

Welcome to our new look!

Well, here it is, the very first of the new-look *Linux Format*. You'll notice quite a few changes throughout the magazine, but don't worry, all your favourite regulars, like the Answers pages, Hot Picks and our extensive news coverage are all still here.

But you'll find that there are a few new things too. One of the major changes we have made is to introduce three new sections in the magazine, covering the Desktop, Developer and, for want of a better word, Professional Linux markets. Each self-contained section has its own news analysis, comment pieces, reviews, tutorials and in future issues, case studies, interviews and more.

This doesn't mean that if you are a developer you can't look in the other sections too – we're just trying to make it easier for all readers to find the information that interests them most.

You will also, no doubt, have noticed that we've changed the look of the magazine. Again this is to make it easier to find your way around,

and I hope you'll agree, it makes features and tutorials easier to read and follow – try taking a look at the new-style Perl tutorial on page 68, or our iptables guide starting on page 82.

Speaking of features, our eight-page special on embedded Linux is definitely worth a bit of study. We've investigated this booming market for Linux development, and you'll find all sorts of info on embedded development of all types, not just the narrow field of PDAs. Find out who the major players in this market are, what tools and choices exist for the embedded developer, and the exciting devices already in the pipeline! Check out the full story on page 40.

Also not to be missed this issue is our look at Freenet, the peer to peer network that vows to escape censorship from any quarter – we answer your questions on page 60.

And with a look at Loki's upcoming *Rune*, the latest beta of Opera and more, your new-look Linux Format is as packed as ever.



Nick Veitch EDITOR

>> Aims of the magazine

Linux Format is a magazine dedicated to Linux and the Open Source community. The aims of this magazine are quite simple:

- >> To promote the use of Linux by providing friendly, easy to follow information.
- >> To help our readers get more out of their Linux experience, through our tutorials, features and advice pages.
- >> To provide Linux users with accurate and unbiased, up to date information.

What on Earth is Freenet? Its the latest in p2p networking. Find out more. p60

LFX have something of a coup this month with a preview of Loki's latest stunner. p50

The major players and the devices in the works – all you need to know. p40



60



50



40

Meet Linux Format's team of writers.....



Richard Drummond
As well as writing our Java series, Rich coordinates most of the reviews in the mag.

Mike Saunders
Another regular name in the mag, Mike has been contributing to *Linux Format* since our very first issue.



David Coulson
An *LXF* regular, David is a sysadmin with VA Linux and OSDN. He's also a bit of a Linux security guru.

Jon Kent
In the hot seat for Hot Picks this month, Jon has been rounding up the open source finest for you.



Andrew Channelle
Our resident newshound, he spends his time alternately chasing news and late contributors.

Keith Pettit
Based in the USA, Keith is a full-time Linux consultant and web guru.



Paul Ravening
Compiling our bumper DVD last issue nearly finished him off, but he still found time to fill a cracking CD this issue.

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



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

Jono Bacon
Founder of Linux UK and big KDE exponent, Jono is studying Multimedia at Wolverhampton Uni

>> Contact us

Letters for publication:
lxf.letters@futurenet.co.uk

Subscriptions/back issues:
subs@futurenet.co.uk

Technical help/Ask the Experts:
lxf.answers@futurenet.co.uk

Disc problems:
lxf.support@futurenet.co.uk

General enquiries:
linuxformat@futurenet.co.uk

Website: www.linuxformat.co.uk

Or send your letters to:

**Linux Format, Future Publishing,
30 Monmouth Street, Bath, BA1 2BW**
Phone: 01225 442244
Fax: 01225 732398

More contact info on page 97

The UK's best selling Linux magazine

LINUXformat

www.linuxformat.co.uk

Welcome to the new look Linux Format, your guide to all things Linux!

COVER FEATURE

EMBEDDED LINUX

Discover the players and technologies that are pushing Linux into the tiniest dimensions.

40

SuSE 7.1 Reviewed

Kernel 2.4.2 and KDE 2.1 are just some of the features on offer in SuSE's latest distro. Find out how it performed in our tests.

22

Emulating A Legend

It was small, had rubbers keys and beeped – would you really want to go back the Spectrum? Simon Goodwin shows you how – just in case.

56

Freenet Explained

Peer-to-peer networking Freenet-style. Is it a tool for anarchists or libertarians? Find out as we ask, What on Earth is Freenet?

60

>> REGULARS

NEWS	6
What's new in Linux around the world.	
LETTERS	12
Your chance to air your views.	
USER GROUPS	16
Locate a LUG near to you.	
ANSWERS	18
Your technical questions answered.	
WHAT ON EARTH	60
The whys and wherefores of Freenet.	
SUBSCRIPTIONS	88
Find out how to never miss another issue.	

>> TUTORIALS

WINMODEMS	54
We explain how to tame your errant Lucent Winmodem under Linux.	
PERL	68
Charlie Stross puts regular expressions to good use filtering junk mail.	
SPEAKING JAVA	73
How to tackle more complex GUIs.	
MASTERING IPTABLES	82
A complete guide to using kernel 2.4's new firewalling features.	

>> REVIEWS

SUSE 7.1	24
Hot on the heels of 7.0, can version 7.1 of this popular distro offer anything extra?	
OPERA	26
The betas keep on coming from Norse browser maker, <i>Opera</i> . We take another look.	
BOOKS	27
This month's top page-turners are O'Reilly's <i>Network Printing</i> and Wrox's <i>Professional Linux Programming</i> .	
ROUNDUP	28
Mike Saunders makes sweet music in this issue's comparison of Linux software.	
HOT PICKS	28
More of the best free software that is available for everybody to download and use.	
RUNE	50
Axe-man Paul Cavanagh takes a sneak preview at Loki's in-progress Linux game conversion.	
NAUTILUS	52
The first stable release of Eazel's next gen desktop system benchtested.	
BLACKADDER	66
Now that's a cunning plan! A visual IDE for Python from the Kompany.	
NETVAULT	80
Nick Veitch investigates this enterprise-level back-up solution from BackBone.	

LINUXformat

In your favourite sections this month...

>> DESKTOP

Everything for the Desktop Linux user

48 NEWS
Who will survive in the increasingly crowded browser market.

50 REVIEWS
Includes a preview of Loki's *Rune* and the first stable release of *Nautilus*.

54 TUTORIALS
Includes how to get your Lucent Winmodem working under Linux.

>> DEVELOPER

Latest hacker tips, tools and techniques.

64 NEWS
Including an exclusive chat with new Debian supremo Ben Collins.

66 REVIEWS
The Kompany's visual Python IDE, *BlackAdder*.

68 TUTORIALS
Includes the latest installments of our series on Perl and Java programming.

>> PROFESSIONAL

Expert advice and info for the Enterprise

78 NEWS
Red Hat seem perilously close to turning a profit.

80 REVIEWS
BackBone's *NetVault*. Is it the last word in back-up solutions?

82 TUTORIALS
Mastering IP Tables, the Linux 2.4 firewalling features explained.

LINUXformat Coverdisc

Over 640 MB of Linux software for you to install and use



Save yourself the trouble of locating and downloading software. We hand-pick and deliver the latest and best Linux applications and tools straight to your PC.

On the CD



Please read the coverdisc instructions starting on page 93 before installing from the CD!

LINUXformat Newsdesk

Space, Spain, the battlefield, your living room. This quartet of disparate places have one thing in common: they're all being invaded by Linux, for better or worse.

Uniting Linux and space exploration: a marriage made in geek heaven?

Reach for the stars

You may not realise, but Linux has already hitched a ride into space as far back as 1997 – when a Debian system was used in experiments to monitor plant growth in microgravity – but with an American recession looming and budget cuts already proposed in their space programme, Linux could be on the verge of going boldly where few OSs have been before.

Leading the campaign to make Linux the default OS for space operations is Flightlinux, a real-time variation on the operating system designed specifically for spacecraft use. The aim, say project leaders, is to 'use the open source concept, with a modified real-time, embeddable kernel' to address the unique problems that space travel presents. The project has even received official blessing and is the subject of a NASA Advanced Information System Technology (AIST) research programme.

Linux has been chosen over proprietary OSs such as pSOS and VxWorks because its open source nature means code can be adjusted to suit specific missions and environments. Flightlinux also cites 'commonality of ground and space-based environments' as a key benefit.

Chasing the station

However, Linux in space is more than just a theoretical proposition, Linux applications are actively being used – or are under development – on the International Space Station in some of its most vital systems.

The Automatic Transfer Vehicle (ATV) is a part of the International Space Station project developed by the European Space Agency (ESA), and is designed to deliver supplies and cargo to the station. The mission brief says: 'The ATV establishes

rendezvous and docking with the ISS in the Russian segment of the station. Rendezvous operations start at about 20km behind the station, moving faster in order to catch up the docking port. A fairly vital job then.

The problems of docking a small spacecraft with a big lumbering space station are many, and involve some extremely complex maths. One solution being tested is RACSI, a Linux-based system which monitors the rendezvous and allows users to temporarily interrupt the mission or execute a collision-avoidance manoeuvre. ESA says that Linux was chosen for this project because it provides four desirable features: reliability, performance, portability, affordability.

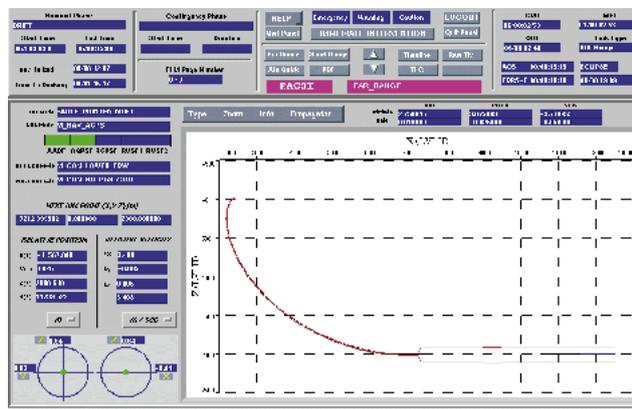
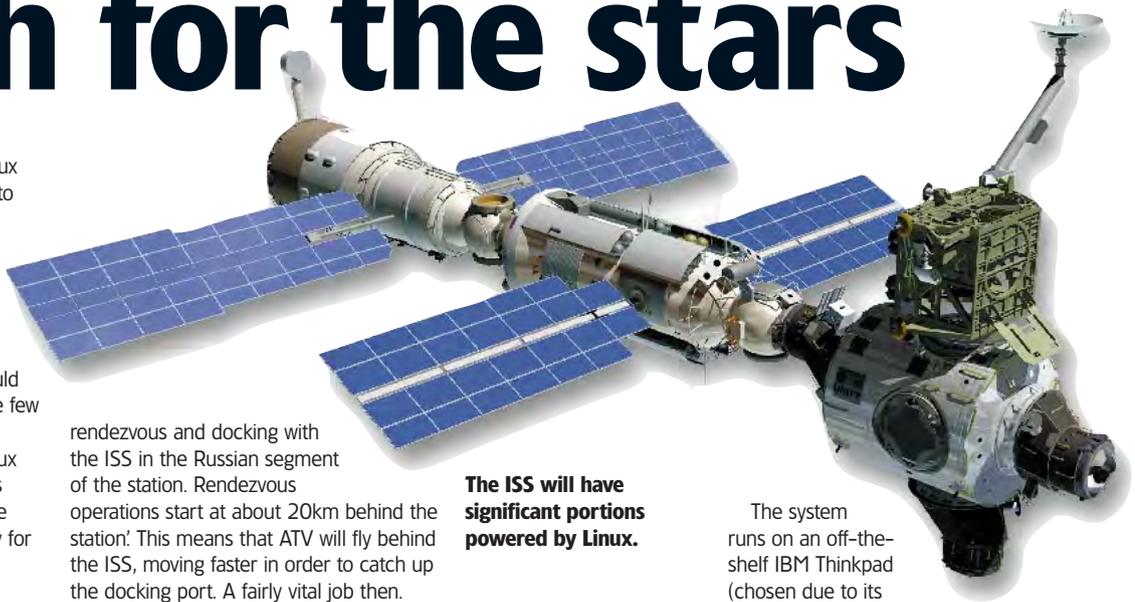
The ISS will have significant portions powered by Linux.

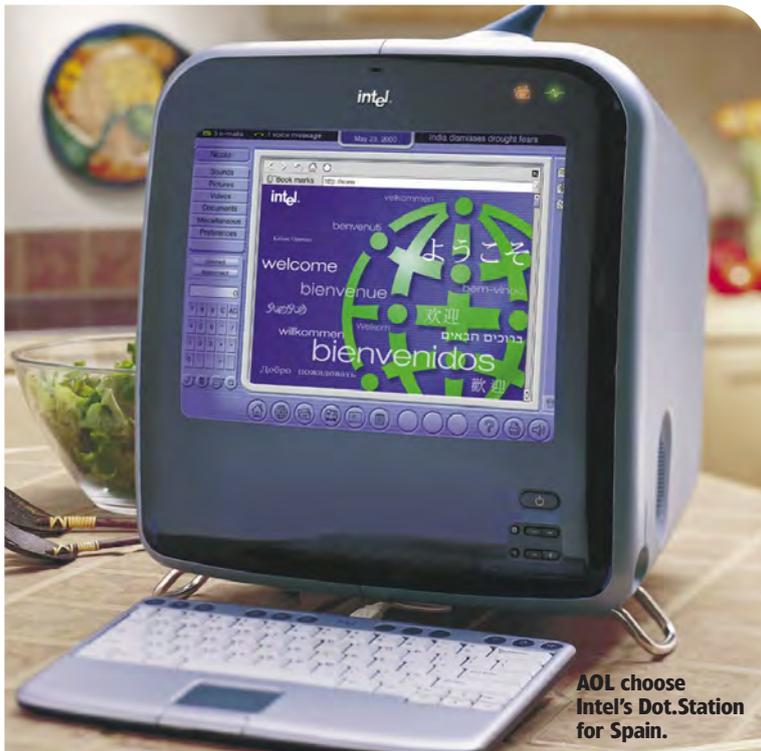
The RACSI system runs on an off-the-shelf IBM Thinkpad.

The system runs on an off-the-shelf IBM Thinkpad (chosen due to its good resistance to radiation) equipped with 64Mb RAM. Presently it is running on Slackware v3.0, using X-windows and the FVWM window manager, but future plans include a full system upgrade to allow users to take advantage of the latest multimedia capabilities. Future features will include the integration of live video and a live grid display which could provide astronaut guidance and navigation.

While those on the ISS have RACSI, their Earth-bound counterparts have the Ground Operator Assistant System (GOAS) to assist them in the docking procedure. GOAS compliments the RACSI system, allowing mission controllers to handle fault detection and recovery and, if needed, it can be used to override onboard systems. Though GOAS was developed on an UltraSPARC machine, a Linux-based edition is up and running on a 233Mhz Pentium system with 48Mb RAM.

<http://flightlinux.gsfc.nasa.gov/>
<http://www.estec.esa.nl/wawwww/ESC/racsi.html>
<http://www.estec.esa.nl/wawwww/ESC/goas.html>





AOL choose Intel's Dot.Station for Spain.

250,000 new Linux users

Linux growth-spurt predicted in Spain

Just as the web-appliance market seems to be going through a slump, the world's biggest chip manufacturer has announced a massive new deal to supply 250,000 Dot.Stations to AOL/Time Warner and Spain's biggest bank. The deal may see Banco Santander Central Hispano giving away the \$500 devices in an attempt to kickstart its online operations. The partnership will also launch AOL Avant, a leisure portal offering ecommerce services to Banco Santander's 10 million clients.

The Dot.Station has been designed to offer email, web access and telephony in a single device and, naturally, runs on Linux – with Mozilla providing Internet services. Claude Leglise, Vice President of the

Intel's architecture group, said the product was designed to meet the needs of users who don't want or need a fully-fledged PC. "The Dot.Station is the result of extensive research and close cooperation with our customers," he said.

Though the product has been available for a number of months, the AOL deal is the first big-name signing and, Intel hopes, will lead to a wider adoption of the technology among ISPs.

The announcement of the deal came a day after a number of high profile players announced they were pulling out of the Internet Appliance market (see Embedded Linux News) due to the expected downturn in the world economy.

www.intel.com

Now online at the Linux Format website.....

Filling a niche

Linux is a great server OS, but will it ever take off on the desktop, and is success on the desktop even necessary? If the figures are to be believed, the future of computing is in small, specialised devices that will record TV, organise your life or tell you when your milk is going off. The embedded sector seems to be growing at a phenomenal rate (Audrey and Kerbango notwithstanding), so should kernel developers concentrate on optimising 2.5/6 for this (as they concentrated on enterprise for 2.4) and just leave the slim pickings of the desktop market to Windows?

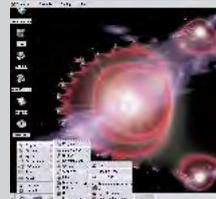
Tell us what you think. www.linuxformat.co.uk

NewsBYTES

After the recent **Ramen** worm scare, the Linux community has been hit with another virus: the Lion worm. Lion is said to be far more malicious than the Ramen worm, and has the power to steal passwords, gain root access to your system and even install tools to allow its creator easy access to your system and data.

If the news that Apple had managed to patent 'themeing' made you laugh, you'll be pleased to know **Microsoft** have apparently applied for, a patent for online polling. Just don't tell any of the sites that have been using it for years.

Caldera have announced the beta release of OpenLinux Workstation 3.1. The product, which is said to reflect the companies strategic move away from the desktop market, is available for download from www.caldera.com



Libranet are taking an innovative step with their latest downloadable distro: they want you to pay for it. Explaining the \$15 download fee, Libranet say: "Linux distribution companies spend large amounts of time and money in putting together their distributions. Unless they get paid they will soon be out of business." See www.libranet.com for more information.

Eazel, the makers of the *Nautilus* file manager, have announced an enhanced software catalogue service which allows users to download and install branded Linux software with a single mouse click.

There has also been bad news on the Eazel front with the company cutting 40 jobs and looking at new ways of funding, including the possibility of joining forces with a bigger company.

After cutting jobs in the US earlier in the year, **SuSE** have confirmed their plans to consolidate their European interests and have also announced a substantial investment in the company by PC giant **Compaq**. SuSE's forecasts suggest the company will achieve a profit within 18 months.

Open (source) warfare

Linux-enabled soldiers

America's Defense Advanced Research Projects Agency (DARPA) have announced plans for an 'infrastructure free' field radio network based on Linux. The agency claim they will be in a position to test their concept live in 2002.

Based on MontaVista's embedded Linux distribution, the project would provide each soldier on the battlefield with a StrongARM-based cell phone. The system would get around the problems associated with cell phone technology – i.e. there are no relay stations on the average battle field – by using a self-configuring low-power radio network to keep soldiers connected. Field tests are currently being designed to discover how robust the system is – whether it will work in all terrains from rainforest to desert – and whether the devices can be shielded from enemy jamming techniques.



