memory Stick in exactly the same way. But for devices that use any kind of flash card but don't have their own USB ports, FlashGO! makes an ideal intermediary. I've also used it successfully to attach a 1GB IBM Microdrive.

## Late news flash

Since I wrote the foregoing I've been sent an Archos MP3 Recorder. Like the other gizmos it's advertised only as being useable with Windows and Mac, and you have to scour the Archos Web site (at <http://www.archos.com>) pretty closely to discover a link to some useful work done by Linux developer Bjorn Stenberg (<http://bjorn.haxx.se/isd200>). The Archos uses an ASIC chip called an ISD-200 to translate the hard drive logic across to the USB, and Bjorn's page discusses how a Linux loadable module has been developed which understands the ISD-200.

The bottom line is that current Linux kernels (2.4.10 or more recent) with their associated modules come fully equipped to attach the Archos (and other devices using the ISD-200) as a hard drive – automatically if your distribution is properly set up. Earlier kernels, back to 2.4.5, can be tweaked (see Bjorn's web page).

I've just upgraded to the new Mandrake 8.2 beta, which comes with the 2.4.17 kernel. When I connected the Archos to the USB port an apparent bug in the beta resulted in a complaint that the `usb-storage` module wasn't available, but loading it manually fixed the problem. As I write this I'm listening to an MP3 track on the Archos playing directly into *Xmms* on my Linux box. I'm very impressed... :-)

[LXF](#)



The FlashGO! USB connector and its three associated adapters. Also included is a Kodak CompactFlash card, a Sony MemoryStick (top right) and the IBM Microdrive.

## Some other caveats

From the murky world of obfuscated standards

If you're tempted to see USB-Storage as the Holy Grail of cross-platform computing, it's worth taking a reality check by reading the source code. You don't need to understand C to get the drift of the following comment in `uhci.c`, currently maintained by Johannes Erdfelt:

- \* WARNING! The USB documentation is downright evil. Most of it
- \* is just crap, written by a committee. You're better off ignoring
- \* most of it, the important stuff is:
- \* - the low-level protocol (fairly simple but lots of small details)
- \* - working around the horridness of the rest

Much of the information about USB-Storage (including the full list of manufacturers involved

in it) is under NDA. I asked Matthew Dharm whether this presented a problem for the open source developer. He told me that the NDA status of material isn't actually the biggest problem, because, after all, the specs are publicly available. However, says Matthew:

"The mass storage specification is (like many specifications) slightly ambiguous. Clarification on some of the grey areas is generally to be had by looking at various implementations. The problem, of course, is that people often implement these things differently – the only really good source is the people who wrote the specification, and to talk to them (and their mailing list), you need to be a USB-IF member, which is way too expensive for an individual."

```

bidmead@artist.cbidmead.home.edu /misc/ldal/home/bidmead
[ root@artist bidmead ]# lsmod | grep usb
usb-storage      25952    0 (unused)
usb-uhci         20672    0 (unused)
usbcore          47248    1 [usb-storage usb-uhci]
scsi_mod         86036    3 [usb-storage ide-scsi sd_mod]
[ root@artist bidmead ]#

bidmead@artist.cbidmead.home.edu /misc/ldal/home/bidmead
[ root@artist bidmead ]# fdisk /dev/sda -l

Disk /dev/sda: 64 heads, 63 sectors, 522 cylinders
Units = cylinders of 4032 * 512 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           1           521     1050304    6   FAT16
[ root@artist bidmead ]#

```

Blue window: the `usb-storage` module has automatically loaded.

White window: running `fdisk` on `/dev/sda` shows it to have a single partition, `/dev/sda1`.

## WEB PROGRAM INTERFACE

# An introduction to CGI programming

**Dave Cross** starts a new series of tutorials on creating dynamic web content with CGI programs.

**C**GI is the “Common Gateway Interface” a very grand sounding name for something that is really very simple. It’s nothing more than a protocol that defines how a web server can interact with an executable program to produce dynamic web pages. It defines how the web server sends information to the program and how the program can send information back to the web server, which the web server will, in turn, send back to the browser.

We’ll look at this in more detail soon, but first let’s ensure that we’ve got everything set up so that we can run CGI programs.

## Start with Perl

Firstly, you’ll need a web server installed. Just about every Linux distribution comes with a version of *Apache*, which is by far the most common web server on the Internet. In this series of tutorials we’ll assume that you’re using *Apache*, but the examples should work just as well on any other web server. See the box *Getting, Installing and Configuring Apache* for more details on how to set *Apache* up on your system.

You’ll also need a programming language to write your programs in. You can use just about any language to write CGI programs, but the majority of people seem to use Perl. The main reason for this is Perl’s unequalled ability to manipulate text. Other advantages that Perl has include the fact that it runs on more platforms than just about any other programming language, and that it has a great repository of free modules and language extensions at [www.cpan.org/](http://www.cpan.org/). Perl also comes as a standard part of every Linux distribution, so you’ll already have it installed.

To keep this tutorial down to a manageable size, I’m going to have to assume that you already know something about programming in Perl. If not, see the box *Learning Perl*.

## Our first CGI program

Let’s write a simple CGI program in Perl. We’ll stick with tradition and write a “hello world” program. Here it is in its entirety.

```
#!/usr/bin/perl -wT
use strict;

print "Content-type: text/plain\n\n";
print 'Hello world';
```

Save this into a file called “hello” in your cgi-bin directory and run it by entering the URL

```
http://localhost/cgi-bin/hello
```

in your browser. You should see the the text “Hello world” in the browser. If not, read the box on debugging CGI programs to get help tracking down the problem.

It’s not a very complex program at all. There are only a couple of things that distinguish it from a standard Perl program that you’d run from the command line. The first of these is the **-T** option on the shebang line. This puts Perl into “taint mode”. In this mode Perl will automatically distrust any data that comes from the user and won’t let you pass that data to the operating system until you have untainted it. We’ll see why taint mode is a good idea, and how to untaint data, in next month’s article. In the meantime it’s just a good habit to get into.

The second difference is the addition of the line that prints the content-type header. This is our first brush with the CGI protocol. All CGI programs should output a set of headers which the web server will pass back to the user’s browser. In most cases the only header that we will need to return is the content type header. This tells the browser what kind of data the program is sending. In our example we returned plain text, so the content type was **text/plain**. More usually we’ll return HTML, so the content type will be **text/html**. There’s no reason why your program couldn’t return more complex data, for example, PNG (**image/png**). The set of headers should be separated from the actual data with a blank line. This is achieved by printing two newline characters after the final (or, in this case, only) header.

Also notice that the program writes its output to STDOUT. The web server will have arranged to capture any data written to STDOUT and to pass it back to the browser.

## A (slightly) more complex example

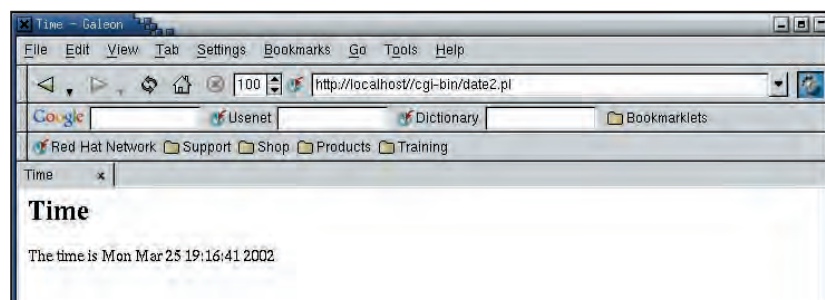
Our previous example really didn’t do very much at all. There’s no need to have a CGI program which just displays fixed text. – an HTML page will do that. Let’s write something which has dynamic output. Here’s a script that prints the current time.

```
#!/usr/bin/perl -wT
use strict;

print "Content-type: text/plain\n\n";
my $now = localtime;
print "The time is $now";
```

This script works on exactly the same principle as the last one, but we get the system time using the Perl function

The output from the HTML date program.





## Getting, Installing and Configuring Apache

### A simple HOWTO

If you don't already have *Apache* installed you can get the source code from <http://httpd.apache.org/>. Different Linux distributions put the various parts of *Apache* in different directories, but if you build your own copy from the source code you'll find that the default installation puts everything you need under `/usr/local/apache`. This directory will contain a number of subdirectories. For our purposes, we'll only need to look at five of them:

**bin** – this is where the *Apache* program (called *httpd*) is found together with other various utility programs.

**cgi-bin** – this is where you'll put your CGI programs. It will already contain a few sample CGI programs.

**conf** – this contains the configuration files for *Apache*.

**htdocs** – this is the root for your web site. All of your HTML files will go underneath this directory.

**logs** – this is where the log files are found. It's much easier to debug CGI programs if you have access to the error log.

### Starting Apache

If *Apache* was installed as part of your standard Linux installation then it will

almost certainly also be configured to start automatically when Linux starts up. You can see if it's running by typing

```
ps -e | grep httpd
```

at your command line. If *Apache* is running you'll see a number of *httpd* processes running. If it isn't running you'll need to start it yourself. There's a program called *apachectl* in the `bin` directory. Simply run

```
apachectl start
```

and you should see a message saying that the server has started. You can also run

```
apachectl stop
```

to stop the server and

```
apachectl restart
```

to stop and immediately restart the server. This last option is useful when you've changed the server configuration.

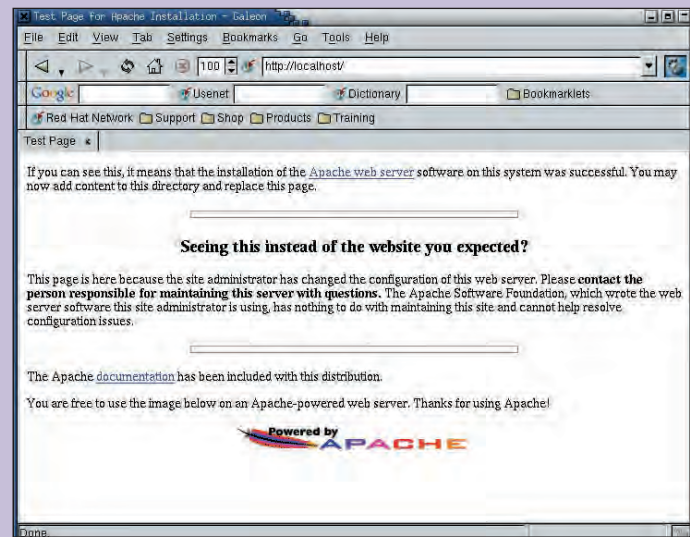
Having started your *Apache* server we can check that it's working properly.

Simply start up your favourite browser and type

```
http://localhost
```

in the location box. You should see the *Apache* web server default page. If not, you'll need to investigate a bit further to find out what the problem is.

Assuming that *Apache* is working correctly, the next step is to check that it is configured to run CGI programs



This is what you'll see if *Apache* is running correctly.

correctly. We'll try to run one of the *Apache* sample CGI scripts. The URL is `http://localhost/cgi-bin/printenv`

If your server is configured correctly you'll see a list of environment variables displayed in your browser. If not, it's time to start looking at the *Apache* configuration file. The file is called `http.conf` and it's in the `conf` directory under the main *Apache* directory. Search

this file for a line like

```
ScriptAlias /cgi-bin/
"/usr/local/apache/cgi-bin/"
```

This defines which directory will be your `cgi-bin`. Ensure that the directory path is correct and that the line isn't commented out. Having made the relevant changes, you'll need to restart *Apache* as described above.

`localtime` and print that value. Each time you reload the script the time will be updated. Once again this script is outputting plain text. Let's see a version that returns HTML.

```
#!/usr/bin/perl -wT
use strict;

print "Content-type: text/html\n\n";
my $now = localtime;

print <<END_HTML;
<html>
<head>
  <title>Time</title>
</head>
<body>
  <h1>Time</h1>
  <p>The time is $now</p>
</body>
</html>
END_HTML
```

This program looks a little more complex, but really it's doing exactly the same thing as the previous example. The added complexity comes purely from the fact that we have to display a complete HTML page. Also note that we have changed the content type to reflect the fact that we are now returning HTML. The browser will recognise the content type and will convert the HTML into a web page. On the left you can see the output from this script.

Some people think that it looks untidy having all of that HTML

in your Perl program. In the standard Perl library there's a module called **CGI.pm** which allows you to "hide" the HTML generation behind function calls. Here is the same script using **CGI.pm**.

```
#!/usr/bin/perl -wT
use strict;
use CGI 'standard';
print header;
my $now = localtime;

print start_html(-title=>'Time'),
  h1('Time'),
  p("The time is $now"),
  end_html;
```

We've made a few changes to the script. The most important one is to add the **use CGI** line. This loads the **CGI.pm** module and asks Perl to import a standard set of functions from that module into our program. We then use five of these functions to display the page. The **header** function produces the content type header. The default content type that it returns is **text/html**, but you can pass it a parameter to return other types. **start\_html** returns the standard block of HTML that always begins an HTML page. Notice that it takes an optional argument which defines the contents of the **<title>** tag. **h1** and **p** create headers and paragraphs and **end\_html** prints the final **</html>** tag.

It doesn't really matter whether or not you use these HTML functions in your CGI programs. If you're more comfortable putting the raw HTML into your code then feel free to do so. You should still use **CGI.pm** however, as it has other features which we'll see very soon.



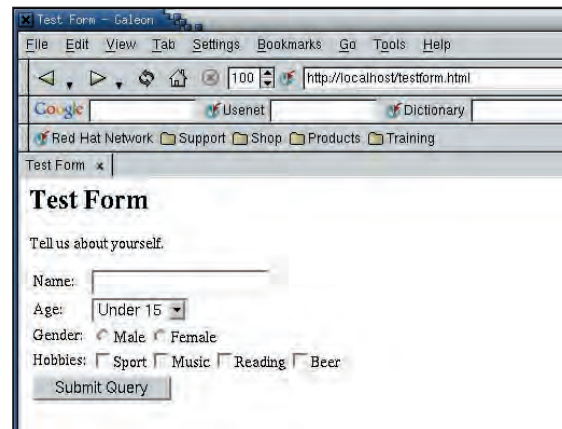
## ◀ Getting input - HTML forms

The previous section told you just about all that you need to know about displaying output from CGI programs. The other side of the equation is getting input into the program. To talk about this, we need to take a slight detour and look at HTML forms.

HTML defines a number of input widgets which you can use to pass user data to a CGI program. This is much the same set of widgets that you'll be familiar with from any GUI application – text fields, radio buttons, checkboxes, pop-down menus and many others. Here is a very simple form that makes use of a number of these widgets.

```
<html>
<head>
  <title>Test Form</title>
</head>

<body>
  <h1>Test Form</h1>
  <p>Tell us about yourself.</p>
  <form action="/cgi-bin/form.pl" method="get">
    <table>
      <tr>
        <td>Name:</td>
        <td><input type="text" name="name"></td>
      </tr>
      <tr>
        <td>Age:</td>
        <td><select name="age" size="1">
          <option>Under 15</option>
          <option>15 - 25</option>
          <option>26 - 35</option>
          <option>36 - 45</option>
          <option>Over 45</option>
        </select></td>
      </tr>
      <tr>
        <td>Gender:</td>
        <td><input type="radio" name="gender"
value="Male">Male
          <input type="radio" name="gender"
value="Female">Female</td>
      </tr>
      <tr>
        <td>Hobbies:</td>
        <td><input type="checkbox" name="hobby"
value="Sport">Sport
          <input type="checkbox" name="hobby"
value="Music">Music
          <input type="checkbox" name="hobby"
value="Reading">Reading
          <input type="checkbox" name="hobby"
value="Beer">Beer</td>
      </tr>
    </table>
    <td colspan="2"><input type="submit"></td>
  </tr>
</table>
</form>
</body>
</html>
```



A simple web form.

```
<input type="checkbox" name="hobby"
value="Reading">Reading
  <input type="checkbox" name="hobby"
value="Beer">Beer</td>
</tr>
<tr>
  <td colspan="2"><input type="submit"></td>
</tr>
</table>
</form>
</body>
</html>
```

This form contains a text input (for the name), a drop-down list (for the age), a pair of radio buttons (for the gender) and a set of checkboxes (for the list of hobbies. Most importantly it also has a button to submit the data to the web server, see pic, above.

Notice that each of the form input elements in the HTML page has a name attribute. This will be important when we come to access the data. Also notice that the input elements are within a **<form>** tag and that one of the attributes of the **<form>** tag is called **action**. This attribute defines the CGI program that will be called when the form is submitted.

Here's the script that processes the form data

```
#!/usr/bin/perl -Tw

use strict;
use CGI 'standard';

my $name = param('name');
my $age = param('age');
my $gender = param('gender');
```

## Learning Perl

### A bestiary of books

Here are some suggestions of some places to go to get information on Perl.

The best book for learning Perl is called *Learning Perl*, and is by Randal Schwartz and Tom Phoenix. It's also known as the Llama book because it has a llama on the cover. The Llama is good if you already have experience of programming in another language. If you've never programmed before then you'll be better off reading Andrew Johnson's *Elements of Programming*

with Perl or Simon Cozen's *Beginning Perl*.

The definitive Perl reference book is *Programming Perl*, by Larry Wall, Tom Christiansen and Jon Orwant. This has a camel on the cover so it's also known as the Camel book. Other Perl books worth reading include *The Perl Cookbook* by Tom Christiansen and Nathan Torkington and *Effective Perl Programming*, by Joseph Hall with Randal Schwartz.

For more details about CGI

programming with Perl I recommend *CGI Programming with Perl*, by Scott Guelich, Shishir Gundavaram and Gunther Birznieks, *Writing CGI Applications with Perl*, by Kevin Meltzer and Brent Michalski and *The Official Guide to Programming with CGI.pm*, by Lincoln Stein.

There are a number of good web sites where you can get information about Perl. The main Perl site is at <http://www.perl.com/>. There's a site

for Perl beginners at <http://learn.perl.org/> and the best place to get help and advice on Perl problems is a site called Perl Monks at <http://www.perlmonks.org/>.

Perl users' groups are called "Perl Mongers". To see if there is a group near you go to <http://www.pm.org/> – or have a look at the map: <http://bath.pm.org/map/>.

And, of course, there's a Perl tutorial column every month in *Linux Format*, which is written by Charlie Stross.



```
my @hobbies = param('hobby');
my $list;
```

```
if (@hobbies) {
    $list = join ' ', @hobbies;
} else {
    $list = 'None';
}
```

```
print header;
start_html(-title=>$name),
h1("Welcome $name"),
p("Here are your details:");
table(Tr(td('Name:'),
        td($name)),
      Tr(td('Age:'),
        td($age)),
      Tr(td('Gender:'),
        td($gender)),
      Tr(td('Hobbies:'),
        td($list))),
end_html;
```

Once again we've used **CGI.pm** to produce the HTML, but we've also used the same module to get our input. **CGI.pm** has a function called **param** which returns information about the data that has been passed to the script. Called without any parameters, **param** will return a list of all of the parameter names and when given the name of a parameter it will return the value associated with that name.

The CGI parameter names are the names of the input elements on the HTML form. For example the **name** parameter returns the text that was typed into the text box called **name** and the **age** parameter returns whichever of the options in the **age** drop down menu was selected.

When you submit the form you'll get back a page like the one shown on the right. If you look in the location box of the browser you'll see that the URL has changed. It has the names and values of the CGI parameters appended to the program's URL, thus:

```
http://hostname/path/to/program?name1=value1&name2=
value2&name3=value3
```

You can also see that any spaces in the names and values have been converted to **+** signs. Other punctuation marks would be converted to a **%** sign followed by the ASCII code for the character in hexadecimal. For example an **@** character would be represented as **%25** (25 hex is 37 decimal, which is the ASCII code for **@**). This is known as URL-encoding. The piece of the URL containing the parameters (the part following the **?** character) is called the query string.

Before **CGI.pm** was included in the standard Perl library (over five years ago), every CGI program had to do all the work to extract the CGI parameters from the query string. They would get the query string, split it into name/value pairs and undo all of the URL-encoding. It was a lot of code that was included in every CGI program. You'll still see code that does it in CGI scripts of dubious quality today. These days **CGI.pm** handles all of that behind the scenes so we no longer have to worry about it.

In our script, we get the values of each of the parameters in turn and assign them to variables. We then print out these values in an HTML page. This script also demonstrates how to use some more of the HTML functions from **CGI.pm** – in this case the **table**, **Tr** and **td** functions that are used to create HTML tables. All HTML tags have a corresponding function in **CGI.pm**. Notice the function for creating a **<tr>** tag is called **Tr** rather than **tr**. This is because Perl already has a built-in operator called **tr**.

## Debugging CGI programs

### Our fault-finding tips

Obviously not every CGI program that you write is going to work first time. But because the web server is involved, people often find it a bit tricky to debug CGI programs. Here are a few tips.

#### 1. Check syntax first

A number of errors can be caught by checking the syntax of the program from the command line. If you type

```
perl -cT script_name.pl
```

at your command line, Perl will check the syntax of **script\_name.pl** and report on any errors it finds.

#### 2. Check logic

Another good reason for using **CGI.pm** is that it knows when your script is being run from the command line and it fakes a CGI environment for your script to run in. For example, running **script\_name.pl "name1=val1&name2=val2"** will run **script\_name.pl** putting **name1=val1&name2=val2** into the query string. This allows you to check the behaviour of your script under a number of different inputs. You can even run a CGI script through Perl's built-in debugger by running

```
perl -Td script_name.pl
"name1=val1&name2=val2"
```

#### 3. Check error logs

I mentioned earlier that *Apache's* log files are

stored in the logs directory. When you're debugging a CGI program the most important log file is called **"error\_log"**. When there is an error in your program, Apache will display an uninformative **"500 - Server Error"** message to the user, but will write the real error message into the error log. This is where you'll find more detail on exactly what went wrong.

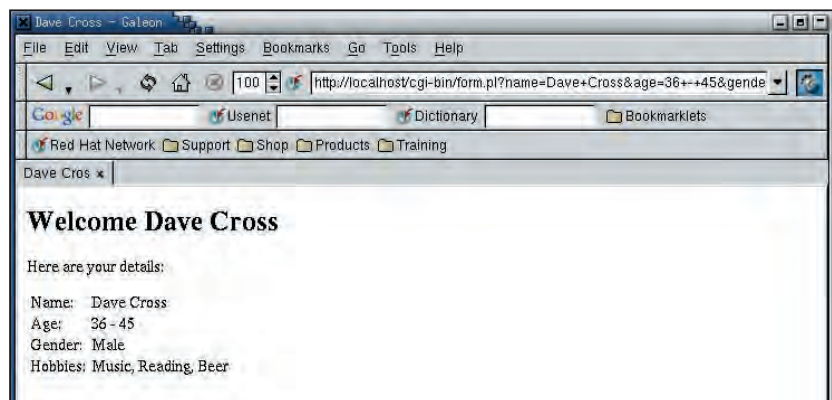
#### 4. Use CGI::Carp

Apache displays the nondescript **"500 - Server Error"** page for a reason. Telling just anyone what errors your programs are generating is a security risk. Only the webmaster needs to know the details of the errors and he can always read the error logs. However, when you're developing a program it's handy to get full error messages in the browser. The Perl standard library contains a module called **CGI::Carp** which can do just that. Simply add the line

```
use CGI::Carp 'fatalToBrowser';
```

to your program and any fatal errors will appear in your browser as well as in the error log. This makes it easier to debug the program. When you've finished development, change the line to **use CGI::Carp;**

and the errors will no longer go to the browser, but the messages in the error log will be nicely formatted and timestamped.



One parameter is handled slightly differently from the others. In our form we had a series of checkboxes all of which had the name **hobby**. If you look at the query string, you'll see how this is handled. The query string contains a sequence like this

```
hobby=Music&hobby=Reading&hobby=Beer
```

There's a **hobby=** section for each checkbox that was selected. If you look at the code for this program you'll see how we deal with this. Instead of assigning this parameter to a scalar, we assign it to an array. If you have a multi-valued parameter like this, the **param** function will return a list of values. We can then examine the array to see how many hobbies were chosen and substitute a suitable value if none were selected.

## Conclusion

So in very simple terms, that's about all there is to know about writing CGI programs. You get your input using **param**, do some processing, and then send some output (starting with a content type header) to STDOUT. **LXF**

The result of submitting our form.

## NEXT MONTH

Next month we'll write some more complex examples and look at some of the security issues that you need to consider when letting people run CGI programs on your web server.

# 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!

## Experts this month

Whatever your question is, we can find an expert to answer it – from installation and modern woes to network administrations, we can find the answer for you – just fire off a letter or email 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.



**Richard**

**Drummond** is an experienced programmer who can answer queries on a variety of subjects. A keen Debian user, he's also our resident Java guru.



**Nick Veitch** is

the editor of the magazine, and answers your easy questions! Or indeed anything to do with *Grub*, *LILO*, *netatalk*, vi...



## Win4Lin

**Q** I have recently upgraded to Mandrake 8.1 and I am running this on an AMD K6 500 processor with 98MB of RAM.

So far no problem, I took the expert install and only had to tell it not to run the ALSA driver at boot due to a sound card issue.

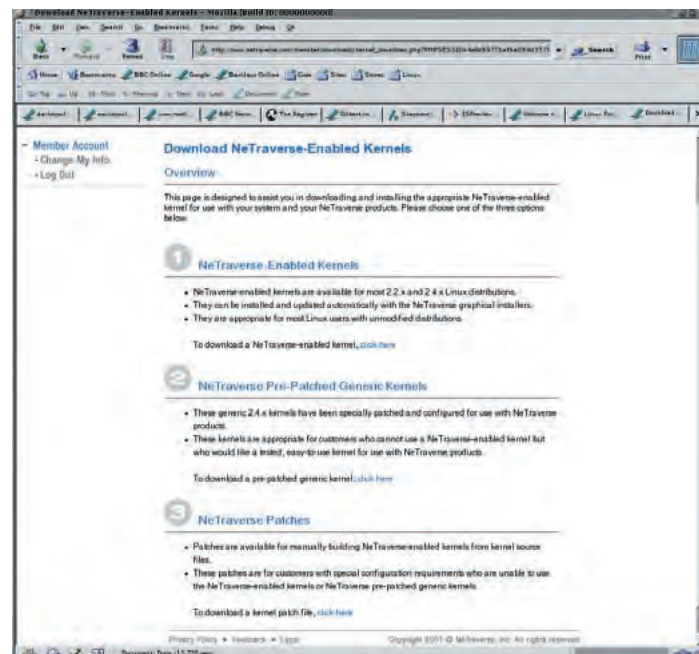
I found that this distro comes with the trial version of *Win4Lin* as an RPM package, with a supported kernel patch. When I try to install this using the graphical install all goes well until I have to boot with the patched kernel. It is already set up in *LILO* but just refuses to boot correctly, in fact it does not even get to the login stage.

Any advice would be extremely useful as the Mandrake site does not appear to have any hints.

**Fergus Thomson**

**A** It's difficult to figure out quite what is wrong with the kernel, as you've not supplied any debugging information. The patched kernel may lack certain capabilities which you need to get your machine up and running, or it might oops during booting. *Win4Lin* is produced by a company called NeTraverse, so you can download one of their kernels from [www.netraverse.com](http://www.netraverse.com), either precompiled, or you can download a patch for an existing kernel. If you patch a kernel source tree, with either the patch from Mandrake or one from [netraverse.com](http://netraverse.com), then you will need to correctly configure it for your system, including all of the device drivers and other capabilities which are required to get your box working correctly.

If you have little understanding of kernel configuration, a pre-compiled kernel may be the best way to go, as it will offer most basic capabilities which you will require, although if you have very specific hardware



NeTraverse have pre-compiled kernels available on their website for use with their various products, as well as patches.

requirements then you may be a little stuck when you come to trying to use certain kit with these kernels.

## Missing drives

**Q** Yes the same old problem (can hear the gnashing of teeth from here).

I've just installed Red Hat Linux 7.2. All went well until I logged in. I cannot get the DVD or CD-RW to mount. I continually get

```
mount: /dev/cdrom + 1 : unknown device
```

I've checked the `/dev` directory and both exist. Both are defined in `/etc/fstab`. I've checked with Red Hat's hardware browser and both `cdrom` drives are listed as `scd0` & `1` respectively. I've edited the `fstab` file and still the error appears, but with the new device settings. I've also rebooted in the vain hope of the setting working – but no luck.