Codewarrior for Playstation2

Linux-loving Playstation 2 games developers will soon be able to get their hands on the CodeWarrior authoring system from Metrowerks. The company unveiled an alpha version of the system at a recent developers conference on Santa Clara, and announced they would have a completed version of it available some time around the middle of 2001.

CodeWarrior provides a complete development environment for creating Playstation2 games and the Linux version is expected to cost in the region of \$1000 per user.

This announcement comes hot on the heels of the company's agreement to provide development tools for the Linux-based Indrema console which will become part of its core SDK.

<http://www.metrowerks.com/>



« The patent debate rumbles on

France says “Non!” to software patents

The war against software patents in Europe took another twist when Christian Pierret, France’s Secretary of State for Industry came out against them. “I am against software patents in Europe,” he said. “It would kill innovation and promote juridical terrorism because multinational publishers would multiply legal disputes against startups.”

Pierret is the latest in a long line of individuals and groups to lend their support for the long-running campaign to deny American-style patents on software in the EU.

EuroLinux, an organisation set up to oppose software patents and promote the open source ideal, welcomed the statement, but said there was still more to be done and urged other European governments to take a similar stand.

The whole issue really hit the headlines earlier this year when British Telecom announced they were seeking payments from a number of major US ISPs that had infringed their patent on the hyperlink (see LXF issue 11). However, a number of examples of ‘prior art’ have been cited in the case which would appear to invalidate BT’s case.

Activists say that patents on software would restrict innovation in the industry and would “play against the development of open source software, independent software makers and independent ecommerce in Europe.” However, others argue that developers need a rigorous patent system to protect their intellectual property rights.

EuroLinux have been running an online petition on the subject and have so far recorded more than 73,000 individual signatures.

www.eurolinux.org
www.freepatents.org

But Britain says yes and no...

In a separate development, the UK Patent Office announced the results of its examination of the whole issue and concluded that the current uncertainty about software patents is damaging to both users and the industry as a whole.

The consultation attracted the attention of a diverse range of interested parties including businesses, computer users and individual developers. One thing that was constant was the lack of consensus among respondents. Individual programmers and open source advocates called for fewer or no patents, whereas the majority of corporate interests favoured a more stringent system.

The report said: ‘The Government’s conclusion is thus to reaffirm the principle that patents are for technological innovations. Software should not be patentable where there is no technological innovation, and technological innovations should not cease to be patentable merely because the innovation lies in software.’

Patricia Hewitt, the E-Minister at the Department of Trade and Industry, said: “Our key principle is that patents should be for technical innovations. So a program for a new machine tool should be patentable but a non-technical innovation, such as grammar-checking software for a word-processor, should not be.” She added

that most of the people who responded to the survey opposed patents for ways of doing business on the Internet.

Consumer Affairs Minister Kim Howells said: “The consultation showed that at present there is confusion, and that that is damaging. The European Commission is currently evaluating its own consultation on software patents, and we shall be pressing them for an early Directive which embodies our conclusions.”

The full text of the report can be found at the Patent Office website.

www.patent.gov.uk/about/consultations/conclusions.htm
www.open.gov.uk



E-Minister Patricia Hewitt spelt out the UK’s position on patents for software.

Competition for The GIMP?

Paul Nolan Software have announced the release of *Photogenics 5.0* for both Linux and Windows. *Photogenics* is a graphics package that, unlike applications such as *Photoshop* and *GIMP*, has been designed primarily with image creation – rather than manipulation – in mind.

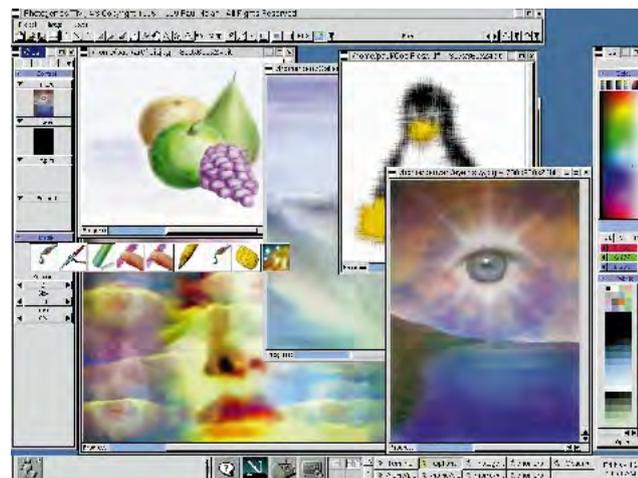
As well as a comprehensive range of realistic media tools such as chalks, crayons and pencils, *Photogenics* also features the unique Paint Layer Technology which allows mistakes to be selectively rubbed-out with the right mouse button. Artistic types can get their hands of ‘paint on image processing’, allowing them to apply any one of the 80 supplied effects – or Paint-on

Pyrotechnics such as flames and lens flares – simply by painting on the screen with any of the media tools.

Completing the feature-set is a range of web design tools, including HTML colour display and an image tuner to give users complete control over compression and image quality, and SANE/TWAIN support for image acquisition.

Photogenics, even in the hands of a novice appears to be capable of producing some fairly stunning original imagery, as well as being a competent processing tool. But is it worth \$99? We’ll have a comprehensive look at the full release next issue.

www.PaulNolan.com



Psychedelic art the Photogenics way.

Is TiVo keeping tabs on you?

Watching the watchers

The makers of TiVo, the Linux-based personal TV appliance, have come under fire for collecting massive amounts of data on their customers.

The Privacy Foundation intercepted messages sent from the TiVo device and discovered that the service “gathers enough information to track individual users’ home viewing habits.” The Foundation said this was happening despite claims in the company’s privacy statement that ‘all of your personal viewing information remains on your PTV receiver in your home!’ The Foundation also castigated TiVo for shipping units with out-of-date privacy information in the manuals, which it said misleads customers about what data is collected and held.

TiVo responded to the allegations by claiming that, though data was collected from the devices, they have taken steps to ensure that ‘anonymous viewing files’ couldn’t be correlated with the diagnostic

information that could identify individual users. They also said there were no plans, now or in the future, to offer information about viewing habits for advertising or marketing purposes, and that though



printed documentation was out of date, all affected customers had been contacted about the new terms and conditions.

After examining discrepancies between TiVo’s actions and its privacy policy, the Privacy Foundation concluded: ‘TiVo would be wise to consider its potential legal exposure for breach of contract, deceptive trade practices, invasion of privacy, and other legal theories.’

www.privacyfoundation.org
www.tivo.com

GNOME sweet GNOME

Hot on the heels of KDE’s recent series of releases, GNOME developers have revealed the latest stable version of their acclaimed desktop environment.

GNOME 1.4 just about made its April 2nd deadline, after a small bug discovered in Nautilus 1.0 threatened to scupper the release.

Havoc Pennington, Chair of the GNOME Foundation, said the release represents a major leap forward in the GNOME desktop environment. “GNOME 1.4 has a wealth of new features, from an updated Sawfish window manager to enhanced support and interoperability with other desktop environments.”

1.4 is the first version of the desktop to include the much touted integration of the *Nautilus* file manager and web browser from developers Eazel.

Other key features include:

- Better interoperability with KDE and legacy X applications.
- New Application Launch Feedback lets the user know when a program is in the process of being loaded.
- Easier to use and more intuitive help browser and help system, boasting extensive documentation covering all aspects of GNOME operations.

www.gnome.org



The GNOME 1.4 desktop, now available with added Nautilus.

LinuxWebWatch



<http://www.linuxnewbie.com>



<http://www.linux.org>



<http://gimp-savvy.com>



<http://www.htmlprimer.com/>

Oh, it’s all so complicated!

But there are places to go where your questions will not be met without a steely glare and a condescending answer.

<http://www.linuxnewbie.com> should be the first port of call for anyone setting out on the road to OS heaven. It has a selection of useful – and easy to follow – advice on the best distros for inexperienced users, helpful tips on getting the most from your software and hardware, and links to hundreds of other relevant Linux resources. Recent highlights on the site include a comparison of various distributions and their suitability for new hackers, a

preview of the latest Debian release and a tutorial on accessing disks. If you outgrow Linuxnewbie, <http://www.linux.org> has a very useable Linux 101 section which has comprehensive introductions to many areas of the Linux/open source community. It’s aimed at users who may not have a vast experience with Linux but are pretty savvy when it comes to computers in general. It also has a comprehensive links section.

The GIMP is one of the best known open source programs, and its reputation is well deserved.

<http://gimp-savvy.com> is an online companion to *Grokking The Gimp*, the rather fine artistic primer by Carey Bunks. Here you’ll find genuinely useful tutorials and resources for all your artistic Linux needs.

Though not Linux-specific <http://www.htmlprimer.com/> is an excellent introduction to web authoring

and should help you have your homepage up and running in next to no time. The site is divided into easy to handle lessons starting with basic page construction. There is also an introduction to more sophisticated ideas such as frames and javascript. And if you’re really stuck, don’t forget to check out the forum at www.linuxformat.co.uk where you can get answers to your questions from LXF’s highly intelligent readers.



« Embedded Linux News

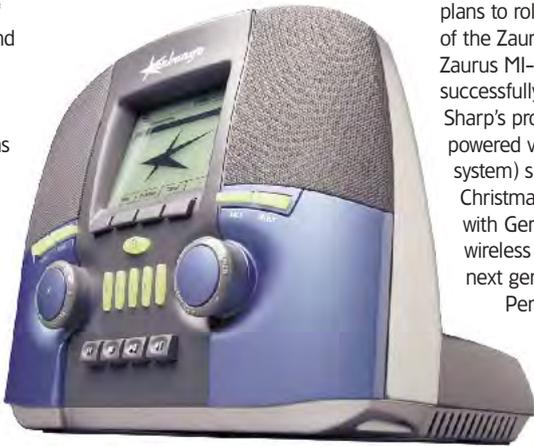
Adios Audrey, farewell Kerbango

3Com have announced the demise of its fledgling Internet Appliance Audrey and the Internet radio Kerbango leading to fears of a big slump in the embedded Linux market.

Audrey, based on the QNX system, was announced back in October as the first product in the Ergo range of IAs, while Kerbango was acquired for \$80 million in a blaze of publicity a few months earlier. The announcement may also see the closure of 3Com's whole Internet Appliance division.

Analysts are speculating that 3Com's decision - when taken with Gateway's rethink of the IA market - suggests that users may not be ready for simplified surfing devices, especially as a standard PC does the job for just a few dollars more.

Korean electrical giant LG announced their own Linux-based Internet access device at Germany's CeBIT computer fair. The Digital iPad features a 200MHz StrongARM processor, 8.4 inch touch-



3Com paid \$80 million for the now abandoned Kerbango Linux radio.

sensitive screen and built-in multimedia features such as MP3 playback and a word processor. The device should be available toward the end of the year.

Sharp used the show to announce its

plans to roll out European and US versions of the Zaurus PDA running Linux. The Zaurus MI-E1 has already been successfully released in Japan using Sharp's proprietary OS, but the Linux-powered version (using the Amiga Tao system) should be available before Christmas. Sharp also announced a deal with Germany's PacketVideo to provide wireless multimedia capabilities to its next generation of handheld devices.

PenguinRadio have announced a deal with Live365 which should see 29,000 Internet radio stations available on the company's Linux-based audio device. Penguin's streaming audio player is expected to go on sale for about \$200. But have they learned anything from the Kerbango story?

IBM have demonstrated a prototype of their second generation Linux wristwatch. It's very Dick Tracy, but with a battery life of just two hours, a little impractical.

OCR for the Linux masses

ScanSoft - the developer of the ubiquitous (in the Windows world) Omni-Page Optical Character Recognition (OCR) software - have joined forces with Vividata to provide Linux and Unix customers with the latest OCR technology.

Mark Liebman, President of Vividata, said the agreement would see high quality OCR coming to Linux for the first time. "There is a large demand for this technology in markets where these operating systems are prevalent, including federal and local government and international sectors."



ScanSoft's OCR engine, which provides full support for 54 languages, features in the latest version of Vividata's *OCR Shop* product which is available for a number of Unix and Linux flavours.

Vividata have also unveiled a new pricing policy for its range of products, offering discounts for personal and 'not-for-profit' users.

www.vividata.com
www.scansoft.com

Cluster management from VA

VA Linux systems have announced the availability of a powerful new remote management solution for server farms.

The VA Linux 100, which is based around the company's respected *VACM* cluster management software, will turn racks of VA Linux 1120 and 1220 1U servers into a unified server array, enabling administrators to manage the whole system from anywhere in the world.

Each VA Linux 100 can handle up to 42 server nodes and multiple devices

can be reigned in under a single *VACM* interface. They also feature server power management capabilities and a built-in

LCD interface for access to system information.

Ali Jenab, President and Chief Operating Officer of VA Linux, said the system was designed to manage thousands of servers distributed among data centres: "Scalable management infrastructure is critical for our enterprise-level customers."

www.valinux.com



Ali Jenab, COO of VA Linux.

Happy Printing from HP

Hewlett-Packard have announced the release of Linux drivers for over 30 of their most popular inkjet printers. The drivers are the result of the HP Linux Driver Project, which is an add-on to the GNU Ghostscript application, and will be provided under a variety of licenses.

The project overview claims the driver should provide basic printing support - i.e. very good print quality on either plain or photo paper - for most of HP's Deskjet printer models.

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

Supported printers

- e-Series
- DeskJet 600C Series
- DeskJet 600C Series Photo
- DeskJet 630C Series
- DeskJet 800C Series
- DeskJet 900C Series

Check the website for individual model numbers.

NewsBYTES

AMD have upped the stakes in their battle with Intel by releasing a 1.3GHz chip. The Hammer chip is a significant move by AMD as it will be incompatible with Intel IA-64 chip, meaning software makers will be forced to develop applications specific to either, or both, chips.



FreeBSD is to become an officially supported platform for the GNOME desktop in all new releases.

Results of a survey carried out by **TNS EMNID**, suggest 56 percent of PC users had heard of Linux and 23 percent have considered switching when upgrading their hardware. The survey also suggests that Linux is now second only to Microsoft Windows in the popularity stakes.

Andamooka, the online library, have added another quartet of books to their database. The latest additions are: *GNU Autoconf*, *Automake*, and *Libtool* by Gary V. Vaughan, Ben Elliston, Tom Tromeo, Ben Elliston, Tom Tromeo and Ian Lance Taylor; *The CVS Book* by Karl Fogel; *Lessons In Electric Circuits* by Tony Kuphaldt;

Professional Programmer's Guide to Fortran 77 by Clive Page. Andamooka have also improved their search engine to make the library even easier to use. See www.andamooka.org.



65 key kernel hackers gathered at the end of March in San Jose to discuss ideas and proposals for **kernel 2.5**. Proposals from the meeting included support for hotplugging, better integration of high-performance filesystems and improvements in virtual memory management.

Lineo have announced the availability of their new embedded Linux SDK. The company said: "Lineo Embedix SDK 2.0 levels the playing field between Linux and proprietary embedded operating system providers."



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: **linuxformat@futurenet.co.uk**

Driven inSANE

As a newbie I have read your tutorials with interest, but I have yet to see a discussion regarding the availability of Linux software for scanners. I am aware of SANE but this does not seem as yet to have developed support for many scanners. Most distros seem to avoid reference to the subject in their manuals.

Would it be possible for you to review this subject in a future edition of *Linux Format*?

John Craske, via email

» We're still working on a comprehensive SANE tutorial. The main problem with scanner support, as with other types of hardware, is the reluctance of scanner manufacturers to cooperate when it comes to supplying the critical information required to create a working driver. The results can be seen with a quick look at the list of SANE supported scanners. Epson did release info to developers, and therefore many of their scanners are supported – Agfa have been less forthcoming with the expected results.

This can be a bit frustrating if you already own a scanner, but if you are buying one, wouldn't you want to buy a model from a manufacturer who actually recognises the existence of Linux?

I suspect that another reason for the overall lack of support in scanners is the paltry amount of desktop DTP software for Linux in general. Whereas you can choose from a dozen or so financial packages, how many DTP packages can you think of?

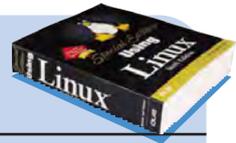
Back to basics?

Just about a year ago I knew little about Linux, just an occasional mention about free programs which might one day compete with Microsoft products. At that time I was having frequent problems with Windows and was informed that Linux was now a viable alternative and as easy to use as Windows.

By chance I saw your first issue of *LXF* and decided to give it a try with the cover disc. I had no problems installing it but nothing worked as expected, I could not get connected to the Internet, read disk drives, use my scanner or camera and spent endless hours trying to fix this. I became interested, so subscribed to *LXF*.

★ Letter of the month

This month's winner receives a copy of Que's **Using Linux**



Linux in Tibet

My wife and I have just come back from three weeks in Dharamsala, in North India, where the Tibetan government-in-exile is based. Since we are a Linux/UNIX consultancy company (Gatekeeper Technology Ltd), we gave a lot of our time there teaching free classes on Linux.

Linux is a good technology for them. They use a lot of Windows right now, but they can't afford any licenses and indeed can hardly afford to keep up in the hardware arms race that new versions of Windows involve. They had heard of Linux and were very interested in it, but didn't know where to start learning about it. They even tried to find commercial classes in Delhi to teach them, but there aren't any.

We got them started, built them some servers and desktops and now they can build their own. They have now chosen Linux as their Internet-gateway-of-choice for all the schools in the Tibetan Children's Villages school system, which has 14,000 pupils. We are hosting a Tibetan LUG mailing list on our mail server. They're really going great guns for Linux, now they've got a start.

What they need is continuing information. I've been a subscriber to *Linux Format* for some months now, and a reader before that. I like it. It has up-to-date coverage of what's happening in

the community. The cover CD would be particularly useful to them, because all Internet access in that part of the world is through dial-up modems, which rarely get beyond about 24Kb/s, so large downloads aren't practical.

Do you think there's any chance you would be prepared to send them, by airmail, a complementary copy of *Linux Format* for two to three years? I can guarantee it would be passed from hand to hand, read to destruction, and the cover CD mounted on more systems that you can shake an inode table at.

I do hope you feel that this is something that you can help with – it would make a real difference! Thanks for your time.

Tom Yates, Cambridge

» Many thanks for your letter. I hope you had a good time while you were there too. We'll certainly sort out some issues of the magazine for Dharamsala, and we'll send along some goodies from the *LXF* office too.

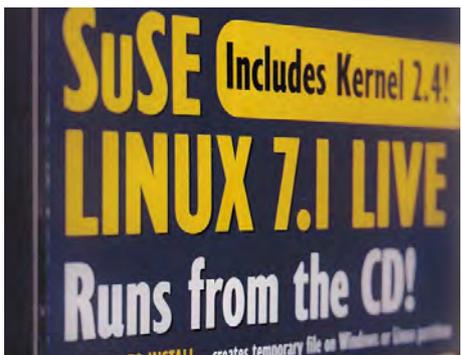
This is the first issue we've featured a Star Letter, where the best letter wins a prize. This month we're giving away a brand new copy of *Using Linux* (Published by Que, ISBN 0-7897-2543-6, price £32.99), which I guess we should also post off to India!