I've used other distributions (Mandrake and Redmond Linux) and have not come across this problem. Any suggestions?

**Ian Farrelly**

**A** If it's mounting `/dev/cdrom`, you'll need to check to see which drive it is symlinked to using `ls -l`. Ideally, you'd set up `'/dev/cdrom0'` and `'/dev/cdrom1'`; and create two separate mount points in `/mnt` for them both, editing `/dev/fstab` appropriately. Try mounting `/dev/scd0` directly with:

```
mount -t iso9660 /dev/scd0 /mnt/cdrom
```

and seeing what happens. If that does not work, then you're probably not using the right devices, or the appropriate modules are not loading automatically.

As you have a CD-RW, you're probably using the `ide-scsi` module, to emulate a SCSI CD-Rom drive, so you can check the system logs with



**dmesg** right after boot to see what exactly is being loaded and which devices it is detecting. Once you've figured out what exactly it is doing to your devices, you can then test it out from the command line, then edit `/etc/fstab` to suit the specifics of the configuration.

## Email attachments

**Q** I have a PIII notebook, with over 300MB memory, running Red Hat 7.2. I usually use *Kmail* 1.3.2 on KDE 2.2.2-1 but have been trying *Mutt* 1.3 out from your DVD. I am also using *sendmail*-8.11.6-3 and have configured *Kmail* to use *sendmail*. I can send and receive emails even with small attachments, but when I try to send larger attachments – around 40KB – using the command `./sendmail -v -q` – or *Kmail*'s SMTP server – the connection times out.

I will try to send you an attachment which shows you that the connection to my ISP was started and one small message sent, one was locked whatever that means and one had a 'Time out' error. Whenever the latter happens



Playing DVDs is possible under *Xine*, as long as you have a plug-in which supports the DeCSS decryption library

I have to close my connection and reopen it again, which is a hassle.

This has happened with different desktops, different distros and also happens when I

configured *Kmail* to use the SMTP server and port settings instead of *sendmail*. Maybe it is a problem with 'wvdial.conf' or other ppp configuration files, limiting the size

of message I can send? I have spent a lot of time searching for answers and using different configurations of *sendmail* but unfortunately without joy. I would appreciate any help.

The other problem I have encountered is that with a lot of browsers – i.e. *Konqueror*, *Netscape*, *Mozilla*, and *Opera* – that I have used I cannot access secure web sites such as Ebuyer.com or 365online.com. However, this does not seem to be a problem with *Galleon* 1.0, which I would recommend to any one, which I got from your magazine.

I informed the Bank of Ireland's web site about this problem and they told me that this was the first time anyone complained, and that most of their customers use Windows. Anyway I think that the problem lies in the software and not with the sites – or does it?

Jim

**A** The email problem you are experiencing is probably an issue with your ISP's mail server, or your connection to it, as *sendmail* should not have problems sending emails of any size, unless you have specifically configured it not to deliver messages over a pre-defined size. Some ISPs automatically route all outgoing emails via their server, so you don't have any choice but to have emails go through their server. If

## A QUICK REFERENCE TO: diff and patch

The **diff** command compares the contents of two files (or two directories of files) and generates as output a list of differences between the two. This list can be saved as a patch or diff file which can then be used to create the second file (or directory) from the first. The task of applying such a diff is performed with the **patch** command. Diffs are typically used to save bandwidth when distributing updates to source code. It is much quicker to send only the changes between two versions of a source tree than to send the whole thing again. Here's an example:

```
diff -u example.v1 example.v2
>example.diff
```

This compares the file **example.v2** to the file **example.v1** and stores the list of differences in the file **example.diff** (all files are stored in the current directory). The switch **-u** specifies that the list of

changes should be output in the human-readable 'unified' format – which is generally preferable.

If somebody has a copy of the file **example.v1** stored under the filename **example**, say, then they can use the above diff to update that file to the contents of **example.v2** with:

```
patch example example.diff
```

When comparing directories, **diff** compares the contents of each file that appears in both directories and outputs the difference. By default, it does not recurse into any sub-directories, but you can tell it to do this with the **-r** switch. Also by default it ignores any files which are not present in both trees. You can tell it treat any such missing files as empty files with the **-N** switch and so include them in the diff. For example:

```
diff -urN source source.new
>source.diff
```

You can apply this diff to a tree named **source** – from the directory

which contains that tree – with the following:

```
patch <source.diff
```

However, if you wish to apply it to a differently named tree, you can **cd** into that tree and apply as follows:

```
cd source-1.5.2
patch -p1 <source.diff
```

The **-pN** (where **N** is a number) tells **patch** to strip off the **N** leading directory components of all filepaths specified in the diff file. By default, the filepaths in a diff are relative to the current directory when the diff was generated. To successfully apply a diff, those filepaths must exist. For example, paths in **example.diff** will all begin with 'source/' – so this diff must be applied from the directory containing 'source/'. The **-p** switch will treat the file-paths as being relative to the directory 'source' instead and so can be applied from within that directory.

# FREQUENTLY ASKED QUESTIONS

## »» How are devices mounted in Linux?

As far as Linux is concerned, a device is just a file, hopefully containing a filesystem (Like ext2 for Linux, FAT for DOS/Windows, etc). Devices are found by the Kernel or kernel modules, and linked to files in the /dev folder. These files are really pointers to the kernel interface to the raw device.

The **mount** command is used to make these files readable to the Linux system. By specifying a 'mount-point' (i.e. where in the filesystem the device is to be mounted) and a filesystem type (i.e. the 'format' of the device, which can be different types for Linux, DOS, Mac, Amiga, etc), the device can be 'mounted' onto the existing filesystem.

The /dev/ filenames for the devices usually follow easily understandable systems. Your first IDE hard disk is /dev/hda, the second is /dev/hdb and so on. Individual partitions can be accessed by using the partition number: hda2, hdb6, etc.

## »» How do I mount a disk manually?

Use the mount command! This is simple to invoke for most cases: `mount -t ext2 /dev/hdb1 /mnt/disk2` The syntax for simple use like this is:

```
mount -t filesystem device
      mount-point
```

As with most Linux commands, you can find out more by viewing the man pages.

## »» How do I tell my Linux box to mount a device every boot?

Put it in the /etc/fstab file. This is basically a list of devices, followed by the mount point, filesystem and other options. Typing **man fstab** will give you more info on the options available here.

## »» How can I use non-Linux filesystems?

Your Linux kernel will have to be compiled to include the specific filesystem support, or have a module compiled for it. Most distros include modules for common filesystems, including alternative

Linux systems. HFS for Mac, VFAT for Windows and ext3, ReiserFS for Linux are all commonly available.

If you don't have support for the filesystem you want to use, you will have to recompile the kernel for your system.

## »» How do I mount an SMB shared folder?

This uses the *smbfs* module, and works in pretty much the same way as the standard mount command. The difference is that there is no 'device' on your system for the SMB server. It works like this from the shell:

```
mount -t smbfs //SMBSERVER/Share
      /mnt/Shared -o
      password=xxxx,username=xxxxx
```

Though you can also use GUI tools like *gnomba* to browse the available servers and set up mount points for you.

## »» My Zip drive won't mount?

Depending on the type of drive and how the disk was originally formatted, you may find that the disk actually uses a number of partitions, even if only one is visible. This is for compatibility with

other OSes that use this data. The data partition in this case is number 4, so you would use, e.g. /dev/sda4 instead of just /dev/sda

## »» How can I automount removeable media /networked devices?

there are two solutions for this – the *autofs* kernel system, and the *supermount* patch, used most noticeably by Mandrake. This latter eliminates the need to unmount, for example, a CD before ejecting it.

*Autofs* is used mainly for network mounts. These are automatically negotiated when you try to access them, so there isn't a heavy network burden on leaving 'shared' file storage space open. *Autofs* is part of the more recent kernel series, and a full HOWTO is available at [www.linuxdoc.org](http://www.linuxdoc.org)

## »» How come I have SCSI devices but I don't have a SCSI controller

Probably because your system is using the *sg* and *ide-scsi* modules. The *ide-scsi* module pretends that suitable IDE drives are in fact SCSI

« other traffic, such as regular HTTP transfers, work happily, then you'll probably find that it's specific to your ISP's SMTP server, and you can take

it up with them as to what can be done to fix the problems.

All of the browsers you mentioned should be able to perform requests

over SSL for encrypted web pages, although earlier versions were distributed without crypto libraries due to US export restrictions. As you did not say what the appropriate error messages were for the applications, it's difficult to pin down the problem to something specific, so you may end having to just download the latest version of your browser of choice from the Internet, rather than use those distributed on the CDs.

## DVD decryption

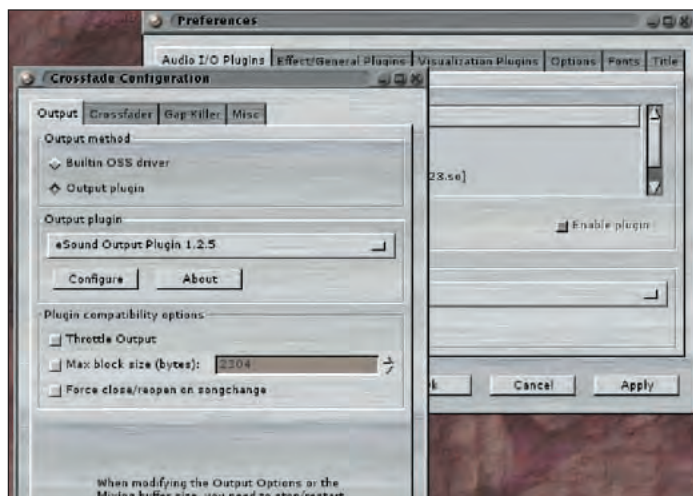
**Q** I installed *Xine* in the hope that it would enable me to watch DVD movies on my Linux box Unfortunately, attempting to play a movie results in the following error message: **There is no input plugin available to handle dvd://VIDEO\_TS.VOB**

Any ideas?  
Steve Roper

**A** *Xine* can only pay unencrypted DVDs, as it is not distributed with any DeCSS code, which forcibly breaks the encryption on regular DVDs, due to the rather questionable legal status of such software.

However, you can happily play DVDs under Linux with *Xine* by using one of the many plugins, such as *dvdnav*, which can be found at <http://dvd.sourceforge.net/>, along with a whole load of documentation regarding the installation and configuration of the plugins.

There are a number of other plugins for *Xine*, such as *dmd*, which offer DeCSS capable decryption features; although each have different capabilities and features, so you will need to decide which suit you best based upon the types of DVD which you watch.



Many applications support the output of their sound through a daemon such as *artsd*.



devices (usually `sg0,sg1` etc). These are often NOT set up using *devfs*, so you'll have to create them if you need to burn CDs and are using the *devfs* module.

The *sg* devices are also commonly used for scanners and other devices which don't fit into other generic SCSI designations: *sd* (for disk-type devices), *scd* (for cd-type devices).

### »» What about USB?

USB disks which use the mass-storage driver are also usually linked to scsi designations - E.g. a camera might appear as `/dev/sda`

### »» What is Devfs?

The device filesystem is a way to reduce the clutter in the `/dev/` directory and make it transparently clear which devices are present. Before *devfs* the `/dev` directory on most distros contained pretty much every device allocation it could think of, just in case they were used.

The *devfs* system makes it much easier to see which devices actually exist, by only creating links for devices that are really there.

You can, of course, install more than one plug-in, and select which one you want to use via the *Xine* panel.

## Sound device

**Q** Could you please help me with a very strange sound problem? It seems to have been caused by a program on your coverdisc, so several other readers may also have been affected.

I am using Mandrake 7.2 and have a Creative Labs SoundBlaster 128 sound card. Everything had been working perfectly for months - until I installed *GLAME* from this month's coverdisc.

Since then *XMMS* has started producing the error message **Can't open /dev/dsp - file or device in use** when I try to play an audio file, even though no

other program associated with audio is running. The command line programs *play* and *plaympeg* simply crash without displaying any error message or playing the file, whereas before they worked correctly.

Cold reboots do not cure this problem, and neither has completely removing *GLAME* from my system by typing **make uninstall**.

However, *Audacity* (a sound recorder, player and editor) still works properly. Once I have run it and opened a sound file, *XMMS* starts working again. (Surprisingly, I do not even have to play the sound file or close *Audacity*.) On the next boot, the problem returns.

Stranger still, the problem only occurs when I use KDE (version 2.0).

If I run GNOME instead, *XMMS* works straight away - even if I have run nothing else since booting the system. The problem comes back if I then log out of GNOME and start KDE.

The problem affects all users so it cannot be something wrong with my personal KDE settings. Nor is it likely to be a driver problem as the boot and shutdown messages indicate that the driver is being loaded and unloaded without errors.

Also I know that it is not a hardware fault as I have a dual-boot system and have never seen any sound problems in Windows 98.

Could it be something to do with the *ArTS* daemon, which I believe is involved with multimedia in KDE? Does it have a configuration file which might have been altered by *GLAME*?

One final question: Is there a way to find out which process, if any, is using `/dev/dsp`? If so then the cause of the problem might well become obvious.

Alistair Green

**A** Taking the last question first just to confuse everyone, it's very straightforward to find out what process is using `/dev/dsp`, by using *lsof*:

```
root:~# lsof /dev/dsp
COMMAND PID USER FD TYPE
DEVICE SIZE NODE NAME
sinek 709 david 9w CHR
14,3 1383 /dev/sound/dsp
```

```
root@niamh2:root (pts/11)
PUMP(8) Linux Administrator's Manual PUMP(8)
NAME
pump - configure network interface via BOOTP or DHCP protocol

SYNOPSIS
/sbin/pump [-krRsdP] [-c ARG] [-h hostname] [-i iface] [-l
hours] [--lookup-hostname] [--usage]

DESCRIPTION
pump is a daemon that manages network interfaces that are
controlled by either the DHCP or BOOTP protocol.

While pump may be started manually, it is normally started
automatically by the /sbin/ifup script for devices config-
ured via BOOTP or DHCP.

Once pump is managing an interface, you can run pump to
query the status of that interface. For example,
/sbin/pump -i eth0 --status
will print the current status of device eth0.
```

**Pump is an alternative to *dhcpcd*, and is often more suited to certain DHCP servers.**

```
sinek 727 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 728 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 729 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 730 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 731 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 732 david 9w CHR
14,3 1383 /dev/sound/dsp
sinek 733 david 9w CHR
14,3 1383 /dev/sound/dsp
```

The chances are, *artsd* is hanging onto `/dev/dsp`, and only *artsd* aware applications are working correctly by sending their sound output through the daemon, rather than trying to open `/dev/dsp`.