After reading your review of Mandrake and how easy it was to install, I tried it from the cover disc with much the same result. I am well aware that the response to my difficulties might be 'read the ***** manual', but it should be remembered that I knew nothing of Linux or Unix, I just wanted to get a working system and as your magazine is clearly directed at those with programming experience it was not a great help. Where your articles are written for newbies they are often not complete; missing out vital steps which are obvious to your experienced writers but serious obstacles to a beginner.

I identified some of my problems as BT ISDN card, a sound card not supported, and no support

for my USB Zip drive, scanner and camera. Recently SuSE 7.1 with Linux 2.4 became available claiming to solve all my problems so I bought the Personal copy expecting the books would give advice on installation. BT Home Highway uses a German card so it had no trouble getting me online. But none of the other devices worked. A detailed study of the supplied literature did not give instructions a beginner could use to get a working box.

I know that all the information needed is probably available on-line and in various help files, but to locate the bits needed is a mammoth task when starting from scratch. The point I am trying to get across is that this is of little use to a beginner



Is Linux getting easier to install?

trying to get started. If Linux is ever going to be acceptable to the average Windows user then either the installation of all devices should be automatic, as in Windows, or detailed step by step instruction provided with the distro.

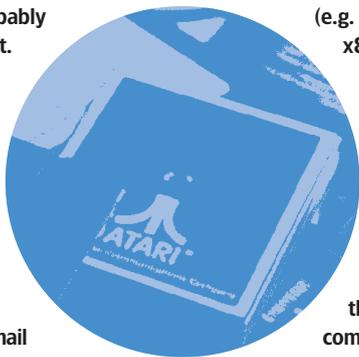
I have found the study of Linux so far interesting but hard going, a steep learning curve, which provides a lot of stimulation. But I think it is misleading to suggest as your reviews and distro adverts do that Linux is easy to get started, it has not been my experience. May I suggest as an experiment that you find someone with no knowledge of Linux but a user of Windows and ask them to install SuSE 7.1 or any other distro and note where they have problems. This would give you an insight of the problems faced. I use a separate HDD for Linux so avoid all the difficulties of partitioning.

I have seen other letters in *LXF* complaining of the lack of basic help for beginners and your explanation that it would take up space better devoted to competent users. I would suggest that this is a mistaken view and that if more effort were made to attract newbies your sales would improve dramatically.

John Hartley, via email

» Thanks for your comments. I don't think we ever said that we didn't think it was worth the space to explain things to beginners, but it is true that there is a finite amount of space in the magazine. One of the problems is that people take to Linux differently – a lot of people seems to want the ways of Linux translated into Windows-speak, but that's probably the wrong way to learn about it.

We do try and give as much help as possible, but it doesn't hurt people too much to use a `/man/` command now and then to find out how to do something. We include instructions on compiling software in almost every issue, and still people occasionally email us saying "how am I supposed to install xxxx from the CD?" We'll keep on trying to help. If there are specific things you have problems



understanding or following, then please let us know – it'll help us make things clearer for you in the future.

What about Ataris then?

I've noticed in your magazine that many Amiga owning readers are concerned about the lack of support for their systems when it comes to Linux, and fair enough, the support for Amiga's when it comes down to Linux is very poor compared to PCs.

How often does this apply to Atari users then?

My Atari runs Linux Debian (Potato) better and faster, than Mandrake 7.2 runs on the AMD – by quite a long way actually, but is there any support for me, and the rest of us? It took me days to actually get it installed, but when it finally did go, I was able to set everything up, even the graphics card – which it clearly states, is not supported! (Yes, Atari's do have graphics cards – and soundcards, and all new ones even ISA & PCI Slots by the way!), it even lets me access both SCSI Ports and the IDE line!

While I am willing to argue that Linux and *GNOME* is not as powerful, as *Magic* and *Jinnee*, I still feel that Linux is the way to go. And the only fair way is for there to be a little more support of the lesser-known systems out there, because there are obviously far more uncatered-for users than you think.

Damion Jones, via email

» The great majority of Linux users run x86 architecture, so it's no surprise that that's where all the attention is focused from the distro side of things, with notable exceptions like Debian. As the Amiga and Atari platforms saw the first ports of Linux, it's no surprise that these platforms are actually well supported by the kernel.

The only time we deal with specific installation instructions is when we have a big distro on the CD (which tend to be x86 only). The rest of the magazine is equally as applicable to all Linux users, with a few exceptions (e.g. commercial packages which are x86 only).

If we did install guides for every architecture that supports Linux, it would fill the whole magazine and then some. But what specific support do you want? It may be possible to do install guides on CD, but I'm not sure they could ever be more complete than the Debian documentation that already exists (and runs to thousands and thousands of words for Atari). If you could let us know specifically what

you think is problematic in installing Linux on an Atari system, maybe we could help add to the Debian documentation.

And, for balance...

I've been subscribing to *LXF* since the demise of *Amiga Format* (now an *AmigActive* reader) and currently use AmigaOS3.9. I am also interested in Linux, but Linux seems to be facing problems that were solved on Amiga OS 15 years ago. For Linux to be accepted as the next home computer desktop operating system, these factors have to be considered.

- Ease of use but at the same time being modular.
- Simple installation of software.
- One standard, so that all software works on every distro.
- Fast Software downloads.

You say there's more activity in the Linux world, but I would argue there's still life in the Amiga yet, especially with great games like *Payback* still being released and running on low spec Amigas. PCI

“My Atari runs Debian better and faster than Mandrake 7.2 on the AMD – by a long way actually, but is there any support for me and the rest of us?”

expansion cards & G4 CPUs will also extend the life of the system.

Maybe Linux users should take a look at UAE and see that things are still happening in the Amiga scene. Then maybe even open their minds get a real Amiga and support a RISC PPC distro instead of the old CISC x86 technology.

Mark Dale, via email

» I think there isn't too much debate about some of the points you mention. Everyone agrees that software should be easier to install, which is sort of dependent on point three, that all distros conform to certain rules. There are some ways in which distros comply to a standard in terms of where particular things live, but there are many more that can cause problems. It's a tricky thing to sort out without some sort of combined effort from all the distro providers, something that seems difficult to achieve.

Die GNOME, die, die!

So when are we finally going to become creative? What I'm trying to say is that it's about time that we move with the flow and not be handcuffed by a monopoly. When will we forget about Windows? For this reason I would like to scrap all the *KDE* and *GNOME* interfaces (no hard feelings guys ;-)) since they look so much like Windows with all these icons and start buttons. There are great alternatives such



« as *Window Maker*. It's different, original and new. So as a constant reader of *Linux Format*, I would humbly like to ask of you if possible to give WM users more support by your articles.

George Vartabedian, *Beirut, Lebanon*

» Blimey. First of all we aren't supporting various types of hardware, now we're not supporting different window managers. I'm not sure I really get this "not supported" business. Apart from in a tutorial on, e.g. developing for KDE, in what parts of the magazine is *Window Maker* not supported? We have run articles on configuring KDE, and we've done the same for *Window Maker*. Is there something else we should be doing?

“As a professional software developer, I know you need the right tool for the job, and Linux isn't always the right tool – except in the case of CD-ROMs!”

Please people, if you are going to write in suggesting that such and such is “not supported” in the magazine, can you please be more specific about what you would like to see?

Too much desktop

Having bought your last two magazines – and intending to continue buying them into the future – I noticed that the majority of articles are for desktop users. While this is helpful, I would also like to see more stuff about server setups, and also more software on the cover disc for servers. I guess a lot of readers are desktop users, as I am. But I also enjoy tinkering with web servers etc. And the work that goes with it.

As for my current setup, I am currently using a MuLinux – a mini Linux installation – which is very fast and fits into a few megabytes on the family PC with little danger of crashing it. However, so far none of the software on the cover discs has worked, except *cbb* which installed and looks nice but doesn't do anything. I am hoping to get another machine which I will dedicate purely to Linux, and network it to my two 486's to get a pretty sophisticated firewall and server set up.

Richard Kirkcaldy, *via email*

» Well, we do actually have quite a few network-based reviews and features in the magazine. As I hope you will have noticed, from this issue we have rearranged the magazine to make it easier for those with particular interests to locate reviews and tutorials that will be of interest to them. The Professional section will include networking and server related material, but you should also check out the

Desktop and Developer sections for features and reviews of interest.

I think your coverdisc problems are probably related to missing support libraries. MuLinux is very neat, but doesn't include a lot of the libraries that some software takes for granted. We do try to include libraries you might need on the CD, but to provide everything would mean we were putting the best part of a whole distro on each time!

Find your language

I'm a Linux newbie and was just after some advice. I'd like to learn a programming language as a hobby but I'm not sure which one to learn. Can you guys help me out? Which is an easy one to learn, what

programs are available to develop with and can you recommend any good books.

Any help you give will be appreciated.

Abz Hussain, *via email*

» There are dozens and dozens of current programming

languages (would anyone be too miffed if we dismissed Algol, Comal etc as being a bit past it?), and which is best depends on your ability and the sort of programs you want to write.

Probably the most generally useful language to learn in conjunction with Linux is C or C++. That's simply because most software for Linux – including the kernel – is written in C.

Some people say that C is not particularly ideal for beginners, but I personally disagree. It is more of a leap, in my opinion, to go from something like Basic to C, than to start off with C in the first place!

However, if you are more interested in web/CGI or script programming, then you should be looking at Perl and Python, and possibly PHP for web stuff. In many ways Python is the easiest to learn here, but Perl is probably still more widespread. PHP is fairly ubiquitous on the Internet now.

As for development tools, again there are

loads. *Glade*, *KDE Studio*, *Kdevelop* and more are available for C programming. *BlackAdder* is an IDE for Python and there are plenty more.

If you are interested in programming for Linux in particular, a good book to get started with is *Linux Programming in 24 Hours* (Warren W. Gay, Sams Publishing, ISBN 1-57595-352-8). It's a good intro to the concepts of programming for Linux, though you will probably want a general programming book for C or whatever as well. For learning C, *The Absolute Beginners Guide to C* (Greg Perry, Sams Publishing ISBN 0-67230-510-0) is good, and the Wrok Press *Beginners Guide to C++* (Oleg Yaroshenko, ISBN 1-87441-626-5) and O'Reilly's *Practical C Programming* (Steve Oualline, ISBN 1-56592-306-5) are also pretty good.

Scorn in the USA

As a professional software developer, I know you need the right tool for the job, and Linux isn't always the right tool (a point you seem to make to Matt Oakley in a reply in the February 2001 issue).

However, upon mounting my *Linux Format* companion CD (for the February 2001 issue) I noticed all the file names were truncated with “~1” or “~2” etc. Now I don't know all the details, but that usually means that a DOS filesystem is being used (but maybe it has something to do with UK vs. US?), and indeed when viewing the CD through Windows all filenames are just fine.

I certainly don't think a Linux magazine is under any obligation to be anti-Windows (I do believe in choice, and Windows is one of the choices). But how would a Windows user feel if their Windows magazine companion CD came in ext2 format (besides the obvious answer of, “they wouldn't even be able to read it!”)? I know since Linux is so friendly to foreign formats you might think it's OK to do this, but since Linux fails to seamlessly read DOS formats, it could be regarded as a slap in the face to Linux users. It would be far better to master your CD in a 100% Linux friendly manner, because in this case Linux is the right tool for the job!

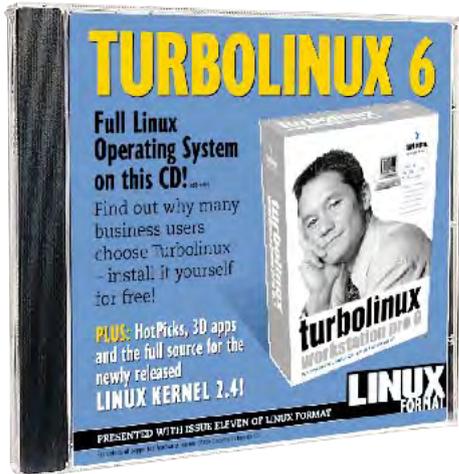
Jason Mott, *New York*

» Actually, the 8.3 filename convention is also a limitation of the ISO9660 level 1 format, which, as I'm sure you know, is the standard data CD-ROMs adhere to. Long filenames are supported by extensions such as Joliet for Windows, or RockRidge in the case of Linux/Unix systems.

Mostly our CDs are created with Joliet and RockRidge support, so they can be read on many systems. There should be no disadvantage to Linux users in including Joliet support (and the Linux kernel itself supports Joliet extensions too, which are normally enabled in the kernels built by various distros).

I'm not sure of the exact master settings on this CD (it reads fine in the collection of Linux





The LXF CD: made for Linux users!

boxes we have here) but I assure you, we do create them with Linux in mind!

Response To Prof Lamb

I really do feel that Joe Lamb (Mailserver, LXF13), and others who make comparisons between Windows and Linux in your letters pages, are missing the point entirely.

I agree that many people use their operating system for word processing, graphics, games, etc. and to those people I say 'If it ain't broke, don't fix it - stick with Windows.' However, using a computer is like owning a car - you use your car to get to and

“I really do wish that Windows users would relax a bit! The majority of the Linux user community is not mocking them for staying with Microsoft”

from work, to go shopping in or to take you to the airport when you go on holiday. But you cannot own a car without knowing how to put fuel in it, how to change the tyres and how to top up the oil - anything more serious than that goes wrong with it then you will either dig out a servicing manual and have a go yourself or take it to a specialist who can fix it.

Operating systems, be they Windows or Linux, are no different. Microsoft would have you believe in a utopian world where you take your nice new Windows CD out of the shrink wrapping, put it in your PC and 20 minutes later, you have a nice new, fast operating system to use. Sometimes that is the case but other times, as we all know, you end up with a trashed system or a system that runs like a dog. What I am trying to say here is that you have a choice - either you use your PC purely as a tool for word processing, gaming etc and leave it to an available expert to fix or you do the job yourself.

In the Microsoft world, most elements of their operating systems are proprietary so if something goes wrong and you cannot fix it immediately yourself, you can try and learn a bit more and go deeper or put yourself in the hands of an expert or prostrate yourself at the feet of Bill Gates and hope for a bug-fix.

Linux is entirely different. It assumes that if you are going to use it, then you have some willingness to have a go yourself and to put in some time and effort to learning a bit more about how the PC and the operating system work. It also assumes that you have a reason to use it and that you are not simply a trend follower - before you deploy any Linux installation, you should be very clear exactly what uses you intend to put it to so that you can go and do some initial research and find out how well Linux meets your needs.

Joe Lamb and others need to recognise that on any Linux system, much as distributions and installers get more and more automated, just as with Windows, you are never going to get a system that entirely meets all your needs out of the box and since Linux follows a UNIX structure, invariably that means dropping to a command line on some occasions to do some work on your system.

I really do wish that Windows users would relax a bit! The majority of the Linux user community is not, contrary to popular belief, mocking them for staying with Microsoft because the whole point of Linux is that you have the ability to customise your operating system environment to the way you want it to work

- if Windows fulfils that then stick with it!

Linux is about having a choice and Linux users are people who do not feel that Microsoft offers them enough choice.

Peter Andrijczko,
via email

» I guess the point is

that various aspects of Linux could be a lot easier, but like you say, it's a question of what you are willing to learn. I would also hazard to suggest that when Windows screws up, people seem to accept it, but if they have a problem with Linux, they somehow blame the OS...

Ease of Use?

Reading your statement in a recent issue that "KDE has pretty good documentation available" left me totally incredulous: is that really what you meant? I have found KDE documentation awful.

For a start, the search function returns multiple duplicates and only gives the page heading with no context. There are numerous pages with no information at all and repeated apologies for brevity.

When you actually find documentation it is largely concerned with the menus, which are usually self-explanatory, and ignores how you actually use

the program and the way it works.

Look at some examples: my desktop keeps losing items. This has wasted weeks of time tracking down how the desktop is setup. I found nothing in the KDE help and little in my 800-page manual I am still only half-way to a fix.

Again I want to edit the KDE configuration files in Autostart. I have failed to find any indication of the valid values which can be used. For example I can specify the location of X program windows with -geometry, but KDE takes no notice.

Can anything be done? I have noticed that the most useful part of other documentation is the 'troubleshooting' section. How about an 'Error of the Month' article?

Cecil Wallis, via email

» Well, we would still stick by the statement that the KDE project is well documented. Fantastically well documented compared to many other examples of Linux software. Some of the individual applications which run under KDE are not so well documented, and as this is the responsibility of the individual authors, you can't blame the core KDE team.