## CD-RW recognition

**Q** I have had problems trying to get Linux installed on to my machine, which is a AMD Athlon 800 with three hard disk drives (1 Windows 98, 1 Linux and 1 Windows NT using NTFS); and a Samsung IDE CD-RW.

The latter piece of hardware, I found out, is the problem. Red Hat 7.1 will not mount or use the Samsung CD-RW, but the old version of Red Hat 6.1 will!

To get Linux installed on my machine I had to borrow a CD and replace the CD-RW, and it installed fine.

Do you have any ideas why this has happened? I contacted Red Hat, as it was a boxed set I bought, but have had no response.

I have been playing with Linux for the past 3 years (on a very limited basis) having bought

various books with free version of Red Hat, it seem funny to me that the old version was able to use the CD-RW as a CD, but the new version cannot.

Peter Cole

**A** There are a number of things which could cause this problem, and the best place to look is in `/var/log/messages` just after the machine has booted up, so you can see exactly what devices have been detected, and what `/dev` entries they will be accessible under. It may be possible that you are using the ide-scsi emulation, so the CD-RW will be accessible with `/dev/sr0`, rather than `/dev/hdc`, assuming it's an ATAPI device in the first place. If your BIOS detects it correctly, then you should take a look in `/dev/ide`, and see what drives are shown there.

There is really no difference between a CD-RW and a CDROM drive, so quite why Red Hat 7.1 is assuming that there is, other than if there is a fundamental difference with the two drives you have used. You didn't say at which point of the installation Red Hat 7.1 failed at, so you may need to investigate what you can do to work around the problems during installation.

## Bootable CDs

**Q** Excuse this newbie asking what is probably a pretty basic question. I want to install the RH 7.2 distro which was on the Xmas 2001 LXF DVD, so I copied the iso images to three CDs. However even though I've set my BIOS to boot from CD first when I boot up, my PC still rushes head long into Windows 2000.



« I have read somewhere that not all CD drives can boot from iso images – is this right? I've also tried to boot from my DVD drive, but still no joy, so I guess that what I need is a Boot.img on a floppy to boot from instead – the things is where can I get this from?

Am I on the right track or am I barking up the wrong tree?

Any help gratefully received.

Trevor Scowen

**A** If you don't want to throw out your CDs and burn them again, you can download the boot image from <ftp://ftp.redhat.com/pub/redhat/linux/7.2/en/os/i386/images/>, or look in the 'images' directory on one of the CDs you created. These can be written to a floppy disk using the *rawrite* application in DOS, or using *dd* in Linux/BSD/nix.

The ISO images contain the information to make them bootable, so if they don't boot up, it's likely that you need to change your BIOS settings to enable booting from a CD..

## Missing backspace

**Q** I know I should spend more time reading the "effing" manual, but I'm feeling lazy at the moment. Forgive?

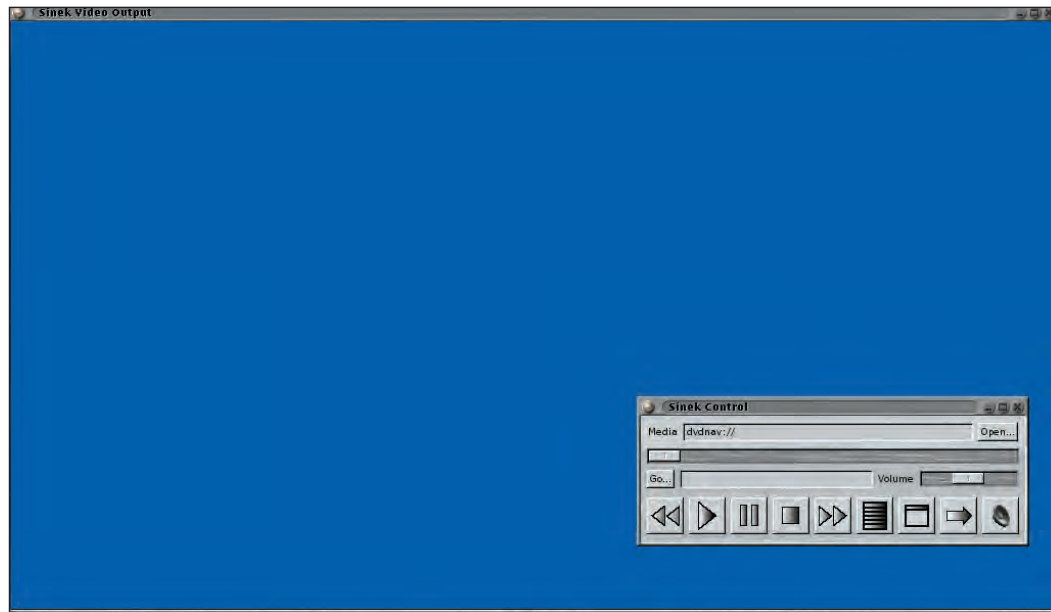
I'm running Debian 2.2r5, and my problem is that *bash* does not seem to recognise my backspace key. (The delete key works all right, though.) I have vague memories of using *stty* to mess around with key bindings when I used AIX at university, but that was 6+ years ago. Anybody care to refresh my memory? Is *stty* the way forward?

Incidentally, I only have this problem in X, which makes me think it's not a shell thing.

From the LXF forums

**A** On certain installations with *XFree86* 3.3.6, the backspace key does not get mapped to 'BackSpace' correctly, so any X clients, such as *xterm*, don't think you're really pressing the backspace key and it won't work correctly. The quickest way to fix this is to use *xmodmap* to setup a mapping from the physical backspace key, to the 'BackSpace' keyboard event in X, which you can include in your *.xsession* or *.xinitrc*;

```
xmodmap -e "keycode
22=BackSpace"
```



When you try to take a screenshot of *Xvideo* output, you end up with a coloured block where the video used to be, so you need to use a different video output method with your application.

## Dropped IP address

**Q** I am running Red Hat 7.1 server install which is connected to a Telewest cable modem, it gets the IP address from Telewest via DHCP (*dhcpcd* 1.3.18p18-10). The server acts as a gateway/firewall to my local network. I am running an *iptables* firewall on the server.

From time to time the Telewest IP address gets dropped (if I do *ifconfig* the IP address line is missing). I can reload the interface with *dhcpcd -n eth0* and the IP address gets picked up okay – but this is a bit of a nuisance.

Any suggestions whether this is a problem with my server or Telewest? If it is my server how do I correct it?

From the LXF forums

**A** Once *dhcpcd* has picked up an IP to begin with, it should sit there and wait for the lease to expire so it can re-request an IP. Checking */var/log/messages*, and searching for 'dhcpcd' should result in some useful diagnostic information as to why *dhcpcd* did not pick up an IP, although it does sound like a problem with Telewest's DHCP server which hands out IPs to their clients.

Once you have some more diagnostic information resulting from the request for an IP which fails, you can take it up with their tech. support bods.

As a quick fix, you could write a small shell script which checks *ifconfig* for the status of the interface then runs your DHCP client again to get an IP. There are a number of other DHCP clients for Linux, including *dhclient* and *pump*, which you may want to experiment with to see if you have better results.

## Blue movie

**Q** Anyone know how to obtain screenshots of your desktop, and NOT get a blue window when an app is using *libXv* – i.e. I want to get a screenshot of my desktop whilst playing a DVD using *Ogle*.

I know the blue window is because *Xv* runs \*on top\* of your desktop, but have no idea how to capture an image from it. Any ideas? I use RH 7.2.

From the LXF forums

**A** The *Xvideo* extension uses an overlay system, so your X client only outputs a coloured box, and your video card inserts the actual video output over the top of it.

When you come to take a screenshot, *xv*, *Gimp*, or whatever application you use to take the screenshot only captures the coloured box and not the video, and there is no way for something to capture the overlaid video.

The only option is to have your application output using a different plugin, such as *Xlib* or *Shm*,

depending on what the applications supports, rather than *Xvideo*, then the video is actually written to X, rather than just to the video card.

Naturally, this is significantly slower than using *Xvideo*, but it does mean that you can take a screenshot of the output. **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: [lxformat@futurenet.co.uk](mailto:lxformat@futurenet.co.uk)



# Coverdisc

Neil Bothwick is your guide through the wonders of this month's distro – the all-new Mandrake 8.2.



## On the CD

Wherever you see this logo it means there's related stuff on the CD

### Important notice

Before you even put the CD or DVD in your drive, please make sure you read, understand and agree to the following: The Linux Format CD/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 CD/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.

## CD CONTENTS AT A GLANCE

(Disk 2)

mupen64	Bomber
rboy	Instinct
zsnes	GTablature
SNEul	Sylpheed
Fake64	phpMyadmin
Heavy Metal:	Gentoo 0.11
FAKK2 patch	Abiword 0.99
Tripwire 2.3	Perl:
Saint 3.4.6	Netserv code
Portsenry	from this
OpenSSH 3.1	month's tutorial
LSB	Complete code
Bastille	for all nine
Acidlaunch	Kylix tutorials!

## Installing Mandrake 8.2

We have a treat for you this month, the latest version of Mandrake Linux, 8.2. This comes on three CDs, so CD users only get the first two (less a little, more later). DVD users get the full lot, and there's no need to burn ISO image files to CD-R before installing with this one. With some help from Warly at Mandrakesoft, we have created a version of the installer that runs directly from the DVD. We had to trim a little from the second CD in order to fit in the Magazine directory, but it was only some Japanese, Korean and Chinese fonts and one of the documentation translations. We'll probably include these on a future CD, so that you have the full distro eventually. If you can't boot from CD/DVD, read install.htm for details of alternative installation methods.

### Starting installation

Installing Mandrake is a breeze. Put the DVD, or the first CD, in the drive and reboot your computer. You will see a Mandrake splash screen for ten seconds before booting continues. You can press Return to skip this or F1 to get a console prompt where you can type extra options. Typing "rescue" at the prompt boots into a console, which is useful if you have somehow broken your system so that it will not boot properly. You can also change the screen resolution here, which is handy if you have a laptop or TFT monitor, which may not display 800x600 very well. To get a 1024x768 display type **linux vga=791** at the boot prompt. Change the number to **785** for

640x480 or **794** for 1280x1024.

Once the installer has booted and displayed the software licence, you'll be given the choice of a fresh install or an upgrade, and whether you want to use the Recommended or Expert installation method. Expert gives you more choices, but offers sensible default options, so you don't need to be a Linux Guru to use it. This means some of the options mentioned won't be available if you chose a Recommended install. These include choice of the type of mouse and the security level. Some sections will also be skipped if you are upgrading instead of doing a fresh install. The standard security level should be fine for normal Internet use, but you should consider one the the higher levels if you are running any sort of server.

The list on the left of the screen shows the various stages of the install, the current stage has an orange

"lamp", completed stages are green.

You can skip back to a previous stage by clicking the appropriate green lamp. The Recommended install method offers you various partitioning choices, while Expert drops you straight into the DiskDrake partitioning tool. If you plan on altering the partitioning on your disk, the standard advice to back up anything even remotely important applies here.

### What to install?

CD users will be asked which CDs they have: untick the box for disc three. DVD users don't see this question as their installer only uses one disc. You should then be taken onto the package selection tool. It may be tempting to install everything in sight, on the basis that is just may come in handy one day. This isn't usually a good idea. Install what you know you'll use, and the standard programs that are called on by so many other



You can always choose more options if you wish.



There's no need to install in English if it's not your first language.

programs, such as the *MySQL* and *PostgreSQL* database servers. You can always install other software later using the Package Manager. You select packages by categories, then the next screen lets you select or deselect individual packages, provided you have ticked the "Individual package selection" box.

It can take a while to install the packages, depending on the speed of your computer and the number of packages you have chosen to install. If you're installing from DVD, put the kettle on, run a bath or go watch Coronation Street, depending on how much you're installing. CD users will have to hang around to change discs at half time.

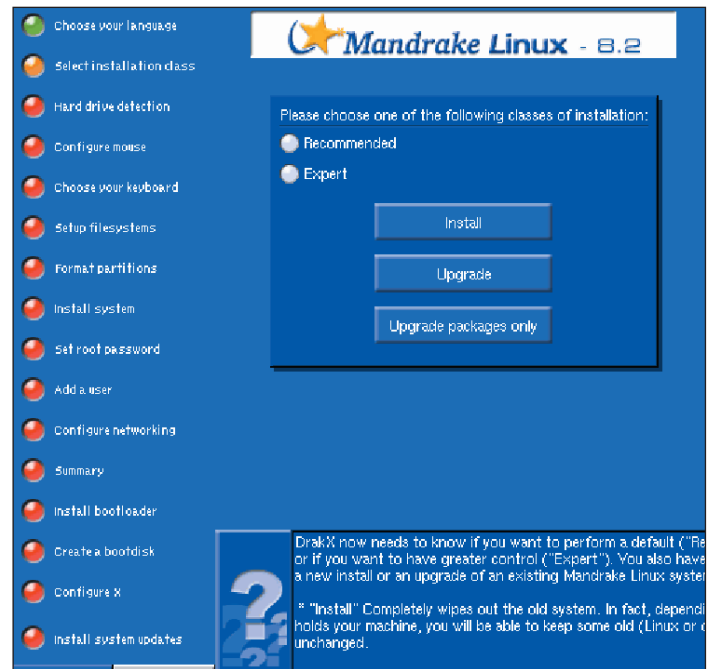
Once the packages are installed, it's time to set up the system. The first thing you're asked for is your root password. There is an option to not set a root password – selecting it would be a really bad idea. Then you need to set up at least one "normal" user, the root account is only used for system administration and configuration. You are then asked whether you want a user logged in automatically, and which window manager they should use by default, the choices depending on which you chose to install. If you are the only user of your computer, it makes sense to enable auto-login. If your computer is in a place where others could use it, it would be best to skip this step and log in manually each time.