Many of the core KDE applications also support the geometry switch. Briefly the solution to your problem is to create a template (see KDE documentation) by right clicking on the desktop and choosing Create New>link to application. Click on the execute tag and type in the command and options you want, e.g. `konqueror -geometry 300x250+20+20`. Then you can copy the link to your Autostart folder if you want. All of this information can be obtained for the KDE documentation, or by running the associated command with the -help switch. LXF

Mailserver Hot Topics

We have introduced Mailserver Hot Topics to help gauge your opinions on the things that matter most. Please feel free to continue writing in on any subject you like (except the glues we use on the CDs, everyone's heard enough about that), but we would be extra keen to hear your views on the hot topic of the moment. Without further ado, next month's topic is: **Viruses and Linux - are we afraid or are we prepared?**

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: linuxformat@futurenet.co.uk

User Groups

Wherever you are in the world, there'll be a Linux User Group somewhere near you. There are hundreds of dedicated User Groups all over the UK alone, so find the one nearest to you now! And if you're outside the UK, you needn't feel left out either, there are thousands of User Groups worldwide, 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.linuxformat.co.uk/links.php or at www.lug.org.co.uk

1 Hampshire

URL www.hants.lug.org.uk
Contact Ken Adams

2 Bristol & Bath

URL www.bristol.lug.org.uk
Contact Dave D

3 Scottish

URL www.scottish.lug.org.uk
Contact Tony Dyer

4 Oxford

URL www.oxford.lug.org.uk
Contact Alasdair G Kergon

5 Bromcom (Kent)

URL www.kent.lug.org.uk
Contact John Mills

REVISED
DETAILS

6 Brighton

URL www.brighton.lug.org.uk
Contact Johnathan Swan

7 Sussex

URL www.sussex.lug.org.uk
Contact Mike Pedley

8 Northants

URL www.northants.lug.org.uk
Contact Kevin Taylor

9 Anglian

URL www.anglian.lug.org.uk
Contact Martyn Drake

10 Milton Keynes

URL www.mk.lug.org.uk
Contact Denny De La Haye

11 Doncaster

URL www.doncasterlug.org.uk
Contact Andy Smith

12 South East

URL www.southeast.lug.org.uk
Contact Ian Reason

13 West Wales

URL www.west-wales.lug.org.uk
Contact Ken Adams

14 Wolves

URL www.wolves.lug.org.uk
Contact Jono Bacon

15 Peterborough

URL www.peterboro.lug.org.uk
Contact Steve Gallagher

16 Edinburgh

URL www.edinburgh.lug.org.uk
Contact Alistair Murray

17 Tyneside

URL www.tyneside.lug.org.uk
Contact Brian Ronald

18 Leicester

URL www.leicester.lug.org.uk
Contact Clive Jones

19 Greater London

URL glug.linux.co.uk

20 Surrey

URL www.surrey.lug.org.uk
Contact James Chivers

21 Cambridge

URL www.cam-lug.org

22 Devon & Cornwall

URL www.lug.termisoc.org
Contact Alex Charrett

23 Essex

URL www.epos.demon.co.uk

24 Manchester

URL www.manlug.mcc.ac.uk
Contact Ted Harding

25 West Yorkshire

URL www.scs.leeds.ac.uk/wylug

26 West Yorkshire

URL www.wylug.lug.org.uk
Contact Nigel Metheringham

27 Sheffield

URL www.shef lug.co.uk
Contact Richard Ibbotson

28 Staffordshire

URL linux.ukweb.nu

29 North East

URL www.shofar.freereserve.co.uk/NELUG

30 London

URL www.lonix.org.uk

31 Thames Valley

URL www.sclug.org.uk
Contact Nick Lambert

32 Liverpool OpenSource

URL linux.liv.ac.uk/LIV_LINUX_UG
Contact Simon Hood

33 Deal Amiga Club

Email superhighwayman@hotmail.com
Contact John Worthington

34 Chesterfield

Email info@spirelug.org.uk
Contact Paul Sims

35 South Derbyshire

URL www.sderby.lug.org.uk/
Contact Dominic Knight

36 Belfast (BLUG)

URL www.linux.ie
Contact Ken Guest

37 Wiltshire

URL www.wiltshire.lug.org.uk
Contact Jason Rudgard

38 South London

URL www.b-lug.org
Contact Dr. Tim Traveres

39 Cheshire

E-mail enquiry@sc.lug.org.uk
Contact Richard Smedley

40 North Wales

URL www.northwales.lug.org.uk
Contact Jonathan Cole

41 Midlands

URL www.midlandsLUG.cjb.net
Contact Pete Thompson

>>NEW
GROUP

42 South Cumbria

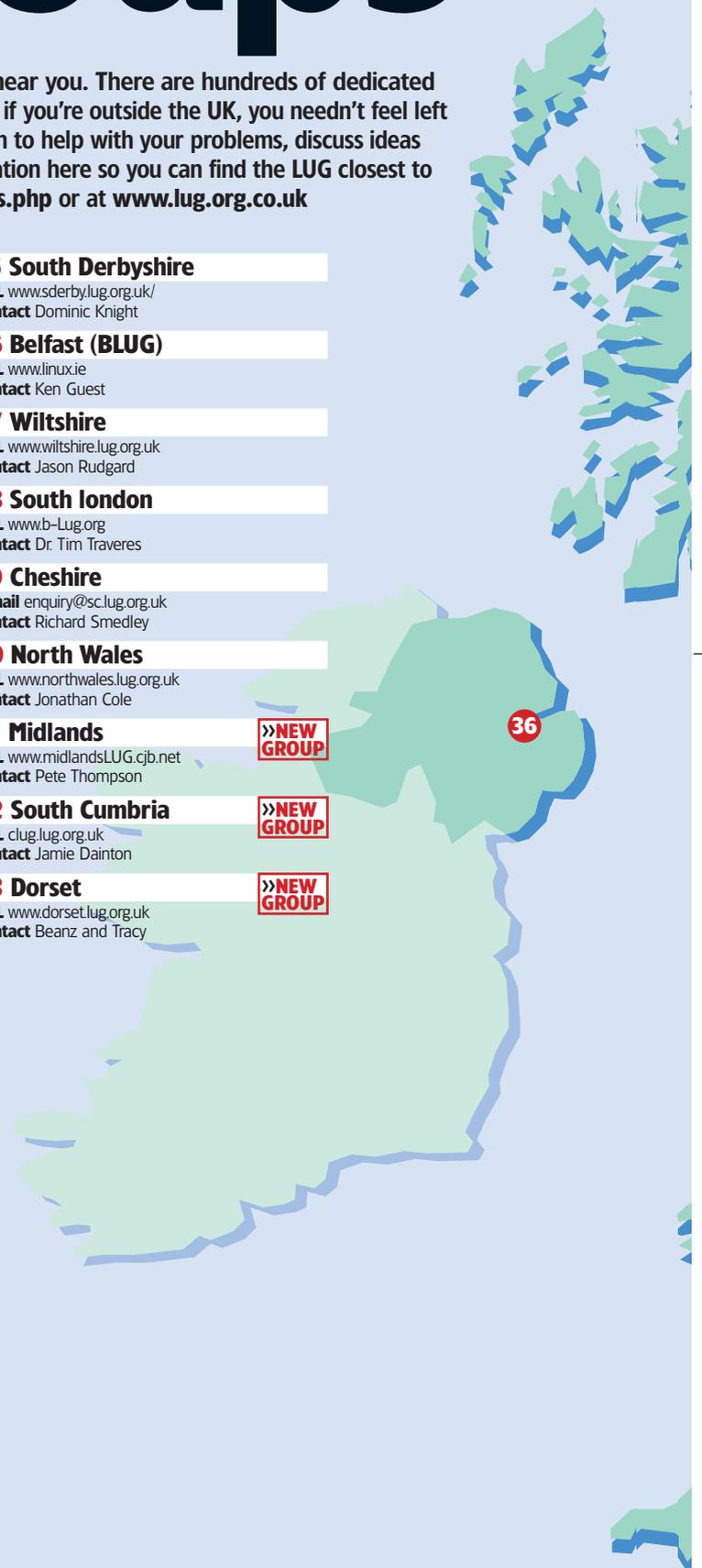
URL clug.lug.org.uk
Contact Jamie Dainton

>>NEW
GROUP

43 Dorset

URL www.dorset.lug.org.uk
Contact Beanz and Tracy

>>NEW
GROUP



STARTING NEXT MONTH

Your group here.....

From next month, instead of having a box of nonsense written by us sitting here, we'd rather have a box full of nonsense written by you!

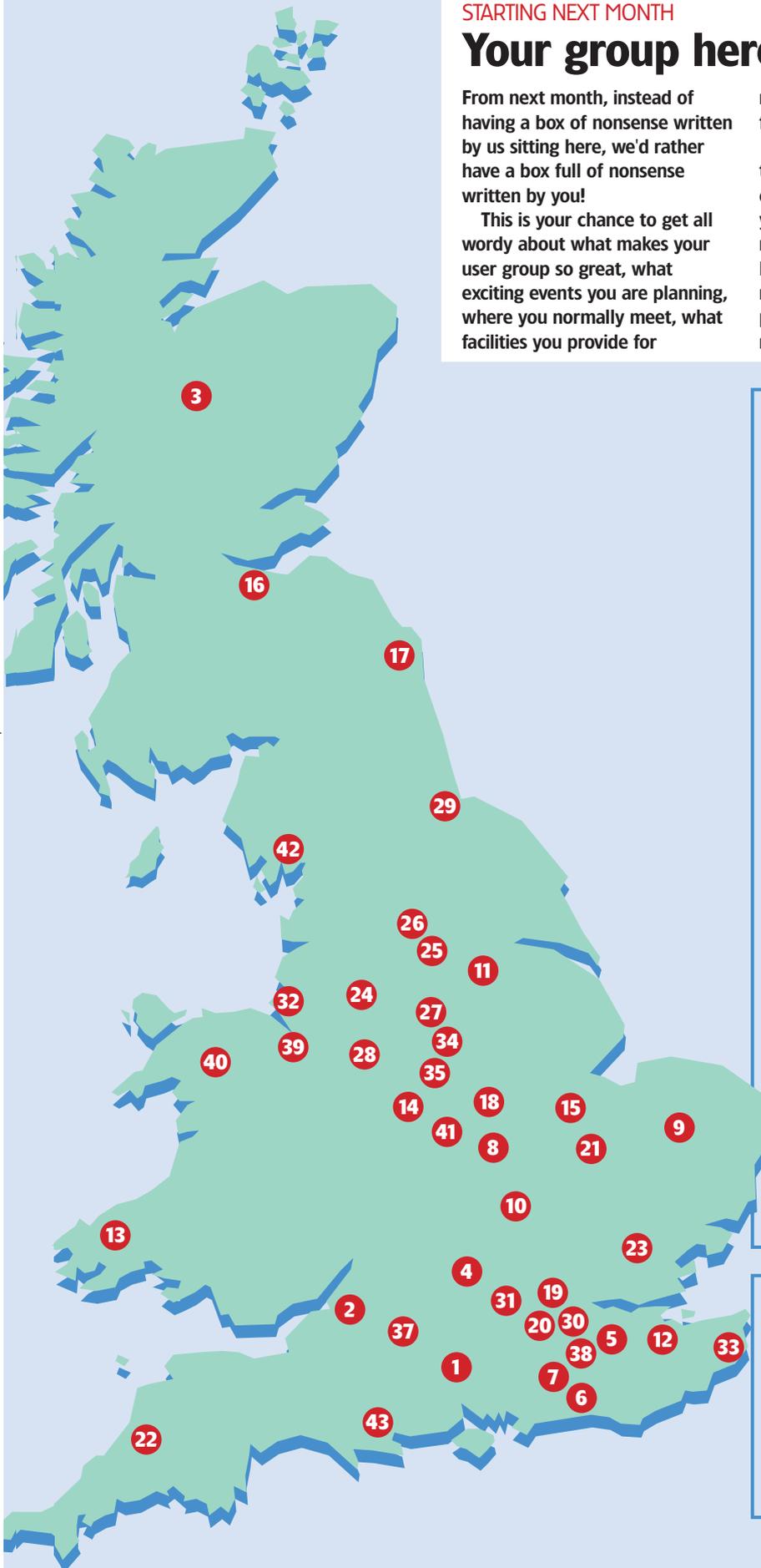
This is your chance to get all wordy about what makes your user group so great, what exciting events you are planning, where you normally meet, what facilities you provide for

members and any other claims to fame you can think of.

Basically, we're offering you the opportunity to promote your club a bit more - you never know, you may attract a few more members or celebrity guests. Include any information you think might be relevant, and a nice photo of one of your meetings/events if possible, and

we'll propel you to fame on these very pages.

You can email us your submission at linuxformat@futurenet.co.uk (please use the subject LUG details or something similar), or alternatively by post to: LUG info, Linux format, 30 Monmouth Street, Bath BA1 2BW. Hope to hear from you soon!

**Worldwide Linux User Groups****Africa**

STELLENBOSCH
 URL www.entropy.sun.ac.za/
 Email ixion@entropy.sun.ac.za

Australia

ADELAIDE LUG
 URL www.linuxsa.org.au
 Email mtippet@anu.edu.au

MELBOURNE, VICTORIA
 URL www.luv.asn.au
 Contact luv-committee@luv.asn.au

PERTH
 URL plug.linux.org.au

Europe

EIRE
 URL www.linux.ie
 Email root@linux.ie

URL www.dilu.org
 Email glossary@dilu.org

GHENT
 URL lsgg.rug.ac.be/
 Email wvdputte@lsgg.rug.ac.be

GOTHENBURG
 URL nain.oso.chalmers.se/LUGG/index.html

LISBON

URL www.students.iscte.pt/~a12593/gul.html
 Email Paulo.Trezentos@iscte.pt

AUVERGNE

URL www.linux-arverne.org/
 Email Cyril.Hansen@wanadoo.fr

India

URL www.river-valley.com/tux/index.html/
 Email anil@river-valley.com

North America

ALASKA
 URL www.aklug.org/index.html
 Email deem@wdm.com

CLARKSVILLE, TN

URL <http://www.cllug.org>
 IRC [#Linux](irc.midsouth.net)
 Email tux@cllug.org

LOS ANGELES

URL www.lalugs.org/
 Email dank@alumni.caltech.edu

BAY AREA

URL www.balug.org/
 Email aftyde@balug.org

DENVER

URL spot.elfwerks.com/~clue/
 Email: lynnd@ihs.com

TAMPA

URL terrym.com/slug/index.html
 Email paulf@quillandmouse.com

BATON ROUGE

URL www.brbug.net/
 Email dpuryear@usa.net

VIRGINIA TECH

URL corvette.me.vt.edu/pages/index.html
 Email nega@vt.edu

VIRGINIA TECH

Email nclug@nclug.org
 Contact Mat Taggart

South America**SAO PAULO**

URL gul.linux.ime.usp.br/
 Contact gul@ime.usp.br

BUENOS AIRES

Email dcoletti@impost.com.ar

MONTEVIDEO

URL www.linux.org.uy/

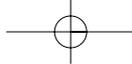
LIMA

URL linux.unired.net.pe/
 Email linux@unired.net.pe

Linux User Group organisers

If you're not listed here, or we have your details wrong, please contact us. It would help if you could write to us with the name of your group, the location, contact name, website address and any other information. Or alternatively fill in the form on our website at www.linuxformat.co.uk/LUGs. Post your details to:

LUGS!, Linux Format, 30 Monmouth Street, Bath, BA1 2BW
 or email your details to: linuxformat@futurenet.co.uk



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 modem 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.



David Coulson is a sysadmin with VA Linux and OSDN. He's well versed in networking and web matters, amongst others.



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...

Hard drivin'

Q I am interested in using Linux alongside Windows on my computer, but I have a problem with installing it. When trying to install TurboLinux from a previous issue of *LXF*, the installer cannot find my hard drive. I am running an ATA 100 HD which I have to define as an ATA 100 device in my BIOS, rather than an IDE hard drive. Does the Linux installer recognise ATA 100s or will I have to wait until a driver comes out?

Liam Sheridan

A Linux will recognise the IDE controller, whether or not it has support for the UDMA extensions. You may want to check to see if TurboLinux recognises the controller and what it says about it during boot, as it will list all drives which are attached to it. Also, it might be useful to see if **fdisk** at the command line (press Ctrl-Alt-F2 to switch to another console) can find /dev/hda which would indicate any problems with the TurboLinux installer. If you still do not have any luck, it is probably best to try Mandrake or SuSE, both of which come with kernels compiled with UDMA support and should recognise your IDE controller without any trouble.

Slouching Mandrake

Q I have recently started on the road to Linux conversion and loaded Mandrake 7.2 from your Christmas 2000 CD, sharing the hard drive with my other OS. But the system seems rather slow. Is this due to sharing disk space?

I have also had problems loading **GNOME** from the September 2000 CD. I chose to use it alongside **KDE** and while it loaded OK, a **gdm** icon appears at the login screen and demands a password which I was not prompted to supply. How can I gain access to **GNOME**?

Roland Crump

A If Linux behaves slowly, it is more likely to be due to having an excessive number of services running, rather than a fault with the installation - lack of disk space certainly would not cause it. If you login from a shell as root and type: **ps ax**, it will list all processes on the machine and

```

2:11pm up 88 days, 2 min, 38 users, load average: 0.08, 0.03, 0.00
151 processes: 133 sleeping, 1 running, 0 zombie, 0 stopped
CPU states: 0.3% user, 2.7% system, 0.0% nice, 96.0% idle
Mem: 255996K av, 241088K used, 11716K free, 55820K sbrk, 894588K buff
Swap: 538104K av, 38812K used, 499292K free

```

PID	USER	PR	NI	SIZE	RSS	SHARE	STAT	LIB	%CPU	%MEM	TIME	COMMAND
32340	root	0	0	456	456	0	S		1.5	0.0	0:00	top
1	root	0	0	128	68	0	S		0.0	0.0	0:07	init
2	root	0	0	0	0	0	SM		0.0	0.0	1:22	klushd
3	root	0	0	0	0	0	SM		0.0	0.0	0:33	klushd
4	root	0	0	0	0	0	SM		0.0	0.0	2:05	ksuaid
5	root	-20	-20	0	0	0	SM		0.0	0.0	0:00	mdrecoveryd
84	root	0	0	0	0	0	SM		0.0	0.0	0:00	krasartfsd
417	bin	0	0	268	224	184	S		0.0	0.0	2:53	portmap
440	root	0	0	92	0	0	SM		0.0	0.0	0:00	rpc.statd
514	root	0	0	13636	6688	212	S		0.0	2.6	22:45	usrapp
551	root	0	0	429	348	276	S		0.0	0.1	54:00	svslogd
568	root	0	0	484	172	140	S		0.0	0.0	0:00	klod
574	dawson	0	0	152	194	76	S		0.0	0.0	0:00	std
583	root	0	0	172	120	84	S		0.0	0.0	0:02	crond
613	root	0	0	156	0	0	SM		0.0	0.0	0:00	cardmgr
780	root	0	0	148	76	80	S		0.0	0.0	0:00	gobind
711	root	0	0	169	124	184	S		0.0	0.0	0:00	gobind

If your Linux box is running too slowly, check that you aren't running unnecessary services.

you can find out, via either **man** or **rpm -qi package** what they do and decide if you think they should be removed or not.

As for **GNOME**, **gdm** is the session manager, so is most likely asking for the password for the user. You can't run both **GNOME** and **KDE** together, as their sessions conflict, but you should still be able to run **GNOME** applications from the command line in **KDE**, or vice-versa.

What?

Q Every time I log in the following message appears in the xconsole:

linux kernel: cannot find map file
and the following appears after it in the messages file:

linux kernel: No module symbols loaded

Is this serious or can I ignore it? What should I do to correct it?

Lindsay Francis

A The map file which your system is referring to is /System.map, or /boot/System.map, which contains all exported symbols for modules. From the message you've been getting, it does not appear as if your kernel has been compiled with support for modules included, so it won't export any, as nothing will be able to use them anyway. The kernel will work fine like this, but it is often best to build a minimalist kernel and have as much extra stuff, such as filesystems and support for various types of hardware etc, loaded in as modules. Many drivers don't need to be present all of the time and, if they are compiled as modules, can be loaded and unloaded on demand to save on memory usage.





Linux+Windows

Q I have Mandrake 7.2 on my box but I would like to add Windows 95 on as a dual-boot. I've tried installing from the Windows CD but LILO just isn't having it. I have seen lots of articles explaining how to add Linux to a Windows machine but not the other way round! Is there a way?

Scott Berry

A Yes, but it is not exactly simple. Windows is very picky about where it is installed, and always like being on the first primary partition on the disk. This may mean that you will have to repartition your drive, or move whatever is your current '/dev/hda1' to another part of the disk. You'll then have to edit /etc/fstab to make sure Linux knows what to mount. Then, you'll have to use `dd if=/vmlinuz of=/dev/fd0 bs=32` to write your kernel image to a floppy disk so that you can boot the machine back into Linux when Windows overwrites LILO.