## Networking setup

Now it's time to configure your networking and Internet connection. Follow the prompts and give the information needed to connect to your ISP. There are options for modem, ISDN, cable, ADSL and LAN connections. If you connect via a gateway on a local network, you'll need to give the address of this gateway or set the Automatic IP option. If using a modem, it may be a good idea to run Windows before starting the install, and make a note of the various modem properties.

You are then given a list of services to be started at boot time. Check the list and disable anything you are sure you don't need. Don't touch the system section unless you understand what you are doing, but there's no point in running, say, an FTP server if you aren't providing FTP access. Running extra servers increases boot and shutdown times, uses memory and increases the number of potential security weaknesses. You can always turn a service on later if you find you need it.

The bootloader section sets up your hard drive to boot into your new Mandrake system. It will also detect a Windows partition and add a menu item to boot that. If you don't install a bootloader, you will need a boot floppy to start Mandrake. It's worth creating a boot floppy even if you use a bootloader. If your hard drive's boot sector gets overwritten, say by



Choose between flexibility and simplicity for your installation.

reinstalling Windows, the boot floppy get you back into your Linux system where you can recreate the bootloader settings from the *Mandrake Control Centre*.

## Choosing a display

The final stage of the installation is to set up the X Window system. The installer should detect your graphics card, and your monitor if it's reasonably modern, and recommend suitable choices, but use the test option to make sure. Before finishing, there are a couple of optional operations. The first is a chance to connect to the Internet and check for updated packages. The second is hidden behind the Advanced button in the window telling you installation is

complete. It's an option to save your list of installed packages to a floppy disk. Either just the list or as an automated install script. This is useful if you need a similar install on more than one computer.

That's it. Click the OK button and your computer will reboot and let you start using Mandrake. [LXF](#)



Success! You are now booting Mandrake Linux.

## Install or upgrade?

No need to format and start again

If you haven't used Mandrake before, you will have to do a fresh install on a newly-formatted partition(s). But if you have a current Mandrake setup, do you wipe and install from scratch or do you upgrade?

Some people feel that a complete wipe and reinstall is a good way of keeping your system clean and running most efficiently. Others feel this is a rather Windows-influenced viewpoint. If you do a complete reinstall, make sure you backup at least /etc, unless you want to reconfigure everything from scratch.

If you choose Upgrade in the installer, there will be a delay while it checks which packages you have installed. Any packages you currently installed will be automatically marked for installation. As to which is best, if you choose upgrade and it goes wrong, you can still do a fresh install, so you'll be exactly where you were had you chosen that option to start with. However, the converse is not true as you will have wiped all your currently installed software and its configuration files. It's your choice, but you've probably guessed that I prefer the upgrade option.



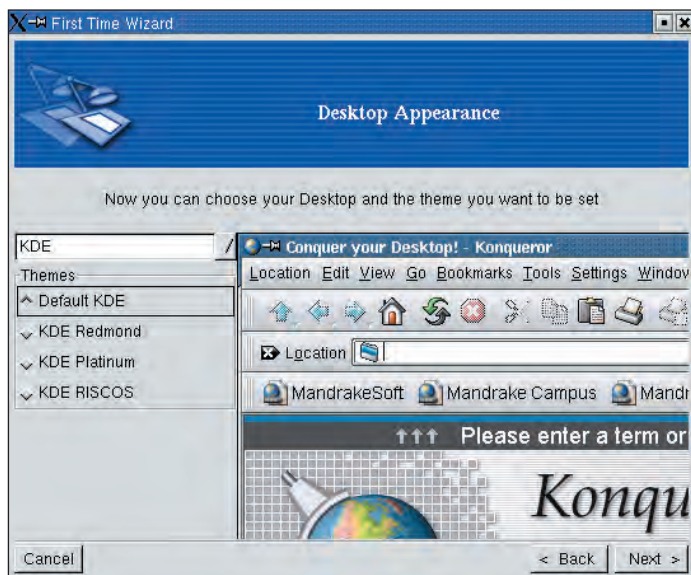
# Mandrake 8.2 – first time

Now that you have Mandrake 8.2 safely installed upon your hard drive, **Neil Bothwick** is your guide to what to do next; showing you how to configure and twiddle until your setup is just right for you.

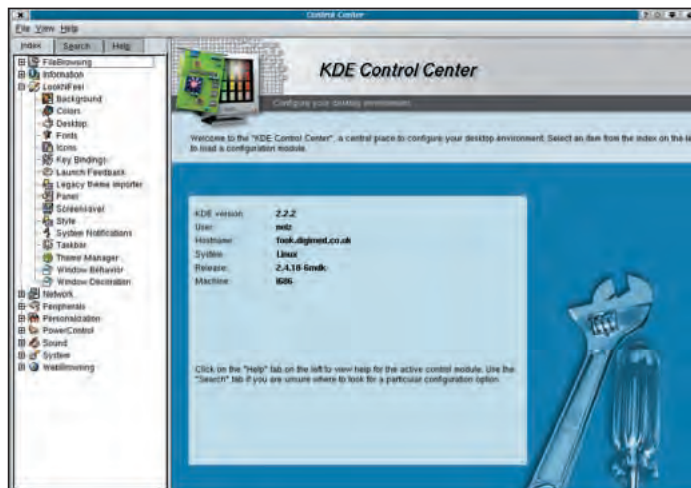
So you've installed Mandrake 8.2 and booted it for the first time. However, instead of getting the desktop you expected, you see a window labelled "First Time Wizard", what's this?

This wizard gives you the opportunity to do some basic configuration and register with Mandrake as a user of 8.2. You should read the statement on what Mandrake do with your data before submitting it, or not.

None of it is compulsory, but you may as well go through the first stage of setting up your default desktop and choosing a theme. If you feel uncomfortable about supplying any sort of registration information to Mandrake, you can skip this by hitting Cancel. If you do so, you won't get to the part where it sets up your email details, so you'll have to start KMail from the toolbar, it's the second icon from the right in the group of icons at the bottom left of the desktop.



The *First Time Wizard* selects the default desktop manager and theme, as well as configuring your email settings.



With the *KDE Control Centre*, you can alter many facets of your desktop's appearance and behaviour.

## KDE Control Centre

Now you've got everything up and running, what are you going to do next? The new "what to do?" section in the KDE menu (bottom left button that looks like a K) contains a selection of programs that you have available. So start some up and start playing.

On the other hand you may be looking at the desktop and thinking it looks a bit boring. If so, start the *KDE Control Centre* by clicking the fourth icon in the toolbar and open the LookNFeel section of the list on the left. When you select a section header, the panel on the right shows a summary of the items it contains and what they do. When you select one of those items, the relevant configuration GUI is loaded into the same panel.

Within this section you can change just about every aspect of the user interface, from colours and background wallpaper to what mouse clicks on various parts of each window and desktop do. There's one idiosyncrasy to watch out for, some mouse behaviour, like acceleration and whether you use a single or double click to open an icon, is set in the Peripherals section. A common complaint about KDE, the default desktop manager for Mandrake, is that

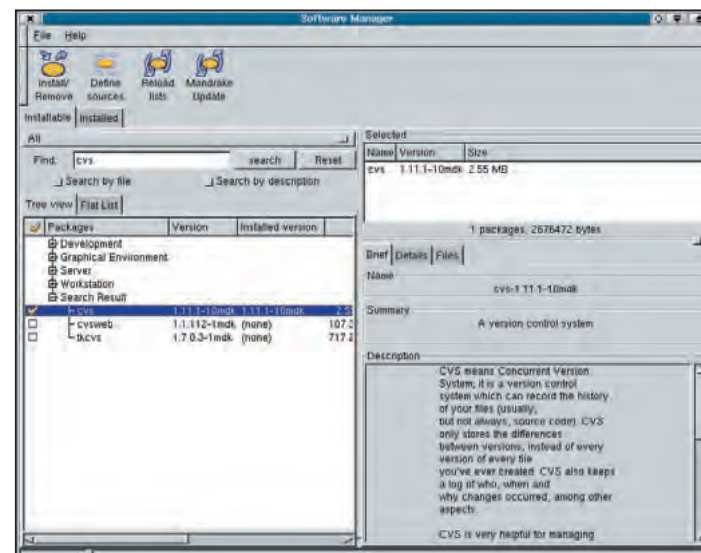
it is too like Windows. While this is true, that's only to make the transition easier for people moving over. There are enough options to make it as unlike Windows as you like, you should be able to get it to look and feel the way you want. Don't adapt to fit your computer's way of working, make your computer change to suit you.

You may have noticed that the windows in the various screen grabs use two different styles. That's because some are taken from a machine with a fresh install of Mandrake, with everything set at the default, the other is from a working machine set up to suit my taste (which probably won't suit yours, but it is my machine).

Most of the changes in this section are at a user level, log in as another user and you'll get a different set of settings. Some parts of the System section affect all users and you'll need the root password to change them.

## Installing/updating

Rather confusingly, there is a *Mandrake Control Centre* in addition to the KDE one. This is a very different program, dealing with things like hardware, booting and software installation. We'll look at the latter. Start the *Mandrake Control Centre*, it will ask you for the root password.



The *Software Manager*, about to install another package.





# Coverdisc

**Neil Bothwick** is your guide through the wonders of this month's jam-packed *Linux Format DVD* – dive in and enjoy.



**T**his month's *LXF DVD* contains over 230 packages, impossible to cover in a few pages, so here's a taster of just a few of the goodies on your coverdisc.

## Desktop/knoda

Linux has some excellent database server software, but much of the front end software is web-based, often requiring you to run a web server as well as the database server. *Knoda* is an attempt to provide a more desktop-oriented solution. It still requires a *MySQL* server to be running (locally or elsewhere), but to the user it presents a typical windowed desktop interface.

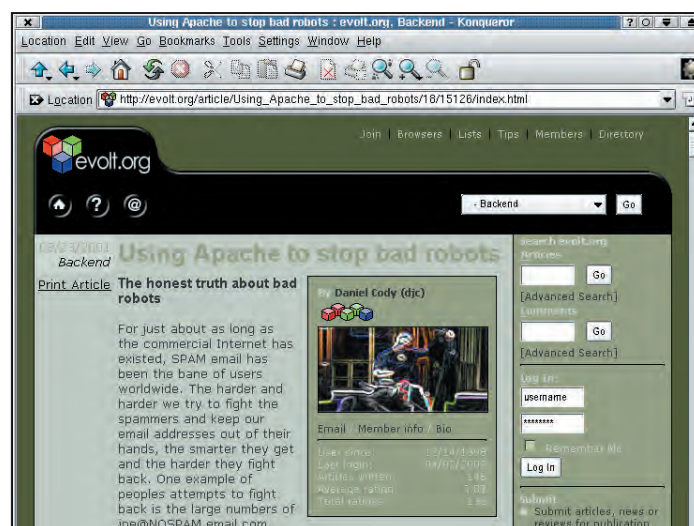
*Knoda* needs *hk\_classes* to be installed, the archive is in the *Knoda* directory. Read the docs before installing *Knoda* as there are options you need to pass to its config script.

*Knoda* can create and modify databases and tables, as well as performing queries on the data. Queries can be saved for later reuse. With a version number of 0.5.1, it is still incomplete, but it does hold promise as an easy way of searching and maintaining SQL databases. Try it now, then keep an eye on its progress.

## Desktop/Wine

No matter how many useful programs there are for Linux, sooner or later you may want to use a Windows program, or access a web site that only works with *Internet Explorer*. There are three main ways of running Windows programs. Dual booting is the obvious one, and the least convenient. You can't run Windows and Linux programs at the same time. When you switch it takes a couple of minutes to shut down one OS and start up the other, and again when you switch back.

Emulators like *VMWare* and *Win4Lin* solve this problem, but they can be expensive to buy if you only need Windows occasionally. There's also the memory overhead of running two operating systems at the same time. The third option is to run Windows programs directly from Linux, yes really, it can be done. *Wine* is a project to replace the standard Windows libraries with Linux/Unix versions. This means it won't run everything – some programs don't react well to running in such a mixed environment. However, a useful number of programs do work well with *Wine*, including *Internet Explorer*. *Wine* doesn't justify lame "webmasters" who don't realise or care that there is more than one web browser, but it can make it easier for you to get things done.



**Robotcop** – keep the spam harvesters away from your email addresses.

## Graphics/Gspy

*Gspy* uses any *video4linux* device to act as a security camera, recording images to a daily MPEG movie. It would be rather boring to watch a movie where nothing moves all day, so *Gspy* uses some clever motion detection software to eliminate pictures that are similar to the previous one. The result is that it only saves a frame of the movie file when something happens, when it detects movement. The frames are time stamped, so you can not only see what happens, but when. There is currently no option to send an alert when something happens, although this is planned for the future, along with several other enhancements.

Installation is fairly straightforward once you have a suitable *video4linux* device connected and working. Unpack the archive and as root type

```
./autogen.sh
make
make install
```

If you have any problems getting it to work with your *video4linux* device, there are some helpful guidelines at <http://gspy.sourceforge.net>, as well as a

tale about how someone recorded a car thief in action while testing with a camera pointing out of the window. No wonder police forces are considering switching to Linux!

## Internet/Morphon CSS-Editor/

Cascading Style Sheets (CSS) are considered an important move forward for the web. They help separate web content from the layout and style information, making sites and pages easier to maintain. However, you still need to create the stylesheets to get the layout you want, and there are many more options here than with HTML 4, so where do you start?

A purist would probably insist on using a text editor, most likely *vi*, and coding everything by hand, but there are tools to make the job easier. They are no substitute for understanding how HTML and CSS work, but they can improve productivity. The *Morphon CSS-Editor* is one such tool. This is a commercial product. We have an evaluation version on the DVD that works for thirty days. To install it, **cd** to the *Internet/MorphonCSS-Editor*



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.

directory of the DVD and run one of the install scripts:

```
sh csseditor-linux-vm1.4-1.2.sh
```

Do this as a normal user, not as root, and the editor will be installed into your home directory. Then run the program by going to your home directory and typing

```
./Morphon_CSS-Editor_1.2
```

You'll be asked for a licence key the first time you run it, so go to <http://www.morphon.com/csseditor> to register for a thirty day evaluation key. This should be mailed to you immediately, paste it into the requester and you've got a month to experiment with the various options.

## Server/Robotcop

There is a standard means of stopping web spiders from accessing certain parts of your web site. The robots.txt file lists which parts are off-limits and is respected by all reputable web crawlers, such as those used to generate the data for search engines. But there is another type of web crawler, the kind that searches web sites for email addresses to add to spam lists. These don't respect the robots.txt file, let's face it spammers don't respect anyone, so addresses on your site could be harvested.

*Robotcop* is an *Apache* module that attempts to tackle this by letting you create directories that are marked as off-limits in robots.txt. If an IP address reads the robots.txt file and then attempts to access a blocked directory, it is probably a spam harvester and access is blocked. The program also has its own list of known address harvesters, anything on this list

is automatically blocked. The intercept methods are being refined, which is essential as the harvesting programs also become more ingenious.

If you are responsible for a web site that contains email addresses, you should take a look at this as a means of protecting the privacy of the owners of those addresses.

## Office/WebCalendar

*WebCalendar*, as the name implies, is a calendar system that works through a web server, as a CGI process. Because all the work is done on the server, no software needs to be installed on the computers that access it, making maintenance much simpler, especially when new versions are released, no nightmares making sure everyone is up to date.

*WebCalendar* is supplied with an installation script: you will need to edit this to suit your setup, it needs to know where your CGI scripts live, *sendmail* details, etc. There are some example configurations for different situations, such as running *WebCalendar* on virtual servers. After that you simply run the install script. There's also a daemon to configure if you want *WebCalendar* to send out email reminders, but that's easily set up with *webmin*. *WebCalendar* won't organise your life for you, but it will let people see who should be where and when, which is a start.

**This is only scratching the surface of what's on the DVD, there are hundreds more programs, so put the DVD in the drive and load the index.html file.**



**Webcalendar** – a web-based calendar makes administration easier.

## » DVD CONTENTS AT A GLANCE

### Desktop

**AccessPointSNMPConfig** Configure Access Points based on Atmel chipset  
**AcerTravelMateLaunchkeys** Handles launchkeys and the mail LED on Acer notebooks  
**Advance** Database backed PIM system  
**AfterStep** NeXTSTEP-alike Window Manager, with enhancements  
**Aware** Distributed event framework for systems management  
**Beaver** Lightweight general purpose text editor  
**bubblegum** Watches file access, modification, and inode change times  
**cfgstoragemaker** Graph all the storage devices of one or more hosts  
**CLEX** File manager with a full-screen user interface  
**CrisoftRiccette** SQL/PHP application for cooking recipes  
**CyberBrau** Web-based program to help the home brewer  
**DVBackup** Backs up files to a raw Digital Video data stream  
**EthGraph** Graphical network device traffic monitor based on Qt  
**EviWM** Minimalist window manager for the X Window System  
**FBmodes** Display mode generator  
**gkrellmGIMPS** GKrellm plugin for the GIMPS client  
**GKrellmMSS** GKrellm plugin with a VU meter and a sound scope chart  
**Gktail** Gktail is a log viewer for your desktop  
**GNOME-Commander** File manager similar to Midnight Commander in text mode  
**Gnome-MySQLClient** Simple, quick, and powerful way to access MySQL databases  
**GNOME-PIM** Personal Information Manager for GNOME  
**GnomeMeeting** H.323 Video Conferencing application  
**GPS3d** Manipulate your GPS from your Linux box  
**GPSdrive** Map-based navigation system  
**Gringotts** Stores sensitive data in an easy-to-read but secure form  
**GTK-Dictionary** Frontend to dictionaries stored in a MySQL server  
**GtkLPforCUPS** Frontend for the lpr that comes with CUPS  
**jGnash** Personal finance application written in Java  
**KAlcatel** Manage messages, calls, etc. Alcatel One Touch phones  
**Kalculate** User-friendly calculator for the average desktop user  
**kdelink2wmaker** Converts KDE links to WindowMaker menu data  
**KExchange** Currency converter that gets exchange rates from the net  
**keychain** Manage RSA and DSA keys conveniently and securely  
**KGoodStuff** Provides a button-bar like FvwmButtons or tkGoodStuff  
**KNmap** Graphical frontend for the popular nmap scanner  
**knoda** Database frontend for KDE  
**Less** Terminal based program for viewing text  
**MC-Burn** Midnight Commander extension to mkisofs and cdrecord  
**MD5Backup** Backup utility for that uses a PostgreSQL database  
**mkautosmb** Autodetects the local SMB (Windows) network  
**MultiSETImonitor** Graphical package to monitor SETI activity  
**MyOwnDiary** Simple, flexible, and easy-to-understand Web diary  
**nano** Nano's ANOther editor, or Not ANOther editor  
**PlexorTool** Queries and sets PlexorMMC CDRW/CD-R(W) drives  
**PPR** Print spooler designed for PostScript printers  
**rfbproxy** Recording screen updates from a VNC server  
**Sipcalc** IP subnet calculator  
**ssync** Minimalistic filesystem synchronization utility  
**TeXmacs** Scientific text editor, v1.0  
**TreeLine** Organize and store text data in a tree structure  
**TreeWM** Window manager that tries to implement a new concept  
**webCDwriter** Make a single CD-writer available to all network users  
**Webmin** Web-based interface for system administration for Unix  
**Wine** Wine Is Not an Emulator  
**Xfsamba** SMB (Windows) network navigator  
**Xindice** Open source Native XML Database

### Development

**Aegis** Software configuration management system  
**Albatross** Toolkit for developing highly stateful Web applications  
**Allegro** Multi-platform game library  
**asmutils** Miscellaneous utilities written in assembly language  
**auth** Obtain and verify user credentials  
**buildpkg** Package build system  
**ccaudio** C++-based framework for manipulation of audio data  
**Cel** Prototype-based, object-oriented programming language  
**CLISP** ANSI Common Lisp implementation  
**jaxml** Python module eases the creation of human readable XML  
**JTTui** Ruby library for text mode (curses) user interfaces  
**KDevelop** C/C++ development environment  
**KRPMBuilder** KDE2 application for building spec files and RPM packages  
**libhtmlparse** HTML parsing library written in ANSI C  
**libmikmod** Portable sound library  
**libsqlora8** Makes Oracle access for C programmers much easier  
**Opale.soya** 3D engine written in and for Java  
**OpenGamingSystem** The OGS is for creating and building role-playing games  
**PFE** ANSI Standard Forth programming language  
**psycopg** PostgreSQL database adapter for Python  
**Python** Release candidate of Python 2.2.1  
**SpiritParserLibrary** Library for writing XWindows client applications  
**Tora** OO recursive descent parser generator framework  
**Toolkit for Oracle**



## LinuxFormatCoverdiscDVD

## » DVD CONTENTS AT A GLANCE (continued)

VisualDebugger  
Xalan-Java  
Xclasses

General purpose graphical debugger front end  
Transforms XML documents into HTML or text  
C layout library for the X Window System

**Distros**

CRUX  
Familiar  
Mandrake 8.2  
SmoothWall

Lightweight, i686-optimized Linux distribution  
Linux for the Compaq iPAQ h3600-series of handhelds  
The latest version of Mandrake, installable from DVD  
Popular Internet Security software package

**Games**

Attal-LordsOfDoom  
Bots'n'Scouts  
Empire  
EscapeOfTheUnicorn  
FishWorld  
GeeWhiz  
JSupermind  
KCheckers  
LBreakout2  
Project  
pyDDR  
Reaper  
SearchAndRescue  
SlasheM  
SurvivalOfTheFittest  
Tunnel  
VegaStrike

Turn-based strategy game for solo or network play  
Multiplayer game that can be played over the network  
Real-time war game that has a long tradition  
2D flying shooter game  
Java applet game  
Game in the style of Wizardry  
Clone of the Super Mind mobile game  
KDE version of board game classic "checkers" a.k.a. "draughts"  
The polished successor to LBreakout  
Cross-platform space simulation game  
Clone of DDR - "Dance Dance Revolution"  
Graphics-intensive 3D-game where you fly a spaceship  
Pilot rescue helicopters to rescue victims  
A variant of the hugely popular Rogue-like game NetHack  
Post-apocalypse Web-based RPG  
Control a capsule going down a tunnel  
Linux action space simulator

**Graphics**

AreaJ  
camE  
camserv  
G3DViewer  
GeoTools  
GNOME-SubtitleEditor  
gPhoto  
gspy  
NvClock  
POWA  
QVCDGear  
Vertex3DModelAssembler  
Vobcopy

Digital photography Web application  
Webcam grabber designed for video4linux devices  
Streaming video from Linux, FreeBSD, NetBSD  
3D file viewer supporting a variety of file types  
Java toolkit for developing interactive geographical maps  
GNOME tool for editing and converting DivX Subtitles  
Transfer digital photos from almost any digital camera  
Retrieves video4linux images and turns them into a daily mpeg  
Overclock your Nvidia card  
Operate a webcam either for a website or personal use  
Graphical front end for VCDGear  
3D modeller geared towards making high performance models  
Copies DVD .vob files to hard disk

**Internet**

AdslShare  
Alicq  
Columba  
curphoo  
Fetchmail  
Galeon  
Harvest  
Kym  
LinVPN  
mailm  
Melon  
MorphonCSS-Editor  
Mozilla  
NapsterClient  
nget  
SPASTIC  
SpeedtouchUSBDrivers  
WonderShaper  
Yahoo2mbox

Powerful data sharing tool using the gnutella protocol  
Tcl/Tk ICQ client  
Highly multithreaded Java email client  
Console Yahoo! chat client written in Python  
Remote mail retrieval and forwarding utility  
GNOME web browser based on Gecko  
Collect and search information using a web interface  
KDE Port of Yahoo Messenger based on libyahoo  
Create a Virtual Private Network from Linux to Linux  
Command-line program for creating and sending mail  
Mailbox flag for X  
Cascading Stylesheet (CSS) editor written in Java  
The latest version of the Mozilla web browser  
One of the oldest Napster clients available  
Command line NNTP file grabber  
Powerful set of email filters to deal with spam  
The GPL Speedtouch USB ADSL driver  
Cures latency problems for cablemodem and ADSL users  
Retrieves messages from Yahoo! Groups archives

**Office**

BeanCounter  
ColorQuote  
Freeside  
OpenCms  
OpenOffice  
Scribus  
STOQ  
Turba  
WebCalendar

Stock market data analysis and performance evaluation  
Real time console-based trading tool  
Open-source billing package for ISPs  
Open Source content management system (CMS)  
Open Office, a Linux Format Award winner.  
Scribus is a DTP program for Linux  
Suite of applications for retail business management  
Contact management application  
Powerful group calendaring and scheduling system

**Server**

Apache  
BBclone  
BigSam  
board-tnk  
Checkservice  
DaDaBIK  
Exim  
eZGuestbook  
filebase

The world's most popular HTTP server  
PHP Web counter on steroids  
Built-In Guestbook Stand-Alone Module  
Discussion board written in PHP  
Perl script that monitors services on remote hosts  
Easily create a Web interface for a MySQL database  
Message transfer agent (MTA)  
Small guestbook using eZpublish  
Configurable DirectoryIndex for Apache

GeekLog  
JavaSOS  
Kiowa  
knocker  
Leafnode  
Luxor  
microCODE  
MyDynaWeb  
MyHeadlines  
mysqlconf  
MySQLControlCenter  
NetSaint  
NewsGrabberPerlModule  
NodeRunner  
NotFTP  
NukeLayoutPHPClass  
OpenH323Gatekeeper  
OpenSLP  
Perdition  
PerlWebmail  
PHP  
phpBB  
PHPShell  
phpStreamcast  
phpSybaseAdmin  
phpWebThings  
Post-Nuke  
Robotcop  
Siege  
Squid  
SSRTEch  
Sympa  
TsunamiBench  
Xerces2

**Sound**

AdaptiveSongSelection  
AMPLE  
AudioGalaxyQueryTool  
AurionMusicManager  
Autools  
E-xmms  
EasyTAG  
gAlan  
JavaMusicPlayer  
LilyPond  
MP3Daemon  
Mp3Kult  
MP3Web

Adapt XMMS' playlist randomization to your own taste  
Simple MP3 server  
Automates queries to the AudioGalaxy music sharing system  
Interface for mpg123, cdrecord, mkisofs, lame and others  
Play, record, generate, modify, and analyse audio files  
E-xmms is an Enlightenment applet that interfaces with XMMS  
Viewing and edit tags for MP3, MP2, FLAC and OGG files  
Audio-processing tool for X and Win32  
Java music player that is WinAmp skins 2.0 compliant  
LilyPond is a music typesetter  
Small, portable, and robust server for streaming audio  
Organises your MP3/Ogg collection in a MySQL database  
MySQL/PHP-MP3 search engine for streaming and downloading  
Build music, sound, signal processing, and MIDI applications  
Plugin-based real-time audio/MIDI signal processing software  
Multithreaded MP3 manager and tag editor  
Emulates the SID chip of the Commodore 64  
XMMS plugin to play A52/AC3 files using liba52  
Allows XMMS to be used as an alarm clock  
XMMS plugin for playing the so-called "SID tunes"

**System**

Alfandega  
apt4rpm  
BackerUpper  
BICK  
EnterpriseVolumeManagement  
ethtool  
ExtendedAttributesAndACLs  
gpgk  
GTK+  
IPTablesTutorial  
LekaRescueFloppy  
LinuxEasyAccessKeyboard  
LinuxKernelPatchesCollection

Creates basic iptable rules  
Creates an apt repository from an RPM repository  
Simple, easy to configure backup utility  
Builds a bootable Linux ISO image from a file tree  
Flexibility and extensibility in managing storage  
Net driver diagnostic and tuning tool  
POSIX-like Access Control Lists for Linux  
Command-line GNUUpdate package management tool  
The Gimp Toolkit  
How to install and set up iptables and netfilters  
Mini-distribution that installs into one floppy disk  
Support for Easy Access and Internet Keyboards  
Useful and interesting patches not in the standard kernel  
Drivers for the proprietary filesystem used by Windows NT  
Essential drivers for monitoring hardware  
Automatic upgrade daemon  
Thorough stand-alone memory test for x86  
Client/server based approach to UPS monitoring  
Firewall designed specifically for standalone computers  
Works with iptables/ipchains to detect port scans  
Security tool to analyze the integrity of systems  
Simple backup solution users and sysadmins  
Highly configurable and tunable SDPL  
Catalogs system events in minute detail  
Development kernels for testing purposes only

LinuxNTFS  
lm\_sensors  
MandrakeUpdateRobot  
memtest86  
NetworkUPSTools  
ParanoiaIPTablesFirewall  
PortScanAttackDetector  
RemoteAccessSession  
REOBack  
SimpleDiskPartitionLayer  
TraceToolkit  
WOLK

# Help wanted

If you don't want to feel bad about not getting out and doing the garden in your free time, **Richard Smedley** suggests that you stay in and help with some Free Software projects.