A better option would be to backup the whole Linux file system on to a CD-R or another hard drive, and install Windows, then Linux, and recover anything you installed or created from the backup.

The I's have it

Q I tried to install the kernel 2.4 as provided with the cover CD of Linux Format Feb 2001, but make zimage fails as "make cannot find a rule for zimage". Do you have any advice?

Eric Gibert

A Sorry, this was a typo that crept into our CD pages unnoticed. Like just about everything else in Linux, the command `make` is case-sensitive. You need to use the rule `make zImage` with a capital 'I' to build a compressed kernel image. If the resulting kernel is too large for LILO to handle, you should try `make bzImage` instead. This will use the more efficient bzip2 method to compress the kernel rather than the usual gzip.

“Windows is picky about where it is installed, and always likes to be on the first partition of the disk. This may mean that you will have to repartition your drive.”

What's the difference?

Q Why do we have different RPMs for Red Hat and SuSE? What's the difference? Can I use a generic (that is, probably Red Hat) RPM on my SuSE 7.0 installation? A good explanation of the difference would be very useful. Keep up the good work.

David Hignett

A Red Hat and SuSE use identical package management systems, but their filesystems and library versions are not always the same. Often configuration files need to go into a different place, or it needs the binary compiled differently, so logs end up somewhere else, but it is usually due to the main distribution having differing versions of major system libraries, which will cause things not to work quite right.

Prints charming

Q After reading 'Cannon Can' from your Linux Answers column Jan 2001, I have an alternative solution.

I have a Cannon BJC 3000 and am running Mandrake 7.2 and Windows 98 on a P233MMX with a 440TX motherboard and 48Mb Ram.

I have tried both USB and parallel port cables for this printer and find that there is little speed difference between USB and parallel print speeds (parallel seems to be a bit faster), but the difference in print quality is immense.

Mandrake has no problem correctly identifying the printer and setting it up when it is connected via the parallel port, but when hooked up via USB it needed a bit of tinkering.

I have fiddled about with things I have no idea about in Mandrake so this may make the difference on my machine, however as parallel cables are relatively cheap.

"silly"

A The main factor which decides how quick your printer is, is the size of the buffer. This determines how much information it can read ahead before it prints it. As you would expect, if it only has a small buffer, which many cheaper printers do, it doesn't matter how quickly you can shovel data into it, because it won't be able to handle a constant stream with that amount of printing information. Realistically, USB and parallel ports are only a factor of ten different in speed, so unless you have a very large and complicated image, such as a photo, it is unlikely that there will be much benefit in using USB unless the printer has vast amounts of memory to cache it all.

Back up, back up!

Q I installed Red Hat 7.0 a number of weeks ago and was happily learning my way around Linux until my hard drive crashed. How truly annoying that was.

Anyway, I have now reinstalled Linux on my

spanking new hard drive, but have realised that a number of files that I downloaded and installed now need to be downloaded again. The thought of getting the 28Mb Java SDK for a second time has me wondering about software for my CD-burner so that I can back it up!

So my question is, is there any good software out there for Linux to enable me to use my CD-burner? If so, could you go over any installation and configuration issues that may arise?

I think that this information would make many newbies such as myself very happy indeed! CD-burning with Linux doesn't seem to be well documented either in books or on the Internet. Thanks very much for your help.

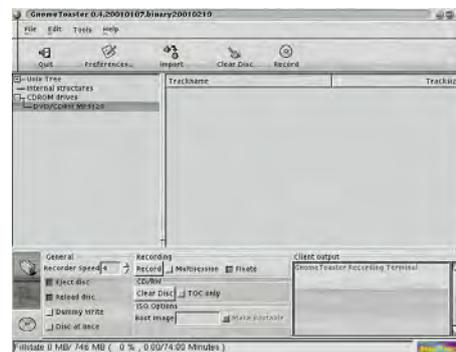
Andrew Hall

A The standard CD-burning software for Linux is `cdrecord` (available from freshmeat.net), but requires the use of the `ide-scsi` module, so that your IDE CD-burner appears as a device on a virtual SCSI bus. This sounds complicated, but it involves nothing more than compiling a few extra modules and making sure that they load correctly by adding the following to /etc/modules.conf

```
alias scd0 sr mod # load sr mod
upon accessof scd0
options ide-cd ignore=hda # don't load
for /dev/hda
pre-install sg modprobe ide-scsi # load ide-scsi
before sg
pre-install sr mod modprobe ide-scsi # load ide-scsi
before sr_mod
pre-install ide-scsi modprobe ide-cd # load ide-cd
before ide-scsi
```

Then, it is just a matter of doing `cdrecord --scanbus` as root, and you'll be able to find the device ID, usually 0,0,0 if you don't have any real SCSI devices. You can either set a configuration file with the device ID and default burn speed, or you can pass them as arguments at the command line. `cdrecord` has a fairly simple set of commands and, assuming you understand how each component will affect the burn speed and functionality, there should be very few problems.

You will also need `mkisofs` to create the ISO9660 filesystem which you will need to use to store data on the CD. This simply builds a filesystem



gtoaster is a very useable frontend for `cdrecord`, the standard Linux CD burner. >>



« from a directory structure or file list, and you can mount it using the loopback device to check that it has encoded the filenames appropriately and then you just burn it to the CD using *cdrecord*.

There are, as you would expect, a number of graphical front ends to make everything a little less painful, the best of which is *gtoaster* which uses both *cdrecord*, *mkisofs* and a number of other tools to help you avoid creating too many coasters.

There is also the CD-Writing HOW-TO over at www.linuxdoc.org, which covers the burning of CDs in Linux in some depth.

Upgrade... Shmupgrade!

Q Your reader who wrote the letter that you published in the Linux Answers section of LX12 is not alone. I had an almost identical experience when I upgraded from Mandrake 7.1 to 7.2.

I have a PII350 dual-booting with Win 98, a Matrox G400 (which is supposed to be well supported) and a 17" WYSE screen. In 7.1 I had 1024x768 resolution, sound, links to my USB Zip drive and all four Windows partitions automatically set up.

I decided not to bother with the upgrade, I just deleted the relevant partitions and did a new install. All went well until it got to configuring X. I had a choice of two versions of XFree86, and I tried both (this involved re-installing from scratch each time). My Matrox G400 was recognised and my monitor as a screen capable of 1024x768 at 85Hz; but the highest display rate offered was 800x600. Hitting 'Show All' gave access to 1024x768 but on selecting that I got a black screen. Finally accepting 800x600, I let the install finish at the third attempt.

When it booted up, I had no sound, although it was showing it had correctly identified my Soundblaster 64 card; all of the links to my Windows drives were gone, also the USB zip, and I gave up on resetting the modem with my UK Linux account.

I went in to *Drakconf* and reset the resolution to 1024x768, and it black-screened on me. After

‘Once you have a functioning Linux installation, there is little reason to do a mass update to the new release unless there are major security issues’

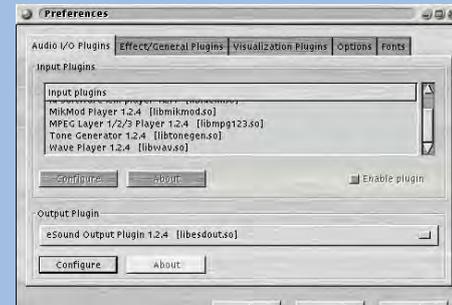
The sound of silence

Q When I boot into Mandrake 7.2, and attempt to play an mp3 audio track through *XMMS*, it often gives an error saying "couldn't open audio". If I select options/preferences, and change the output plugin from the default (OSS Driver) to a different option and 'apply', then back to the default and 'apply', usually everything works in the proper way.

I am certain that the soundcard is set up correctly, because there is never a problem if I use *KMedia Player* for my output. Am I experiencing a known bug with *XMMS* (version 1.2.3) or is there anything I can do in the way of configuration to eliminate the problem?

David Longstaff

A *KDE* uses a daemon which controls the sound output, so that multiple applications are able to access the sound



KDE's audio preferences should allow you to configure settings for a range of applications.

card at once. This is usually handled transparently, but some applications may have problems. We tested *xmms* 1.2.4 and that appeared to work fine with the OSS plugin and *KDE*, so you might find it best to upgrade and see if you experience the same problems.

a reboot it would not go to the graphical login and *startx* said "time out error" and refused to play anymore. So I wiped the lot and went back to good old 7.1.

I will not upgrade any part of my 7.1 system unless I can be convinced that this will not happen again. Perhaps I should wait for version 8 which is in beta already and has *XFree86* 4.something-better, *KDE* 2.1 and kernel version 2.4.1 the last time I looked.

David Robertson

A Once you have a functional Linux installation, there is little reason to do a mass update to the new release, other than if there are major security or stability issues, and there is no guarantee that whatever you install next isn't going to have it's fair share of problems. Apart from major things, including *glibc*, *gcc* and a couple of others, everything else just needs upgrading as, and when, problems are found. Why upgrade *XFree86* if it's working fine already? Why upgrade the kernel, apart from the security issue in pre-2.2.19 kernels, when the sound is happy?

If you want to upgrade, then do it in stages. Before you begin, back up all configuration files, such as */etc/X11/XF86Config*, and then install the new version of *XFree86*. If it works, great. If not, you can reinstall the original release from Mandrake 7.1 and pop your original configuration file back in.

Of course, by the time they release 8.0, the kernel and lots of components will be out-of-date.

SuSE-friendly?

Q I installed your distribution of SuSE 7 on a spare 3.2 GB HD at the weekend, but I am having major trouble setting up my hardware to work with it.

My main concern is not being able to even get a simple 24-bit display of 800x600 sorted. I have a Voodoo3 2000 AGP on my AMD Athlon 650 box, and I am using a Phillips 105s monitor.

I run *Sax* but the best I seem to be able to get (and this is by luck, it seems) is 800x600 or something – so bad I can't even see the clock on *KDE* properly. Also *Star Office* and even *YaST2* seem to need 800x600 to function properly, and I'm normally confronted with 640x480 when I boot up. Please help.

Jerome McKean

A A nicer way is to edit */etc/X11/XF86Config* with a text editor, and manually set the resolution and the bit depth, so that you are 100% sure what X is up to. You will want to change the list of modes for the bit depth you are using, such as:

```
SubSection "Display"
    Depth 16
    Modes "1024x768" "800x600"
    "640x480"
EndSubSection
```

and use the **DefaultColorDepth** setting within the **Screen** section to set the bit depth you prefer. Depending upon the release of *XFree86* which SuSE installs, you may want to check any problems and upgrades for it at www.xfree86.org or SuSE's updates ftp server.

Copy problem

Q As a new reader of *Linux Format* and also taking my first faltering steps in Linux I have a problem in loading the kernel 2.4.1. I have installed Caldera's 2.3 on my system a Pentium133, 48Mb memory and Matrox Mystique graphics card. My problem is cannot



“You may want to just use the kernels from Caldera, rather than trying to upgrade to 2.4. Building a kernel from the source tree is not a simple thing to do”

copy from the CD to hard disk. What I would like to know is how to copy files from the CD-ROM. In issue 12 someone has had a problem in compiling the kernel (from issue 11) using SuSE 7.0, and there is a reference to: `tar zxvf <path>/linux-2.4.0.tar.gz` where <path> is `/mnt/cdrom/othersoftware/kernel2.4.0`.

Now taking this information, I tried to copy from the CD with this line in LILO: `tar zxvf /mnt/cdrom/kernel/linux-2.4.1.tar.gz`. I get this message back: ‘cannot open archive tar zxvf /mnt/cdrom/kernel/linux-2.4.1.tar.gz no such file or directory’.

I have tried loading the 2.4.0 kernel from issue 11 using `tar zxvf /mnt/cdrom/othersoftware/kernel2.4.0/linux-2.4.0.tar.gz` and get the same answer.

Would it not have been easier to put in the correct text, as there are people like me who want to try Linux but with problems like this will give up and revert to Windows.

Robert L Reeves

A To use the kernels from the Linux Format cover disc you need to unpack the archive, which is what the `tar xvzf <filename>` line does, then proceed to configure and install the kernel you have built.

You may want to just use the kernels from Caldera, rather than trying to upgrade to 2.4 and install all of the required packages until you feel confident in doing so. Building a kernel from the source tree is not a simple thing to do, and if you’re not comfortable with Linux and each individual component you are using, such as the kernel source, LILO, tar, et al, then you can quickly become confused and disorientated.

Upgrade now!

Q When I was running my previous kernel 2.2.16 I didn’t have any problem with running *KPPP* to establish an Internet connection. Indeed this email has been sent to you using a 2.2.16 kernel. However if I reboot and use the newly compiled 2.4.0 kernel I get

the dreaded, old “PPPD Died unexpectedly” message. I also get the following PPPD.Log messages:
`pppd 2.3.11 started by root, uid 0`
 Perms of /dev/cua1 are ok, no mesg n’ necessary.
 ioctl (PPPIOCGLFLAGS): Invalid argument
 tcsetattr: Invalid argument
 Exit

I have been unable to locate or identify the cause of this. Quite frankly I don’t really know where to start to find out.

Derrick

A The PPP daemon, *pppd*, is one of the numerous required updates if you want to use a 2.4 series kernel. At least, it is required if you still want to be able to dial up to the Internet. The PPP code has changed a lot, and I’m afraid 2.3.x won’t cut the mustard any longer. You just need to get the latest stable 2.4 build of *pppd*, available from freshmeat.net or the ftp update server for your distribution, and it will work fine. You should also switch to /dev/ttyS0, rather than /dev/cua1, because the latter is no longer the proper device to use with PPP, and has not been since 2.2.x was released.

Q I have enjoyed using Linux since I tried the SuSE distro on a previous issue of the magazine. Since installing it on a dual boot system, I have now upgraded my computer with an additional IDE hard drive. I have managed to install Linux on the new drive (which I want to keep just for Linux), but when I boot up I still get references to the old distro I had installed.

This is still there on the original HD drive, so how can I remove it, and stop it coming up in the menu when I boot?

Jeff Long

A You basically have two problems, one of which is fixing your boot manager, the other is to reclaim the space from your hard drive. The boot problem is fairly easily fixed. You don’t say whether you installed SuSE again on the new drive, or which boot manager you are using (grub or Lilo).

In either case you will need to edit the configuration files. First of all, determine which drive the old distro you wish to remove is on (probably hda something, if it is on the same drive as Windows). Next, find the appropriate configuration file. If you are using grub you’ll find this at:

`/boot/grub/menu.lst`

For LILO it will be:

`/etc/lilo.conf`

Using your favourite editor (e.g. *emacs*, *vi* or whatever) load up the file and remove the sections that make reference to linux on the ‘old’ drive.

For lilo this will be something like:

`image=/boot/vmlinuz`

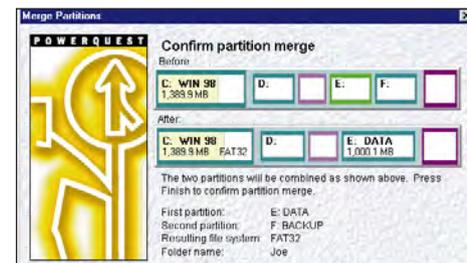
`label=linux`

`root=/dev/hda6`

read-only

Make sure you remove all the lines between the `image=` and the next image info section. The process is similar for grub. For Lilo, you now need to run *lilo* from a terminal to update the settings. When you reboot, the old partition should be gone.

Reclaiming the space from the old partition is a slightly different matter, especially if you want to use it under Windows. To be honest, if you are unsure of what you are doing, you are better off using commercial Windows tools such as *Partition Magic* to reclaim the space and add it back onto your windows partitions.



Partition Magic can reclaim disk space if you’ve removed a distro.

If you definitely want to do things the Linux way, use *Parted* (<http://www.gnu.org/software/parted/>), but in either case make sure you read the documentation thoroughly and make backups, as you can seriously trash all your data otherwise. **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

Reviews

All the very latest software, hardware, games and books put to the ultimate test – the scrutiny of our Linux Format reviews team

IN THIS ISSUE:

SuSE Linux 7.1 Professional

– The first 2.4 distro on test.

Opera 5 Beta 7

– Opera for Linux latest

Network Printing

– O'Reilly's tome on getting your printer working

Professional Linux

Programming – Wrox's guide to Linux hacking

Sound Roundup

– Essential sound utilities on test

Hot Picks

– The very best new Open Source software.

More reviews in these sections...

Desktop 48

Rune – Paul Cavanaugh checks out Loki's latest sword and sorcery epic.

Nautilus – Will Eazel's flagship product revolutionize Linux on the desktop?

Developer 64

Blackadder – Can this rapid application development tool charm Python?

Professional 78

Netvault – Is your data backed up properly? You might want to check out this flexible networked solution.

This month our panel of expert reviewers include **Mike Saunders, Richard Drummond, Biagio Lucini and Jon Kent**

Top Stuff award

If we really, really like something, if it's the best in its field, then we'll give it our "Top Stuff" award. Only the very best will be chosen.



DISTRIBUTION

SuSE Linux 7.1 Professional

■ **DEVELOPERS** SuSE ■ **PRICE** £56 ■ **WEB** <http://www.suse.co.uk>

You've seen the evaluation version, now **Mike Saunders** finds out what SuSE have managed to pack in their Pro box.

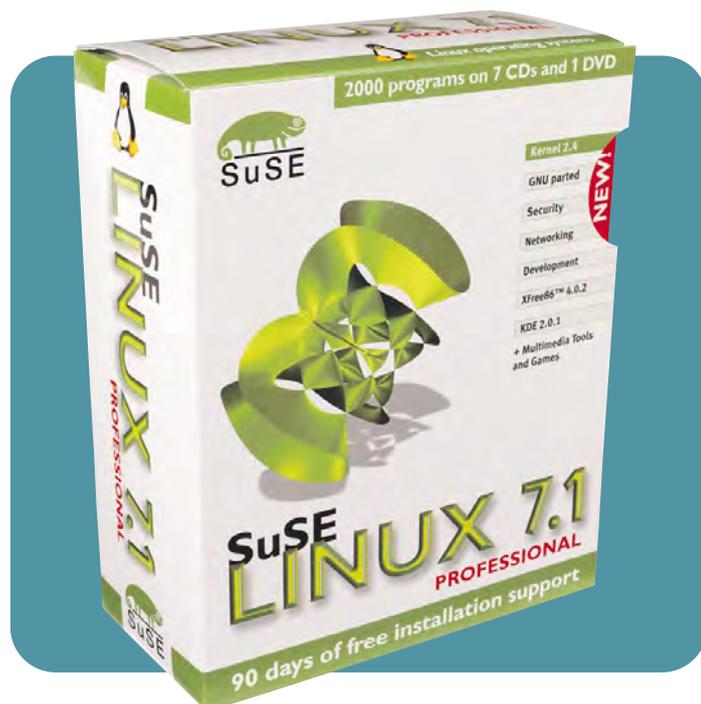
German distributor SuSE GmbH is one of the most widely-known Linux vendors. With a development history rivalling Red Hat and a solid user base throughout many European countries, SuSE occupies a strong position in the ever-growing Linux market and continues to push forward with regular new releases. We looked at 7.0 back in LXF08, and now 7.1 is on the shelves, with many updates and new features.

Since the 7.0 edition, SuSE have followed in their competitors' footsteps by offering multiple versions of their distro – for the newcomer, a Personal edition with three CDs and three handbooks serves as an easy introduction to the OS, while this Professional release offers a more complete solution (for more money) with more software and manuals.

SuSE have earned a reputation for supplying stacks of software and high-quality handbooks, and the previous release we reviewed in LXF didn't disappoint. Moreover, while some vendors such as MandrakeSoft and Storm have both been content to base their distros on other established products, SuSE has evolved into a unique Linux package with prominent integral features, such as *YaST*.

What's in the box?

Along with the two boot floppies (for systems which can't boot directly from the CD-ROM) and installation discs, leaflets advertising other products and services are included together with a sticker sheet and *Geeko* pin badge. The seven CDs which make up the



distro are supplied in a cardboard wallet, and a single DVD alternative is also included for good measure.

Additionally, the wallet has a few tips and general FAQ queries – which is a nice touch, as those who get stuck won't have to wade through the comprehensive handbooks.

Still, SuSE's printed documentation is nothing short of a marvel. In total, four manuals are included in the box, each covering a different aspect of working with the distribution. To get started, the Quick Install booklet steps carefully through the installation process; with a plethora of screenshots

and hint boxes. Each phase is explained clearly, resulting in a highly useful walkthrough for newcomers.

Moving on, the 110-page Applications book discusses basic use of the most common desktop Linux apps – individual sections are devoted to *StarOffice*, *Netscape* and *The GIMP*, all with healthy doses of screenshots and pointers. Smaller tools, including *XMMS*, *Midnight Commander*, *Acrobat Reader* and *Xsane*, are also covered, making the book a handy general reference on any desktop.

SuSE's 270-page Configuration guide deals with everyday administration and setup chores,

Roundup 28

Mike Saunders puts his musical abilities to the test with a round up of the best sound players and utilities available for Linux.

Hot Picks 34

Jon Kent guides you through another collection of open source marvels. This month including **Xine**, **Anjuta**, **Kaptain**, **Exim** and more

detailing such topics as printing, sound configuration, scanners, TV cards and CD writers. Again, each section is written clearly and covers the typical problem areas users are likely to encounter. However, the other chapters – *KDE* and the *Bash* shell – seem strangely out of place, as their sections focus on basic usage and would perhaps be more appropriate in the Applications guide.

Finally, the chunky Handbook tome (which SuSE regulars will remember from older releases) eats up a good 615 pages, and delves much deeper

into matters such as text-mode installation, networking, hardware, kernel configuration and security. Undoubtedly, it's more daunting to approach than the smaller guides, but SuSE old-timers and those with previous Linux experience will find it a valuable resource.

In all, SuSE's magnificent hard-copy docs are the best of all commercial distros available today, and the company deserves a solid thumbs-up for their volume and quality. A couple of issues need sorting out (the occasional German word has slipped

through, and some sections of the Handbook need updating), but they're negligible in comparison to the overall strengths and this sheer amount adds plenty of value to the distro.

YaST2 installation

Since 6.3, SuSE has worked hard on the *YaST2* graphical installer, and a few extra goodies are present in this release. If the BIOS doesn't permit booting from the first CD, the two floppies help out with the second, providing various modules to support extra hardware that may be required

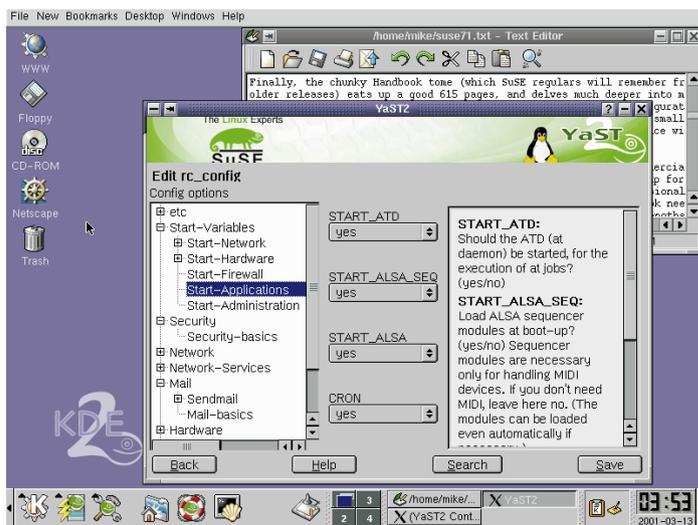
for installation. Also, the initial boot screen offers a choice of the original tried and tested *YaST1* installer (running in text mode).

With *YaST2*, the installation process features six phases, each of which can be skipped back to, should any problems occur. On our test boxes, it correctly identified the type of mouse straight away – failing that, the installer can be operated entirely using the keyboard (though this does become a complete pain when it comes to package installation, and some of the network options are actually impossible to set up using just the keyboard). After initial language selection, *YaST2* moves onto mouse and keyboard configuration before prompting the user for an upgrade or new installation.

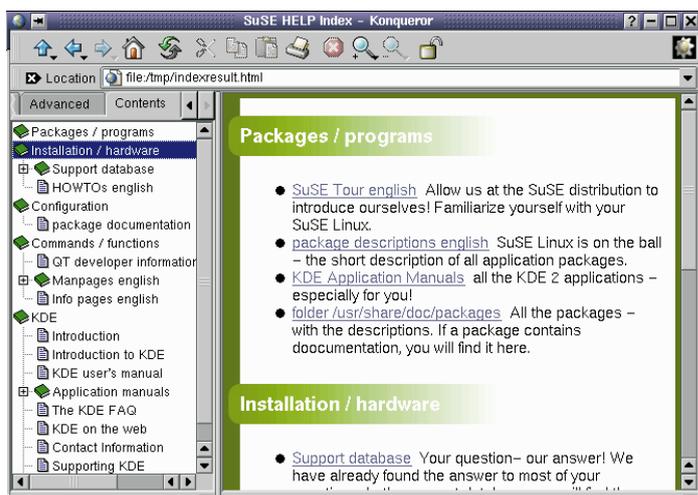
From there, the partitioning phase begins. SuSE's performance in this area has improved with the ability to re-size *Windows 9x/Me* partitions: using a slider, the user can instantly choose the amount of free space on the drive to dedicate to Linux. As many systems have *Windows* pre-installed on a single large partition, this feature is a godsend for those who don't want to get their hands messy with FIPS, though the partitioning tool is not quite as good as Mandrake's *Diskdrake*.

Four main software classes have been pre-configured (All, Minimal, Default and Default with Office), along with a choice of kernels from the mature 2.2 series or newer 2.4 releases, and if both are selected for installation, SuSE handily configures LILO to present a choice at boot-time. For more specific software setup, the 'Detailed selection' option lets the user select individual categories and single packages if necessary.

Before package installation, the LILLO boot loader is configured and automatically detects the presence of *Windows* (customised LILLO setup is also available), and *YaST2* then prompts for default user information and the root password. Following a reboot, the other non-base software is installed and X configuration begins – the automatic monitor and video card detection wasn't perfect on all of our test boxes, but *SaX2* is available later for more comprehensive configuration. >>



SuSE 7.1 has made the step up to **KDE 2.0.1**, here showing the new **rc.config** editor.



Online help has been integrated into the desktop through **Konqueror**.

Configuration issues

As with installation, *YaST* and its graphical brother remain to deal with general system configuration and administration tasks. An attractive front-end gives access to its numerous modules, ranging from hardware setup to network configuration. New in this release is a dedicated **rc.config** editor; this slightly simplifies the modification of the main SuSE configuration file, providing online help and drop-down lists for the options. Many of the other modules work in a similar way to the original *YaST1* versions, with Sendmail setup, security level modification and package management all present. While at first glance *YaST* is similar to Linuxconf, found on Red Hat-based distros, it also uses a system of SuSEconfig scripts which modify parts of the setup, depending on what's installed. Some may find this intrusive, but in most cases it works as expected and is worth leaving enabled. For hardware setup, *YaST* was adequate in sound card detection on our test machines – on one box, we came across errors configuring the sound, but it worked correctly when rebooted. Now *XFree86 4.0.2* is more commonplace, *SaX2* is continuously being improved and did a respectable job on our systems – some manual intervention was required, but its capability is growing in terms of automatic detection on modern 3D cards, and reports on the 'net confirm that it works well in most cases.

« Lastly, *YaST2*'s final phase involves the setup of other hardware, including printers, soundcards, modems/isdn and network cards. Mileage will vary in terms of hardware detection – particularly with more esoteric devices – but on the whole it performed well on our systems. All things considered, then, *YaST2* is a friendly installation program with decent online help and competent hardware detection. Unfortunately, it's also a beefy tool and demands 64Mb to run – but it's worth noting that the character-based *YaST1* still does the job satisfactorily on low-memory machines.

The software side

Once again, SuSE have opted to use the established *KDE* as its default desktop environment – this time, though, it's been bumped-up to version 2.0.1. Sadly for SuSE, *KDE* 2.1 appeared shortly after the distro was released, but updated packages are already circulating around the 'net. Various tweaks to the desktop have been made, and some of the *YaST2* modules have links in the Control Center to give the distro a well-integrated feel.

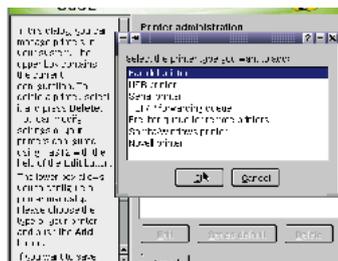
One of SuSE's most championed features over past releases has been *susewm*, a system enabling the currently installed software to be represented on the menus of various window managers. This works well in practice, with an extra 'SuSE' entry on the *KDE* menu with separate sections for each program category (Internet, Development, Games etc.), and other supported window managers include *FVWM2* and *AfterStep*.

SuSE 7.1 is built around kernel 2.2.18 and 2.4.0, supported by *glibc* 2.2.0 and *XFree86* 3.3.6/4.0.2. The newer 4.0.2, while not as complete in terms of support as the older version, is crucial for running X on certain very modern cards and configuration is handled by *SaX2* (see the box). A huge range of alternative desktops are



Additional info

Installer Text/graphical
Package format RPM
Kernel version 2.2.18/2.4.0
Glibc version 2.2.0
Default compiler GCC 2.95.2
XFree86 3.3.6/4.0.2
Default desktop KDE 2.0.1
Support 90 days installation



YaST2 deals with a variety of hardware setups, here showing the printer configuration process.

supplied on the CDs, with such notable inclusions as *Window Maker* 0.62.1, *FVWM2* 2.2.4, *IceWM1.0.4* and *Enlightenment* 0.16.5. Although the emphasis is clearly on *KDE* in SuSE's distros, *GNOME* is also on the discs at version 1.2.4 (with *Sawfish* 0.34), while *Netscape* 4.76 and the new *Mozilla*-based 6.0 is available, and *The GIMP* stands at 1.2.0.

Having seven discs packed with free software and a few commercial demos adds further value to the SuSE distro – users on a dialup modem

connection won't have to trawl the Web and spend hours downloading their favourite apps, and as SuSE RPM packages tend to differ slightly from their Red Hat counterparts, installation is considerably easier too. Virtually all well-known software has made it onto the discs, along with a good deal of source archives. The DVD isn't as chock-full as it could be, but it's only meant to replicate the distro, so no major complaints there.

Those running a server will find *Apache* 1.13.14, *Roxen* 2.1.143, *Sendmail* 8.11.2, *Netatalk* 1.4.99, *Samba* 2.0.7 and *BIND* 8.2.3. The PostgreSQL and MySQL databases are at 7.0.3 and 3.23.32 respectively, and extra tools for LVM and the DHCP and LDAP servers are present too. This is the main difference between the Personal and Pro releases, with the smaller package concentrating on the desktop side and providing less development and network-related tools.

For programmers, GCC 2.95.2 is accompanied by *Python* 2.0 (with a multitude of extension modules), Perl 5.6.0 and the Sun JDK 1.1.8v1. Coding tools and development environments in the form of *KDevelop* 1.3, *Glade* 0.5.11 and *Emacs* 20.7 are supplied, in addition to masses of toolkits and development libs, including Qt 2.2.3, SDL 1.6, ClanLib 0.4.4 and *ncurses* 5.2. This quantity of compilers, interpreters and libraries helps SuSE get well on the way to being a first-class development system.

Various commercial applications are also on the discs, including *StarOffice* 5.2, *Acrobat Reader* 4.05, *VMware* demo 2.0.3 and *Arkeia* backup tool trial 4.2.7. A few more game demos from the likes of *Loki* wouldn't have gone amiss, though. The inclusion of proprietary software in distributions always stirs up debate, but SuSE haven't gone overboard and the freely distributable open source packages completely outweigh their commercial counterparts.

In conclusion, then, SuSE's software range is a rich and varied selection, with packages to suit just about every kind of user. Each has been categorised for easy browsing through *YaST2*, and the proprietary applications have been clearly separated from the rest – always good to see. Aside from very esoteric requirements, developers, desktop users and server administrators will find their favourite tools sitting on the CD/DVD along with the source.

Overall verdict

Despite the relatively small change in version number for SuSE's product, 7.1 is a worthy upgrade for users of older releases and is definitely worth considering for newcomers too. The inclusion of kernel 2.4, *KDE* 2.0.1 and uncountable miscellaneous updates to many other applications and utilities helps it on the road to being the most up-to-date distro currently available, and accessibility features, such as Braille support, improve it even further.

However, MandrakeSoft and Red Hat will be following closely on their heels, so SuSE needs more than just the latest and greatest software to survive. Fortunately, the distro excels when value for money is considered – the colossal amount of documentation and superb range of software, when matched with the £56 price point, is hard to beat. Red Hat's contender at £64 may well have three more CDs, but many are devoted to commercial apps and the printed guides don't compare to the scale of SuSE's.

Of course, those taking their first steps in Linux might feel more inclined to check out the smaller and cheaper Personal version. However, anyone requiring stacks of the latest software, comprehensive documentation and flexible development and server tools should check it out. SuSE have improved and refined the features of 7.1 without any drop in quality, and that's to be applauded. **LXF**



Non-X users need not fear, as the original text-based *YaST1* is still available.



Numerous other desktops are on the CDs, including the lovely *Window Maker*.

Linux Format VERDICT

Installation **8/10**
 Documentation **10/10**
 Features **9/10**
 Value for money **9/10**

Outstanding documentation, a wealth of software and really solid system tools make SuSE 7.1 a distro of excellent value.

LinuxFormatRating

■■■■■■■■■■ 9/10

WEB BROWSER

Surfing with Opera

■ **DEVELOPERS** Opera Software ■ **PRICE** \$39 for the licensed version; adware version free ■ **WEB** <http://www.opera.com>

Biagio Lucini previews the latest beta of the promising Opera web browser. Is it worth selling your eyeballs?

The browser war is entering a new, exciting phase: *Mozilla* is progressing nicely, the first stable release of *Galeon* has just appeared and the improvements in the last *Konqueror* release are impressive to say the least. In this panorama, is there any place for other competitors, especially those you have to pay for?

Opera is the current king of the non-open source browsers and has just seen the seventh beta release of version 5. We've looked at version 4 in the past, but this the first Linux edition to follow in the footsteps of its Windows counterpart by becoming adware. If you do not mind having a banner advert on the toolbar, you can now get *Opera* for free.

We are not going to review the main features of *Opera*, since they were discussed extensively in the Browser roundup in issue 5 and in the preview of version 4 in issue 10. We'll just mention that the speed of rendering, the small size of the package and some revolutionary ideas about window management make *Opera* well worth a trial.

What's new?

This preview comes with a number of long-awaited improvements, including better printing support, better optimization for image rendering, support for foreign bookmarks (useful for those who want to migrate from *Netscape*, *Konqueror* or even *Internet Explorer*) and rescue support for both the navigator and the download

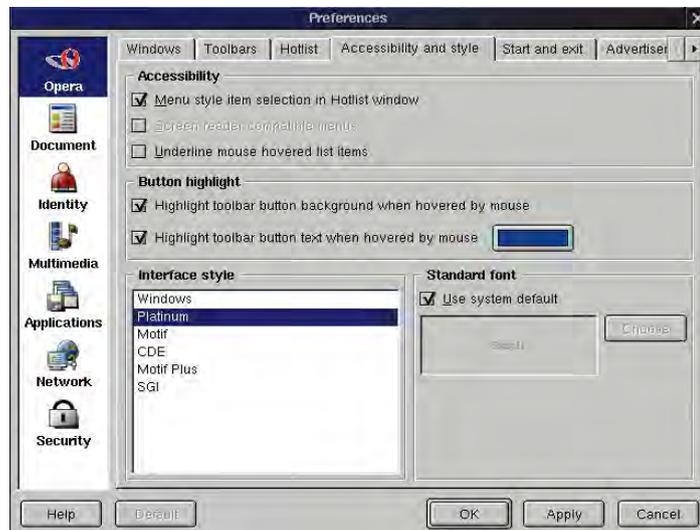
window in case of crashes (this being beta software, crashes can happen occasionally). The application also has the ability to save and restore the current window, improved support for font rendering, and partial drag and drop support (the drop part, only works with *KDE1*). Of course, there have also been a lot of bug fixes.

However, there are still a few high-profile omissions, including any form of Java support, that will be included before the final version is released. Apart from that, the features you would expect from a killer web browser are all present and correct.

The browser package comes with dynamically linked or statically compiled Qt libraries: get one or the other, depending on whether you have installed Qt 2.2.2 on your system or not. Even in the worse case, the download is uncomplicated: the biggest package – the static one – is around 2.6Mb (a tiddler compared to the behemoth *Netscape*). The installation is straightforward – you can choose between *tar.gz*, *rpm* or *deb* package, depending on your distribution – and smooth.

Speed demon

Whatever web browser you use, you'll probably come across nightmare sites due to the lack of standards compliance in the principal applications. This is no longer true: *Opera* is fully compliant with the W3C standard. In fact, the rendering of pages is almost always perfect; we have tested a number of websites



using different technologies that often make the more common browsers fall over. *Opera* handled them beautifully. Even the lack of Java support presents few problems.

Opera's reliability is impressive, in our massive tests we experienced just a couple of random crashes on very heavy pages. However, the strength of this browser is its speed: it seems to render any page, with whatever number of images and animations in matter of seconds, even with a 'slow' 56K modem. It seems that the days of nipping out for a bag of sugar and a cup of coffee before your browser window would display something – anything – are over, finally! *Opera's* speed could save you time and money.