**W**e start the column this month with another *grand projet* built from ideals of freedom of information. Inspired by the Freenet project, askemos goes further by aiming to store *all* data in a distributed form – with advantages not just in fighting censorship and invasions of privacy, but in terms of security, flexibility and growth for corporate data storage and use, as well as in research.

Askemos is an autonomous, distributed operating system on top of

peer to peer networks which significantly raises the level of abstraction in comparison with today's operating systems

Askemos has some outstanding features: A virtual machine is defined at document level, in XML, in terms of abstract trees and pure functional transformation of them. Its access control system is modelled after general key systems – without a superuser to hack. It has persistent processes and utilises fully asynchronous, implicitly parallel computing wherever possible.

This virtual machine has no physical representation at any single computer – all data is distributed and, for security and proof against individual failure, all changes are voted on by at least three different machines.

Askemos will turn a network – or the Internet! – into a huge single computer (from the user's point of view) serving applications anywhere. Sit down at any terminal, install a key and enter a password and you are at your own desktop.

Much help is needed with a project on such a grand scale, but fortunately

tasks have been broken down and quantified, to make it easier for anyone to get involved. Scheme programmers, of various levels, are most in demand, but help is also welcome from those with knowledge of C, SQL, dynamic HTML, SVG and XSLT. Additionally those with sysadmin skills, as well as document writers and someone to help with the website, are being sought.

<http://www.askemos.org/>

Itemised TODO list:

<http://www.askemos.org/ProjectsOnThePlate>

## Opensourcedirectory.org

Getting stable Open Source software

**Steve Mallett writes:**

"Open Source Directory (OSD) is a website for users to find just those apps that have evolved to the *stable* development stage. The aim of the site is to act as a launch pad for companies and individual users to find and use open source apps. That's why the site sticks to stable applications. When new-to-open-source (server, embedded, or desktop) users try open source for the first time it should be an application that 'at least works'.

Of course, the resource has more uses than that. The site collects important meta data information for

developers about each app, sticks to licenses certified by the OSI (opensource.org) so users can verify that an app is actually open source (or verify its license is GPL).

It only works with your participation. They ask that developers register their stable, open source apps, that users insist on complete and accurate information (many eyes make all bugs shallow...) and that volunteers become application editors themselves if their favourite application is absent. OSD also has a social contract, based on the Debian social contract, that ensures that the database info is



**www.OSDir.org – if you want “stable,” rather than bleeding edge.**

licensed under the GFDL (GNU Free Documentation Licence).

OSD's site founder is also the current opensource.org webmaster."

<http://OSDir.org>

<http://OSDir.org/docs/site/>

[social.php](http://social.php)

<http://opensource.org/licenses/>

## Twisted

Event-based python framework for for building networked servers and clients

**Itamar Shtull-Trauring writes:**

"I'd like to flag one of my favourites, Twisted. We really need people to write documentation, and people to think up cool new ways to use this software. Twisted is a event-based Python framework for building networked servers (and clients.)

We do have some docs (possibly better than other similar projects) but the amount of ideas and modules is large enough that *any* help is appreciated."

From the Twisted README:

"Twisted is an event-based framework for internet applications.

It includes a web server, a telnet server, a multiplayer RPG engine, a generic client and server for remote object access, and APIs for creating new protocols and services. Twisted supports integration of the Tk, GTK or wxPython event loop with its main event loop.

The Win32 event loop is also supported, as is basic support for

running servers on top of Jython.

Twisted currently supports the following protocols, all implemented in pure python, most of them as both servers and clients:

- FTP
- HTTP
- SOCKSv4
- SMTP
- IRC
- telnet
- POP3
- AOL's instant messaging TOC

■ OSCAR, used by AOL-IM as well as ICQ (client only)

■ DNS

■ LDAP (client only)

■ finger

■ Echo, discard, chargen and friends

■ Twisted Perspective Broker"

For more information, visit

<http://www.twistedmatrix.com>

or join the list at:

<http://twistedmatrix.com/cgi-bin/mailman/listinfo/twisted-python>



## cdrecord

### Powers your CD-writer

Some days you get no developing done because you spend all day answering support calls. Many of you will have used *cdrecord* – often on a day-to-day basis. It is one of the essential \*nix apps. If you would like it to continue to evolve then please help out the developer, Jörg Schilling, by subscribing to the lists and using your *cdrecord* experience to help to answer support mail there. Basic familiarity with the application will enable you to answer a number of questions straight off.

The two busy lists are:  
 cdrecord-support@lists.berlios.de  
 other-cdwrite@lists.debian.org

To subscribe to the second support list email  
**other-cdwrite-request@lists.debian.org**  
 with the word subscribe in the body. And visit  
<http://lists.berlios.de/mailman/listinfo/cdrecord-support>  
 to subscribe to the first.



## Your help wanted

There are thousands of Free Software projects in need of some sort of help and thousands of Linux Format readers who may be interested in assisting your project. However we cannot publicise what we don't know about.

If YOU have a project that's in need of anything from artists and beta-testers to web-designers and, er, something beginning with Z, we want to hear about it. Email us now at [linuxformat@futurenet.co.uk](mailto:linuxformat@futurenet.co.uk) and give us some details of your project, and what sort of help you are looking for. Please include plenty of info about the project!

## j'ai compris

### OFSET's educational projects

#### Where better to explain the

principles of community, sharing and freedom behind GNU/Linux than in education – which surely aspires to those same values? OFSET (Organisation for Free Software in Education and Teaching) is a French-based, international organisation who certainly believe this. Their freeduc (the **Free Education** software database) project needs a hand finding and adding Free educational software to the database, and making Debian packages.

Full details are at their website, where you will also see two other projects that can do with a hand: *GCompris* (pronounced 'j'ai compris' – French for "I have understood") and *Dr. Genius*. In both cases help would be welcomed with the documentation, which was written by non-native English speakers.

*Dr. Genius* is an interactive geometry program incorporating a *Matlab*-like calculator with an interpreted language, GEL, plus a large mathematical library written in GEL. It allows one to create geometric figures and manipulate them interactively within specified geometric constraints – it's an excellent education tool for secondary schools. *Dr. Genius* is the product of merging Hilaire Fernandes' *GTK Dr. Geo* and George Lebl's *GNOME Genius calculator*. It has been considerably worked upon since then, the latest development has been making it Guile-aware and help is needed with developing a consistent API to bind the geometric engine to Guile.



Further help is needed with various algorithms and changes to the architecture to enable more plug-ins. And yes, naturally this is one of those cool projects with a self-recursive acronym: **Dr Genius Refers to Geometry Exploration and Numeric Intuitive User System**.

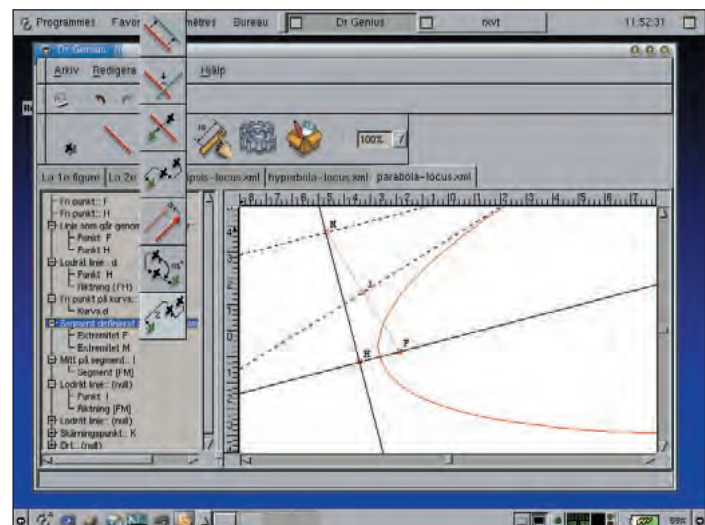
Earlier on in the education process *GCompris* teaches children from age three to use a mouse and keyboard, and educates and entertains them a little along the way. The program consists of a number of game 'boards'. Each board can contain a different sort of educational content – e.g. shape matching; learning the time; counting – so *GCompris* is not so much a program as a whole (pre-school) educational resource. Sound support (through *ogg123*) has expanded the range of boards to include, for instance, letter recognition. *GCompris* is a match for some proprietary pre-school edutainment programs and its easy extensibility should give it a good future, but more help is always needed. The plug-in architecture means that a minimum of C coding is needed to add a new board.

Last month *GCompris* reached the milestone 1.0 release, however there remains much to be done, and all manner of plans to improve the app. Currently *GCompris* boards must be implemented with the GTK Canvas Widget; it is possible that the program may be ported to a better API, such as SDL. If you are one of the many coders getting into SDL this would be a good project to get involved with.

[www.ofset.org](http://www.ofset.org)



One of *GCompris*' many colourful boards – a matching game.



*Dr. Genius* is flexible and easy to use, and has recently been made Guile-aware – more help is now needed on the feometric engine.

# User Groups

Your local Linux User Group needs you! LUGs worldwide are full of members keen to help with your problems, discuss ideas and generally natter about all things Linux. We have collected a load of information here so you can find the LUG closest to you. You can find lots more information online at: [www.lug.org.uk](http://www.lug.org.uk) or <http://lugwww.counter.li.org/groups.cms>

## 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)  
Contact Tony Dyer

## 4 Oxford

URL [www.oxford.lug.org.uk](http://www.oxford.lug.org.uk)  
Contact Alasdair G Keron

## 5 Kent

URL [www.kent.lug.org.uk](http://www.kent.lug.org.uk)  
Contact John Mills

## 6 Brighton

URL [www.brighton.lug.org.uk](http://www.brighton.lug.org.uk)  
Contact Johnathan Swan

## 7 Sussex

URL [www.sussex.lug.org.uk](http://www.sussex.lug.org.uk)  
Contact Mike Pedley

## 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.doncasterlug.org.uk](http://www.doncasterlug.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.wolves.lug.org.uk](http://www.wolves.lug.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/>

## 20 Surrey

URL [www.surrey.lug.org.uk](http://www.surrey.lug.org.uk)  
Contact Jay Bennie

## 21 Cambridge

URL [www.cam-lug.org](http://www.cam-lug.org)

## 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 <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 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.sderby.lug.org.uk/](http://www.sderby.lug.org.uk/)  
Contact Dominic Knight

## 36 Belfast (BLUG)

URL [www.belfastlinux.cx](http://www.belfastlinux.cx)  
Contact Ken Guest

## 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](http://www.sl.lug.org)  
Contact Ben@benguin.co.uk

## 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 [www.midlandsLUG.cjb.net](http://www.midlandsLUG.cjb.net) **WARNING: Popup ads**  
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 Beanz and Tracy

## 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.southwestlug.uklinux.net/](http://www.southwestlug.uklinux.net/)  
Email [southwest@lug.org.uk](mailto:southwest@lug.org.uk)

## 46 South Wales

URL [www.sw.lug.org.uk](http://www.sw.lug.org.uk)  
Contact Tim Bonnell

## 47 North London

URL <http://www.kemputing.net/alt/lug/anlug.html>

## 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 Adam Brookes

## 50 Nottingham

URL [www.nottingham.lug.org.uk](http://www.nottingham.lug.org.uk)  
Contact Godfrey Nix