Small footprint

One of the claims made by Operasoftware is that the application requires little in the way of system resources. We have tested this as well, comparing the memory occupied by *Opera* at running time with one, two and three windows open. Compared with the requirements for *Netscape* and *Konqueror* displaying the same sites, we have found only small differences in memory footprint. It's nothing really significant, however it is possible that these requirements will shrink as the product reaches its final release.

The preferences window allows to access a number of new options.

Final thoughts

While still under development, *Opera* seems to be stable enough for everyday desktop use. Despite the occasional missing feature, it has some serious advantages over the Linux competition. Among them, the speed of rendering and the easily usable (after the initial 'shock of the new') and customisable interface.

With Java fully supported, *Opera* has the power to become the standard browser in the Linux community. The improvements since the previous beta are impressive, indeed the developers have devoted a big effort towards the integration of their desktop with Linux – for instance, *Opera* now supports a range of options in the command line.

Still missing is complete integration with the standard Linux desktops – the use of Qt makes *KDE* the obvious choice, but, in principle integration with any of the popular desktop environments should be fairly light work. **LXF**

Linux Format VERDICT

Fast, surprisingly stable and easy to use. One of the best candidates to become the standard browser for the Linux platform.

Opera - the fastest browser on earth!

BeOS | EPOC | Linux | Mac | OS/2 | Windows

search:

Home

Download

BeOS

EPOC

Linux

Mac

Windows

Purchase

Features

Support Desk

Partners

Promote Opera

What's new in Opera 5.0 beta 7 for Linux

Opera Software has unleashed Opera 5 for Linux, it is now free! If you are longing for a faster, more efficient and simply better Internet experience, without having to pay, Opera 5 is the answer.

Opera, long known as the browser for more savvy internet users now enters the free Linux browser market with the its new sponsor-supported version.

The new sponsor-supported version offers users a \$39 value in return for having one single banner ad in the browser interface.

On installation, Opera defaults to the Sponsored mode. However, there is the option of switching to Paid/Registered mode, in which case all ad-related code will be completely uninstalled. The cost to do so remains at \$39.

Linux users can now pay the same as their Windows running mates.

BOOKS

Network Printing

■ **PUBLISHER** O'Reilly ■ **ISBN** 0-596-00038-3 ■ **AUTHORS** Todd Radermacher & Matthew Gast ■ **PRICE** £23.50

Chris Howells discovers a book which sets out to demystify the process of printing over your Linux or UNIX network.

Recognising the fact that networked printing – and sometimes printing generally – can be a bit of a black art under Linux and other versions of UNIX, *Network Printing* from O'Reilly aims to help network administrators to correctly set up and then troubleshoot network printing to ensure they achieve perfect results every time.

The first chapter takes an introductory look at the basics of printing, and how the technology has progressed through the ages – not particularly relevant, but interesting. In the second chapter, various printer languages are mentioned. These include PostScript, PCL and PJI.

Ghostscript is also covered briefly, should you need to work with printers that don't understand PostScript.

The book takes a number of case studies – such as that of Cisco – in order to highlight the methods used to handle network printing in a range of commercial settings. Aiming to be truly useful to those wanting to work with cross-platform networks, *Network Printing* discusses various flavours of UNIX, including Linux, Solaris, and FreeBSD, where appropriate. Additionally, it also covers working with non-UNIX systems, such as Microsoft Windows (using Samba), Apple Macintosh (using Netatalk) and Novell Netware servers.

In terms of printer daemons, the BSD, SysV and LPRng printer daemons are discussed – this should cover virtually every implementation you are likely to come across in the UNIX environment. Methods used to control printing, such as the /etc/printcap file are covered in detail, and differences between the different print spoolers are mentioned.

Subsequent chapters include administration techniques that can be used to manage printers. This includes sections covering central configuration via LDAP, configuring network resources using BOOTP and DHCP, and a few other issues such as accounting, and the ever important

security. Several Perl scripts are helpfully provided, to show what can be done.

The appendices hold useful information such as the format of various printer configuration files – this is great since it pulls all required information into one location.

Linux Format VERDICT

Recommended to anyone that needs to administer networked printers, and mix several UNIX and non-UNIX operating systems in the process.

LinuxFormatRating



BOOKS

Professional Linux Programming

■ **PUBLISHER** Wrox Press ■ **ISBN** 1-861-00301-3
 ■ **AUTHORS** Neil Matthew, Richard Stones, et al. ■ **PRICE** £43.99

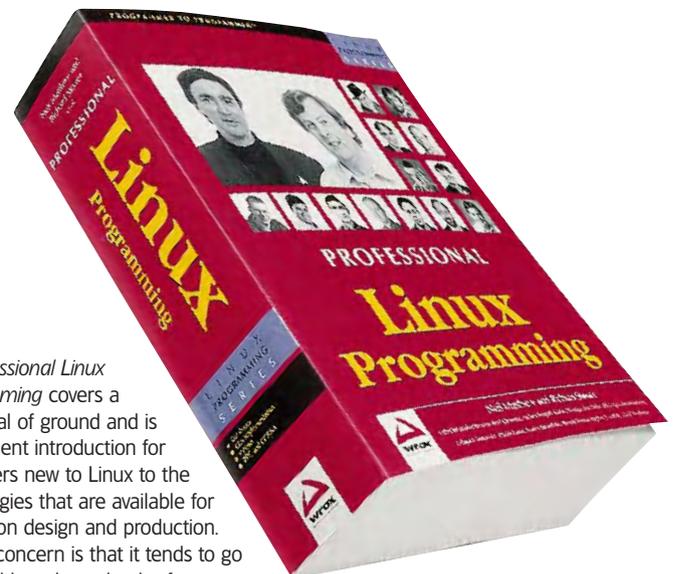
Richard Drummond improves his programming skills with this general reference for the Linux hacker, but finds it could do with more depth.

Professional Linux Programming is a sequel to Wrox's excellent *Beginning Linux Programming*. Its 1200 pages and 16 experienced authors guide the new or intermediate programmer towards building real world applications on Linux.

The book takes a pragmatic point of view, and is hung around an example application: a video database program for a hypothetical video rental store. The focus is development in C or C++, but Python is also covered – though this is only from the point of view of embedding a Python interpreter. The book starts from first principles, covering requirements

specification and program design, before launching into more practical topics like using a concurrent version system (CVS) to manage project changes and selecting and interfacing with a database engine. Also covered are GUI programming techniques – with sections on Qt/KDE and GTK+/GNOME – building world wide web interfaces with PHP, CORBA programming, XML, building RPMs, internationalisation and more. The book is also interspersed with chapters not directly related to the on-going project, dealing with topics like writing device drivers, exploiting multimedia, using Linux on a diskless system and Beowulf clusters.

Professional Linux Programming covers a great deal of ground and is an excellent introduction for developers new to Linux to the technologies that are available for application design and production. My one concern is that it tends to go for breadth and not depth of coverage, and hence the prospective Linux developer will need to turn to more specialist tomes for the real nitty-gritty. Still this volume offers good value and provides a concise overview of the issues and problems facing the prospective Linux developer. It's real world examples should provide a solid foundation to build on.

**Linux Format VERDICT**

A solid one-volume overview of real-world application development for Linux systems.

LinuxFormatRating



Roundup

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



Sound

Music and sound is one of the areas where Linux seems to be lagging. Fortunately, Mike Saunders discovers, the situation is changing...

Was it Clare from Steps who said: "I'm someone who does everything badly except write music." Well, no – after all, they do everything badly full-stop. Neither was it J. S. Bach, for he managed to cram-in the making and raising of 20 children in

his lifetime too (a marvellous achievement, I'm sure we all agree). It was actually the lead-poisoned, hair-ruffled radgematic Beethoven.

Sadly for that great German composer, he was lumbered with merely a temperamental quill and several reams of dodgy paper to craft

his works on. Oh, and a hefty supply of ale, no less. Fortunately for us in the modern world, we have all sorts of tools and computer-related gadgetry at our disposal to make the crafting of tunes so much easier.

The vast majority of music created today – be it pop songs, advert jingles

Our selection at a glance

- Sox
- LilyPond
- Grecord
- SoundTracker
- BladeEnc
- Rosegarden

or cinema soundtracks – has seen the involvement of computers somewhere along the way. From drafting out melodies, editing sound samples or bringing new life to ancient crackly recordings, the role of the computer is increasingly essential to amateur and professional musicians alike.

Play it loud

With Linux steadily gaining further acceptance on the desktop, hackers around the world have turned their attention to the more typical end-user applications that are in growing demand. In our previous Roundups, we've seen that programmers have been working day-and-night to craft the best possible Internet tools, graphics software and games. Of course, one vital area which needs plenty of attention if Linux is to rival other established operating systems on the desktop is sound-related software. Looking back a decade or so, it's clear that Atari's ST range of machines found themselves in the solid position of being a top choice for musicians. Since then, the growth and eventual monopoly of Windows-based PCs has inevitably led to that platform becoming the most common choice for this kind of work.

Still, if you're beavering away on a corking tune (or choon) and have just

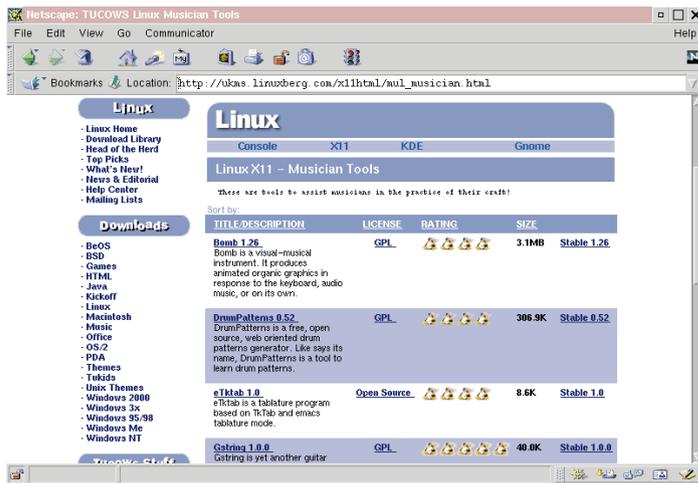
Hit me with your rhythm stick

The needs of users vary greatly when it comes to sound utilities, so we can't put a finger on any specific features that everyone will require. Still, look out for these.

■ **File formats:** when working with sounds or music files from a number of sources, it's essential that the

program can handle them. Trackers work with MOD or XM files, where the sound sample data is stored within the file, while MIDI is more akin to a player-piano in that it only holds the pitch and duration data etc. When managing samples, WAV and raw support is ideal.

■ **Batch operations** – Those who have to deal with masses of sound sample files depend on the ability to convert and enhance multiple files in one job. A good tool should allow large amounts of files to be thrown at it, so you can leave it while it gets the job done.



Tucows, while not as comprehensive as Freshmeat, gives quick pointers to some of the most useful music tools.

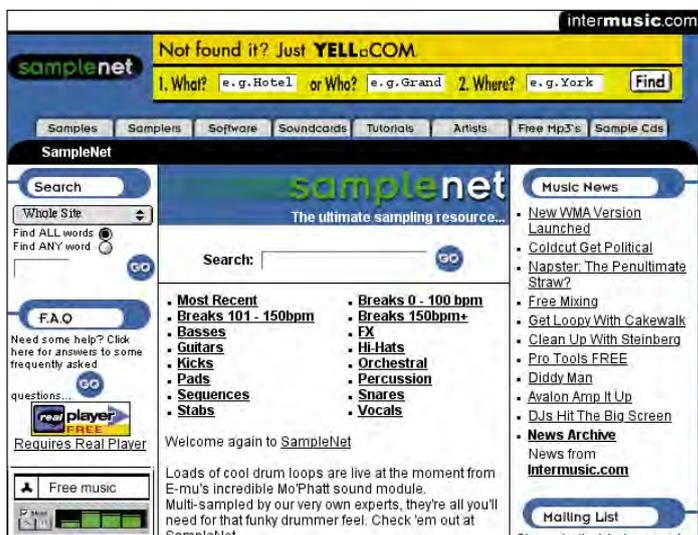
finalised the neatest touches imaginable, watching the system fall to pieces with a Blue Screen Of Death would be nothing less than a dreadful suicide-inducing nightmare. From hell. It's this reason (among the others of price and freedom) that has led to an assortment of sound apps being developed, which will hopefully prise users away from Microsoft land.

The hills are alive...

Long-time Linux devotees will no doubt be satisfied with the recent progress made in the area of sound input and output on Linux. Just a few years back, the task of enabling anything more than a grating beep from your PC's internal speaker was a complex chore; in many cases the kernel had to be recompiled, and all sorts of manual work required before you could get something tonal from your speakers.

Thankfully, if you've recently installed a modern distro then you'll have found the situation to be considerably easier. For starters, packages like Mandrake will automatically detect and configure most soundcards out of the box, so you can be up-and-running with the minimum of hassle. Moreover, we now have another project in the works, ALSA (see the LXF guide in issue 4), which promises a user-friendly and flexible system for sound tools.

This month, we're taking a look at seven of the best and most promising sound applications for Linux. We'll examine a variety of software with differing purposes, from sample editors and MIDI tools to trackers and file converters. So, it's not so much of a direct head-to-head in this Roundup, but a roam through the varying world of sound-related requirements and what Linux has to offer.

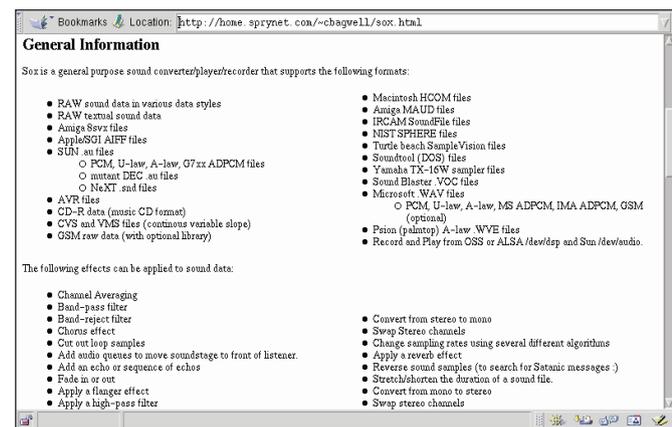


www.samplenet.co.uk is a good place to get, er, samples.

Sox

■ Name Sox 12.17.1 ■ Web <http://home.sprynet.com/~cbagwell/sox.html>

It might not boast a fancy GUI, but Sox is a tool that every Linux musician should have.



Truly interesting screenshots of Sox are nigh-on impossible to take, so here's a grab of the feature list.

Anyone involved with music making and sound manipulation knows that the situation is very similar to the graphics world – all sorts of file formats are battling for domination, with the sheer number making life a tad annoying at times. Similarly, the abundance of formats reflects the platforms on which they originated, and with Linux gaining the image of “all-purpose OS”, it needs to handle as many as possible.

Sox cheerfully describes itself as “the Swiss-Army Knife of sound tools”, giving a hint at the intended versatility of the package. Most distros install Sox by default, but, failing that, it's freely distributable and at 100k, is a small download. Installation is simple, as it depends solely on a few base libraries that everyone will already have.

Four main front-ends to Sox are included in the package: *play*, *rec*, *soxplay* and *sox*. The first three are shell scripts which provide a more intuitive starting point to the range of features and effects offered by Sox, each of which is well documented in the man pages and help files. Also, a couple of cheat sheets are supplied to give quick pointers to common commands.

Sox's supported file format list is excellent – in addition to the popular formats of AU, WAV and raw, a bunch of more esoteric types including Psion WVE,

SoundBlaster VOC, Yamaha TXW, Amiga .maud and Mac HCOM can be used. Various other rarely heard-of formats are also supported, and there's the option to specify “auto” and give Sox a go at detecting the file itself.

While conversion is clearly one of Sox's great strengths, plenty of effects are included to modify samples; once again, the list is impressive. Some of the major effects include echo, reverb, flanger, filtering and resampling. To add to this, mono samples can be converted to stereo, individual channels picked out and reverse effects added.

Overall, then, Sox is one of those toolkits which doesn't have a frilly front-end and, instead, stands up well solely on its feature set. Anyone who works with a range of different sound file formats and requires the ability to do batch conversions and apply effects will find it a supremely handy addition to their program library.

Linux Format VERDICT

Features	8/10
Documentation	9/10
Performance	9/10
Stability	9/10

A marvellous range of supported file formats and effects makes Sox a highly useful tool.

LinuxFormatRating
9/10

LilyPond

■ **Name** GNU LilyPond 1.3.84 ■ **Web** <http://www.lilypond.org>

The Internet may have changed the way music is published, but you can still do it the old way.

Poor Bach. With all those long nights of etching out scores and persuading his wife to make copies for tomorrow morning's cantata performance, he really could've done with an app like *GNU LilyPond*. Continuing the tradition of totally-undescriptive names, this long-running program hopes, according to the developers, "to make music publication software available to anyone on the Internet".

Being a music typesetting package, *LilyPond* accepts plain text files and produces graphical representations of the score. Many boxed-set distros feature it on their CDs, but at around 1Mb for recent RPM packages it's not too heavy a download. You'll need recent versions of *Guile* and *Tetex* to get started, but as it's a console-driven app there are no toolkit or desktop environment requirements.

LilyPond isn't the sort of tool you

can fire up straight away and start fiddling on the spot – it's best to peruse the docs first, which explain how to get started. A number of small test-scores and larger works are available on the site, together with their graphical output, and the overall first-class quality and detail of the documentation is great to see.

Essentially, the source for the music is entered into a plain text editor, with special keywords and structure symbols used to provide order in the chaos – at first glance, seeing "g a bes8 ~ bes16 es, f g as g f es d8 c'16 b! l" can be slightly intimidating, but it's not too difficult to get the hang of. Then, *Tex* output is produced, which can easily be converted to DVI or similar.

LilyPond is a hugely flexible system; almost all aspects of the resulting layout can be modified, including stave size, line width and indentation. Other files can be

Fuga a 3 voci

JOHANN SEBASTIAN BACH (1685-1750)
BWV 847



An example of *LilyPond's* crisp and attractive output.

included, making life easier when working on larger orchestral scores, and perhaps one of the most impressive features is the ability to generate simple MIDI renditions to "beta-test" your musical masterpiece.

Without a doubt, *LilyPond* is very much a specialist app – prior knowledge of musical notation is required, and while the system can be used for a range of music genres, it's best suited to classical works. Still, the comprehensive feature set and beautiful output quality are fantastic.