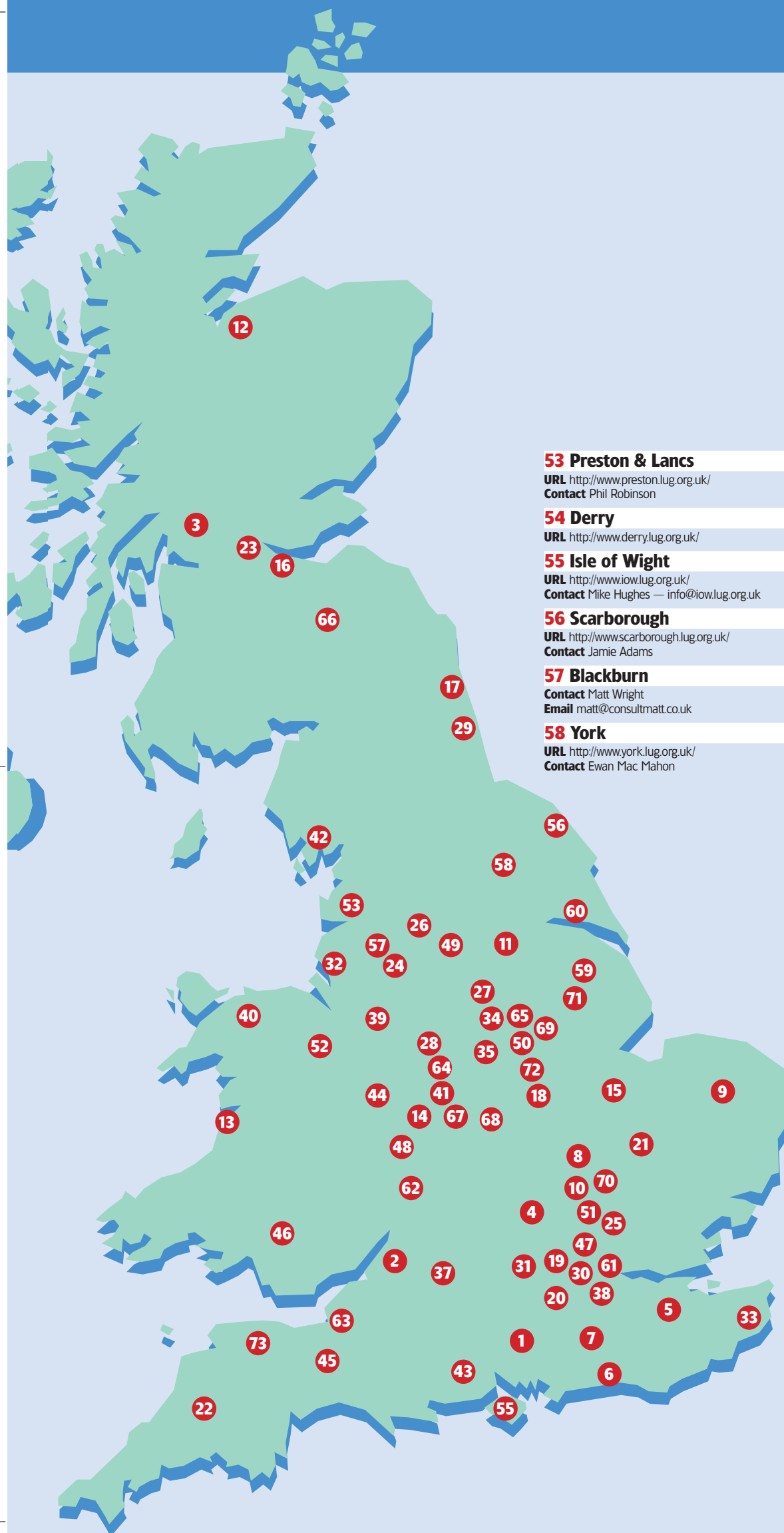
## 51 St Albans & Luton

URL <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** <http://www.preston.lug.org.uk/>  
**Contact** Phil Robinson

**54 Derry**

**URL** <http://www.derry.lug.org.uk/>

**55 Isle of Wight**

**URL** <http://www.iow.lug.org.uk/>  
**Contact** Mike Hughes — [info@iow.lug.org.uk](mailto:info@iow.lug.org.uk)

**56 Scarborough**

**URL** <http://www.scarborough.lug.org.uk/>  
**Contact** Jamie Adams

**57 Blackburn**

**Contact** Matt Wright  
**Email** [matt@consultmatt.co.uk](mailto:matt@consultmatt.co.uk)

**58 York**

**URL** <http://www.york.lug.org.uk/>  
**Contact** Ewan Mac Mahon

**59 Lincs**

**URL** <http://www.lincs.lug.org.uk/>  
**Contact** Chris Lingard

**60 Hull**

**URL** <http://www.hull.lug.org.uk/>  
**Contact** Chrus Burton [chris@hull.lug.org.uk](mailto:chris@hull.lug.org.uk)

**61 East London**

**URL** <http://www.eastlondon.lug.org.uk/>  
**Contact** Jonathan Spriggs

**62 Gloucestershire & Cotswolds**

**URL** <http://www.gloucs.lug.org.uk/>  
**Contact** Barrie Haycock

**63 Yeovil College**

**URL** <http://www.yc.lug.org.uk/>  
**Contact** Adam Parker

**64 South Staffordshire**

**URL** <http://www.staffs.lug.org.uk/>  
**Contact** Oliver Keenan

**65 Mansfield**

**URL** <http://www.mansfield.lug.org.uk/>  
**Contact** Brent Vardy

**66 Borders**

**URL** <http://www.linux.bordnet.co.uk/>  
**Contact** Welby McRoberts

**67 South Birmingham**

**URL** <http://www.sb.lug.org.uk/>  
**Contact** Tim Williams

**68 Coventry**

**Contact** Darren Austin  
**Email** [info@coventry.lug.org.uk](mailto:info@coventry.lug.org.uk)

**69 Newark**

**URL** <http://www.newlinc.lug.org.uk/>

**70 Bedfordshire**

**URL** <http://www.beds.lug.org.uk/>  
**Contact** Neil Darlow

**71 Lincoln**

**URL** <http://www.lincoln.lug.org.uk/>  
**Contact** Jon Shamash

**72 Loughborough**

**URL** <http://www.loughborough.lug.org.uk/>  
**Contact** Martin Hamilton

**73 Exeter University**

**Contact** Nicholas Murison  
**Email** [N.J.Murison@exeter.ac.uk](mailto:N.J.Murison@exeter.ac.uk)

**NEW  
DETAILS**

# LinuxUserGroups

## LUG OF THE MONTH!

### Sheffield

Sheffield Linux User's Group has grown up quite a lot since it started in January 1999. We have our own web site, a PHP-based, Phorum bulletin board and a very active discussion list.

In the past we have organised things like a Linux Demo Week at a local book shop, and more recently we held a DTI-sponsored Linux seminar with UK Club Online at

a local football stadium.

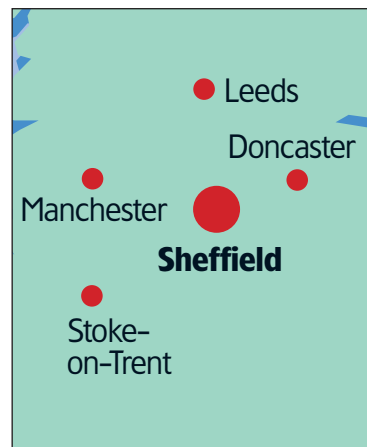
We are presently trying out an evening beer and laptop session at a local hotel, in addition to the monthly meetings at a local University.

We have our own stand at the Birmingham Linux Expo in May, which we very much hope will be a big success. Our GNU/Linux and BSD users come from all walks of life. We have

BT employees and coders from some well known international organisations, as well as home users who want some help with unravelling their installation and config problems. If you do want help or advice then please come along and ask questions.

Bring your computer if that is what you want to do. We hope to see you soon.

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



## Worldwide Linux User Groups

Free Software users across the globe

### Africa

#### PRETORIA

URL [www.plug.za.org](http://www.plug.za.org)

Email [andriesn@icon.co.za](mailto:andriesn@icon.co.za)

#### STELLENBOSCH

URL [www.entropysun.ac.za/](http://www.entropysun.ac.za/)

Email [ixion@entropysun.ac.za](mailto:ixion@entropysun.ac.za)

### Australia

#### ADELAIDE LUG

URL [www.linuxsa.org.au](http://www.linuxsa.org.au)

Email [mtippet@anu.edu.au](mailto:mtippet@anu.edu.au)

#### MELBOURNE, VICTORIA

URL [www.luv.asn.au](http://www.luv.asn.au)

Contact [luv-committee@luv.asn.au](mailto:luv-committee@luv.asn.au)

#### PERTH

URL [plug.linux.org.au](http://plug.linux.org.au)

### Europe

#### AUVERGNE

URL [www.linux-arverne.org/](http://www.linux-arverne.org/)

Email [Cyril.Hansen@wanadoo.fr](mailto:Cyril.Hansen@wanadoo.fr)

#### 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)

#### GHENT

URL [lsgg.rug.ac.be/](http://lsgg.rug.ac.be/)

Email [wvdputte@lsgg.rug.ac.be](mailto:wvdputte@lsgg.rug.ac.be)

#### GOTHENBURG

[nain.oso.chalmers.se/LUGG/index.html](http://nain.oso.chalmers.se/LUGG/index.html)

#### UK

Don't forget the distribution-specific mailing lists:

URL <http://www.lug.org.uk/maillist.html>

### India

URL [www.river-valley.com/tux/index.html/](http://www.river-valley.com/tux/index.html/)

Email [anil@river-valley.com](mailto:anil@river-valley.com)

URL [www.linux-india.org](http://www.linux-india.org)

Email [newsmaster@linux-india.org](mailto:newsmaster@linux-india.org)

### Middle East

#### EGYPT

URL [www.linux-egypt.org](http://www.linux-egypt.org)

Contact Hesham Bahram

### North America

#### ALASKA

URL [www.aklug.org/index.html](http://www.aklug.org/index.html)

Email [deem@wdm.com](mailto:deem@wdm.com)

#### BATON ROUGE

URL [www.brug.net/](http://www.brug.net/)

Email [dpuryear@usa.net](mailto:dpuryear@usa.net)

#### BAY AREA

URL [www.balug.org/](http://www.balug.org/)

Email [afyde@balug.org](mailto:afyde@balug.org)

#### CLARKSVILLE, TN

URL <http://www.clug.org>

Email [tux@clug.org](mailto:tux@clug.org)

#### DENVER

URL [spot.elfwerks.com/~clue/](http://spot.elfwerks.com/~clue/)

Email: [lynnd@ihs.com](mailto:lynnd@ihs.com)

#### FLORIDA

URL [www.flug.orgm](http://www.flug.orgm)

#### LOS ANGELES

URL [www.lalugs.org/](http://www.lalugs.org/)

Email [dank@alumni.caltech.edu](mailto:dank@alumni.caltech.edu)

#### NORTH COLORADO

Email [nclug@nclug.org](mailto:nclug@nclug.org)

Contact Mat Taggart

#### TAMPA

URL [terrym.com/slug/index.html](http://terrym.com/slug/index.html)

Email [paulf@quillandmouse.com](mailto:paulf@quillandmouse.com)

#### UHACC Normal, IL

URL <http://www.uhacc.org/>

Email [lug@uhacc.org](mailto:lug@uhacc.org)

#### VIRGINIA TECH

URL [corvette.me.vt.edu/pages/index.html](http://corvette.me.vt.edu/pages/index.html)

Email [nega@vt.edu](mailto:nega@vt.edu)

### South America

Will be squeezed in next month.

Apologies for the omission

## Linux Install Day



"I can't believe how much better things are since you put Linux on for me!" – last year's Install Day.

### 19<sup>th</sup> May 2002

Install Days (or 'Fests') have long been an embodiment of the community spirit that surrounds Linux and Free Software, giving visitors the chance to bring along their own PCs and install Linux under the supervision of local 'gurus'.

Inspired by the success of last year's UK Install Day, Dorset LUG have taken on the task of coordinating this years events.

Linux User Groups from around the UK will be coordinating events in May and June when visitors will be able to

view demonstrations of the power and freedom of Linux at work in home and business scenarios.

UK Linux Install Day 2002 gives Linux users the chance to promote the benefits of Linux to the wider community, including home users, businesses and schools.

LUGs who want to run an Install Day Event should contact John Robinson on [marketing@linuxinstallday.org](mailto:marketing@linuxinstallday.org) and those interested in attending should visit the website to find an event near them.

<http://www.linuxinstallday.org/>

## 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)**Art Editor** Julian Jefferson [julian.jefferson@futurenet.co.uk](mailto:julian.jefferson@futurenet.co.uk)**Reviews Editor** Richard Drummond [richard.drummond@futurenet.co.uk](mailto:richard.drummond@futurenet.co.uk)**Production Editor** Richard Smedley [richard.smedley@futurenet.co.uk](mailto:richard.smedley@futurenet.co.uk)**Editorial Contributors**

Jono Bacon, Chris Bidmead, Neil Bothwick, Chris Brown, David Cartwright, Paul Cavanagh, Andy Channele, David Coulson, Dave Cross, Hoyt Duff, Simon Goodwin, Steve Heaven, Chris Howells, Maurice Kelly, Brian Long, Robert W Rice, Charlie Stross, Tom Wilkinson

**COVER CD PRODUCTION****CD Editor** Neil Bothwick**Senior CD Editor** Kate Hadley**ART CONTRIBUTORS****Photography** Corbis, Mark Hindess, Kath Lane-Sims, Louise Parker, Photonica, Robert W Rice**Illustration** Paul Bateman, Chris Winn, Shane Collinge**ADVERTISING SALES**

Matt Stanley 01225 442244

[matt.stanley@futurenet.co.uk](mailto:matt.stanley@futurenet.co.uk)

Emma Mimmack 01225 442244

[emmamimmack@futurenet.co.uk](mailto:emmamimmack@futurenet.co.uk)**MARKETING AND PROMOTIONS****Marketing Manager** Fiona Tully**Subscriptions Executive** Nick Skardon**PRODUCTION****Production Co-ordinator** Diane Ross**Group Production Manager** Clare Tovey**Advertising Production Co-ordinator** Jo Crosby**MANAGEMENT****Publisher** Dave Taylor**Group Advertising Manager** Simon Moss**Group Publisher** John Weir**Circulation Director** Sue Hartley**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**Simon Wear [simon.wear@futurenet.co.uk](mailto:simon.wear@futurenet.co.uk)**Tel** +44(0)1225 442244 **Fax** +44 (0)1225 732361**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 recognises all copyrights and trademarks. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight.

**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. We cannot be held responsible for mistakes or misprints. Linux Format recognises all copyrights in this issue. Where possible we have acknowledged the copyright holder. Please contact us if we have failed to credit copyright.**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 © Future Publishing Ltd 2002**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**www.thefuturenetwork.plc.uk**BATH • LONDON • MILAN •  
NEW YORK • PARIS •  
SAN FRANCISCO • WROCLAW

# NEXT MONTH

## Issue 28 on sale Friday 24 May

As Mozilla 1.0 appears (hopefully!) we'll be taking a look at not only the biggest desktop app ever written, but at how the Mozilla project has changed the world, & the technologies and spin-offs it has spawned.

# ATTACK OF THE KILLER WEB BROWSER!

## KDE 3

Don't miss our KDE3 disc bonanza, with the source plus dependencies and as many distro-specific packages as we can fit! We'll even tell you what's new and give you a brief guide to the changes.

## PLUS

Cyclades terminal server, Volution, Mac-to-Linux and Mandrake keyboard extensions.

## Introducing: LINUXPRO

Our new mini-magazine keeps you up to date with corporate-scale Linux developments, trends and opinions. This 24-page tome is included with *Linux Format* from next issue, and the first will include a real-world look at mainframe Linux at Warwick University, and an interview with FSFE's Georg Greve.



## 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