Linux Format **VERDICT**

Features	9/10
Documentation	10/10
Performance	8/10
Stability	9/10

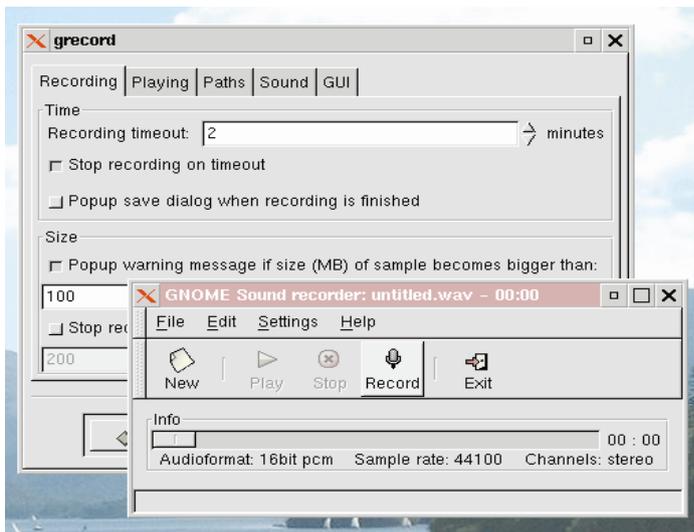
Not the type of tool everyone depends on, but brilliant in its own right.

LinuxFormatRating
■■■■■■■■■ 9/10

GreCORD

■ **Name** GreCORD 0.3.4 ■ **Web** <http://w1.462.telia.com/~u46703521/>

You can really make your compositions unique by recording your own samples.



GreCORD in action with the main window and options box.

Anyone intending to write music on their Linux box needs a good variety of sound samples to start with. Plenty of sites around the 'net offer samples for download, but there's always the problem of copyright – if you're planning to distribute your next masterpiece, things can get tricky.

Often, a simpler solution is to grab the sounds yourself. Most soundcards include a microphone input jack, and, if you've got the device configured correctly, you can sort some leads out and start pulling in the noises yourself. Of course, this can be a tough task if you intend to record real instruments, but it's easy to snag household sounds for that extra personal touch in your work.

GreCORD, as its name suggests, is a *GNOME* sound recording tool which saves the input in the popular WAV format. It's only a tiny download at 130k for the RPM package, and if you've got a recent version of *GNOME* installed it shouldn't spew out any dependency errors.

When started, *GreCORD* presents a simple interface reminiscent of the Windows *Sound Recorder* utility. Below

the set of chunky control icons sits the draggable time slider for stepping through the sample. The menus are equally minimal with just the usual file load/save options and ability to add echo to a recording.

The Settings box offers more features, thankfully, with options for changing the default sample size limit, recording timeouts and sample rates. Even though it won't amaze on the feature front, that's not the goal of *GreCORD* and consequently it's a small and pleasant tool to use when you just need to grab a quick sample.

Linux Format **VERDICT**

Features	5/10
Documentation	6/10
Performance	8/10
Stability	7/10

Nothing major in terms of fancy frills, but works as expected.

LinuxFormatRating
■■■■■■■■■ 7/10

SoundTracker

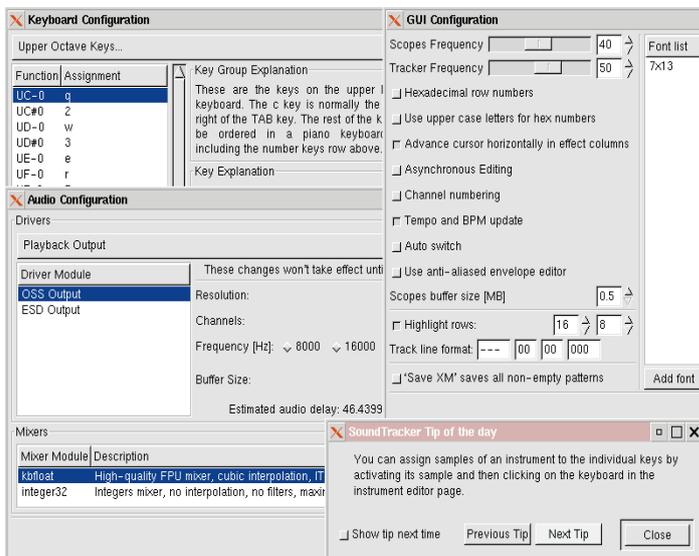
■ Name SoundTracker 0.6.2 ■ Web <http://www.soundtracker.org>

SoundTracker could set you on the road to musical stardom. With some effort. And talent.

Eagle-eyed readers will remember our brief look at *SoundTracker* back in issue 2's Hot Picks. Since then, the program has made steady progress and reached a stable point where a full 1.0 release shouldn't be too far off. Even now, *SoundTracker's* reliability and feature list has enabled it to be used on a day-to-day basis by musicians all around the 'net. Those who remember *OctaMED* and *ProTracker* on the Amiga (and *FastTracker* for MS-DOS) will be familiar with the format of tracker software. Essentially, it's a system for

as a list of instruments and related notes to be played, with the program moving down the list and playing each note as it's encountered. As each track is stepped through in a set time, you can change the duration between notes and create accurate beat patterns just by entering notes in the correct place.

Of course, unless ice-cream van music is your thing, one channel would be useless and thankfully most trackers offer at least four simultaneous channels and some offer far more. So, you can reserve



Configuration box fest here with the keybindings, output and user interface all being tweaked.

creating complete pieces of music from scratch – once appropriate sound samples have been found, you can arrange them into a work of any musical genre, from classical and jazz to pop and techno.

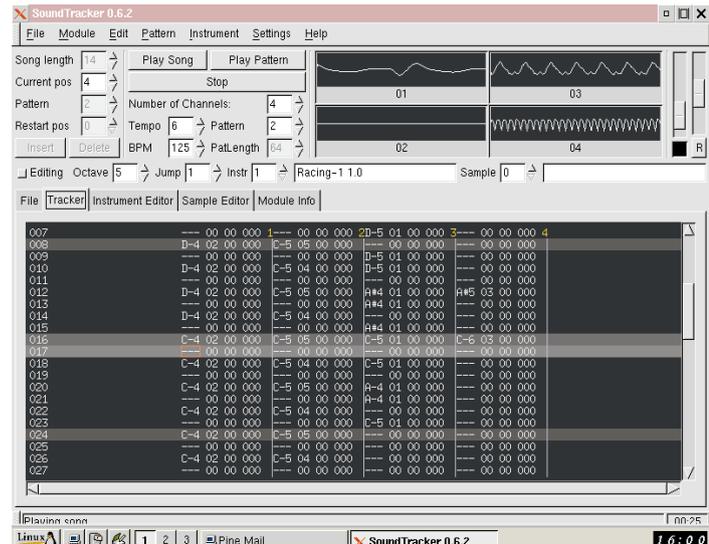
Basic use of tracker software is easy to pick up – imagine each track

one channel for drum-related instruments to add a beat, another for the bassline, and the final two – on a four track system – for individual instrument samples to create the main melody.

Typically, trackers break the music up into patterns, which are just

Ample samples

Before starting work on your music, you'll need some instruments. Typically these are available in WAV or XI format samples around the 'net, and *SoundTracker* sports the ability to assign multiple samples to each instrument. Also included is a handy sample editor, where sections can be copied and pasted and volume ramps applied, while the instrument editor offers more detailed volume and panning envelope editors along with a whole host of other goodies.



SoundTracker's main window, running through a 4-channel tune.

sections of the piece you can copy and paste around. So, if you want to re-use a bassline in another section, you can avoid hand-coding in the notes and just copy it across. Similarly, the order of patterns being played can be edited too, and in all tracker software is hugely versatile for creating music of all types.

California Dreaming

SoundTracker is available from the project's site in both source .tar.gz and pre-built binary RPM packages; with the latter, you can choose between the default version or one built with some extra *GNOME* enhancements. Whichever you select, both are dependant on the Gt+ toolkit (installed on virtually all distros), and glibc 2.1 or later is recommended. At around 230k for recent RPMs, *SoundTracker's* diminutive size is pleasing to see, and when first fired-up, the main window is presented along a handful of general tips for working with the program. Initially, the multitude of buttons and boxes can be daunting to tracker-newcomers, but *SoundTracker's* layout has been carefully tuned and related functions are clumped together.

Along the top sits the basic start/stop transport buttons, together with the pattern-order editor and channel oscilloscope displays. While these remain in constant view, the sections below are presented in a tabbed format, where the main track view, file manager and instrument/sample editors can be found.

SoundTracker supports click and drag selection for copying and pasting chunks of patterns, enabling and disabling of channels on-the-fly (great for DJ-style sample mixing), and a

transposition box for tweaking the current file up and down half-notes and octaves. Also, one other notably nice feature is the ability to bounce down the current musical work as a WAV file – suitable for conversion into an MP3 file, for example.

SoundTracker is capable of supporting up to 32 channels for that ultra-contrapuntal effect, and numerous configuration options are available, with a keybinding editor, playback output setup box (for frequency values, OSS/ESD output etc.) and some switches for modifying the user-interface. In all, it's satisfying to use and does the job in a clean and straightforward manner.

With its small package size, decent running speed and general stability, *SoundTracker* is undoubtedly a wise choice for those wishing to create MOD music of any description on their Linux box. Experienced tracker fans will find a usable and competent interface, while newcomers will enjoy the ability to create full tunes without delving into too many docs. Definitely worth considering if you've got the music in you, but no way of getting it out.

Linux Format VERDICT

Features _____ **8/10**
Documentation _____ **6/10**
Performance _____ **9/10**
Stability _____ **8/10**

A fine example of a tracker: small, fast and simple to use.

LinuxFormatRating
8/10

RoundupSound

BladeEnc

■ **Name** BladeEnc 0.94.0 ■ **Web** <http://bladeenc.mp3.no>

Hop on the MP3 bandwagon with this great encoding utility.

```

[mike@zeil mp3]$ bladeenc bachdude-oboe-5.wav
BladeEnc 0.91 (c) Tor Jansson Homepage: http://bladeenc.mp3.no
-----
BladeEnc is free software, distributed under the Lesser General Public License.
See the file COPYING, BladeEnc's homepage or www.fsf.org for more details.

Files to encode: 1

Encoding: bachdude-oboe-5.wav
Input: 44.1 kHz, 16 bit, stereo.
Output: 128 kBit, stereo.

Status: 7.0% done, ETA 00:04:00 BATCH: 7.0% done, ETA 00:04:00
  
```

Not a great deal to look at, but we can see BladeEnc here in action quietly doing its work.

If you haven't heard of MP3 and the controversy that has built up around it over the past few years, you're clearly reading the wrong kind of magazine. Pop it back on the shelf and pick up *Cross Stitch Monthly* instead, where prominent bands aren't bleating confused threats over the copyright of sewing patterns.

MP3 has proved to be a superb format – while compacting sound files down to acceptable sizes for modem transport, at the same time they retain a good deal of quality and in most cases the difference between direct CD music and MP3 is negligible.

Most of us these days like to keep a collection of our favourite tunes handy on the desktop, and Linux software for creating and managing MP3s is available in all shapes and sizes, with *XMMS*, *Grip* and *GNOMP3* being three popular choices we've looked at in previous issues of *LXF*.

BladeEnc is a command-line encoder which generates MP3 files from WAV, AIFF and raw input formats. It's available for an impressive range of platforms (Linux, BSD, Solaris, Windows, OS/2 and BeOS), and isn't dependant on any obscure libraries so you can be up and running straight away.

By default, *BladeEnc* takes a WAV file from the shell prompt and creates a new MP3 file of the same name, but with a .mp3 extension. Batch operations are supported – you can pass wildcards like "*" to convert all files in the current directory, and also specify exact output file names too. An impressive range of switches are included: bitrates and checksum data can be set, channels swapped around, and input-specific details like signing and byte-order can be set too. As a result, *BladeEnc* isn't just a handy tool for those ripping CD tracks, but for all musicians who want to make MP3s of their creations available to the masses on the Internet.

Linux Format **VERDICT**

Features	8/10
Documentation	9/10
Performance	8/10
Stability	9/10

Does nothing more than encode MP3s, but does the job well.

LinuxFormatRating
 ■■■■■■■■□□ **8/10**

Rosegarden

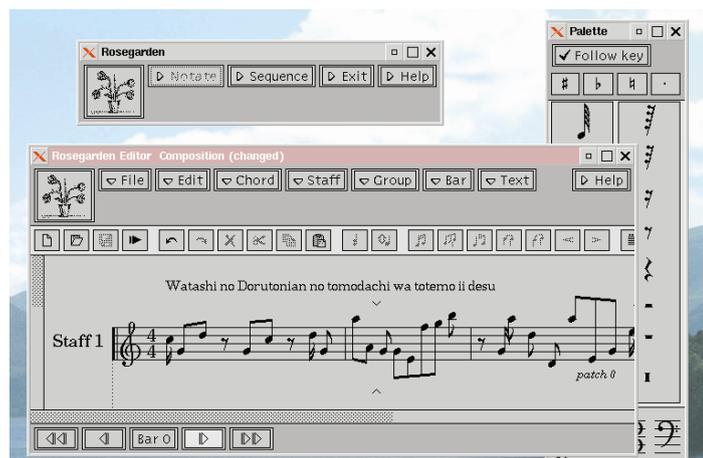
■ **Name** Rosegarden 2.1 ■ **Web** <http://www.bath.ac.uk/~masjpf/rose.html>

I beg your pardon, I never promised you every feature under the sun.

MIDI is still a popular format for distributing music – websites abound for renditions of pop songs, game music and classical works, and more importantly, professional musicians with dedicated hardware such as keyboards rely on it. MIDI isn't seen as so "funky" by some with the development of trackers and instant

music packages, but it's a powerful system and is still in common use. *Rosegarden* is one of the older tools in our roundup and hasn't seen much development in the past couple of years, but it's a mature product and the coders behind it add bugfixes and updates when necessary. At around 1Mb for pre-built RPM packages, the program uses X widgets included in the XFree86 package so you shouldn't be tripped up by any dependency difficulties.

At first glance, and when compared to recent KDE and GNOME apps, *Rosegarden* looks severely ugly and archaic. This isn't an essential issue for a specialist tool, but some work on the interface would be good to see. Initially, a small window pops up to provide access to the two main components of the program: Notate for the editor and Sequence for the recorder and player.



Eat your heart out, Sir Tippett. Now anyone can create atonal rubbish.

Rosegarden's editor sports a neatly-designed drag-and-drop palette box where notes can be placed on the staves, while cut-and-paste is supported, along with the ability to add text and invert selected sections. Various other notation marks can be added and the current work can be exported as a MusicTex file.

From there, the sequencer can be switched for testing; a useful touch, particularly for soundcards which don't support direct MIDI, is the option to use another player like

Timidity. Transpose and quantize features are available for the tracks, while the usual options for selecting devices and patches are also present.

In terms of documentation, *Rosegarden* excels with context-sensitive help for most situations, and it's both detailed and well-written. Despite its unattractive front-end and possible stability concerns (we suffered a couple of crashes in our tests), *Rosegarden* is a detailed tool for composition and editing, and MIDI enthusiasts should check it out.

Linux Format **VERDICT**

Features	7/10
Documentation	9/10
Performance	8/10
Stability	6/10

Looking dated, but still a solid MIDI development suite.

LinuxFormatRating
 ■■■■■■■■□□ **7/10**

Sound THE VERDICT

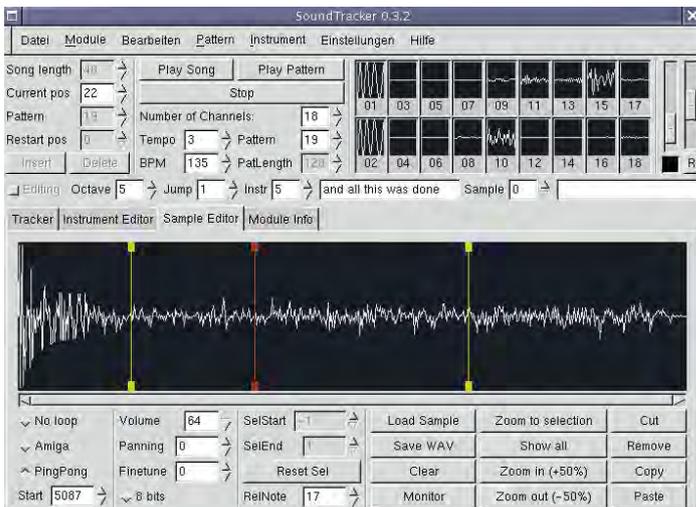
There's nothing to challenge the likes of Cubase, ProTools or Logic here yet, but a number of these applications should be worth keeping an eye on.

These days it's easy to take for granted the huge benefits that computers and related technology bring to the world of music. For instance, glancing back just a decade or so, the ability to get started as an amateur musician was hindered in all areas. If you were lucky, you could get hold of some second-hand instruments from a car-boot sale, find a cheap tape recorder and hand quickly-copied C-60s round your mates in the hope that your talent will be noticed.

Contrast that with today. Now, with some decent PC kit and a 'net connection, embarking on a musical life is considerably easier – using specialist software like the programs we discussed, you can collect together a wide range of samples and effects, carefully arrange them in a tracker, mix

in some vocals and press them on a cheap CD burner. Voila: professional-quality CDs of your own music. No wonder the owners of expensive music studios are looking worried.

Furthermore, the advent of the Internet has improved the situation too. With one of the generous free ISP webspace setups, it's a doddle to convert your music to MP3, pop it on the web, throw a few recommendations around and get noticed. The use of file-sharing utilities such as Napster and Freenet for pirating copyrighted music has occupied the majority of column-inches in the press recently, but this freedom of sharing and co-operation has equally been a boon to upcoming musicians who can use them to develop their fanbase – and a nightmare for record companies



SoundTracker is a very competent musical creation tool.

Table of features

We tested the sound software on a 800MHz, 320Mb RAM Pentium III PC running Mandrake 7.1, with a SoundBlaster PCI 64 soundcard (kernel ES1371 driver). Note: the package sizes and memory statistics below should only be used as a rough

Function	License	Interface	Dependencies	Package size	Memory usage	Alternatives
Sox	Distributable	Command-line	None	100k	N/A	ecasound
LilyPond	GPL	Command-line	Guile	970k	N/A	MusiXTeX
SoundTracker	GPL	Gtk+	Opt. GNOME	230k	6Mb	KegTracker, stracker
BladeEnc	LGPL	Text	None	160k	1.5Mb	LAME, GOGO
Rosegarden	GPL	X11	None	980k	1.5Mb	MidiMountain, MusE
Grecord	GPL	Gtk+	GNOME	130k	3.5Mb	Krecord

A quick mention goes to...

While we've examined a variety of sound software here to cover most requirements you may have, naturally we can't pack everything in and there's bound to be some other goodies that deserve a mention. Many other open-source sound-related gems are being worked on as we speak, and if your fancy a play with some other top apps, have a peek at this bunch:

Kwave – Sound sample editor for the KDE desktop suite
<http://kwave.sourceforge.net>

DJ Krazy – MP3 mixer with beat detection <http://www.linux-support.net/djkrazy>

gTick – Yes, a metronome for your desktop!
<http://www.fortunecity.com/meltingpot/lumpy/846/gtick/>

KGuitar – Set of tools for guitarists <http://kguitar.sourceforge.net>

Pitchtune – instrument tuning utility <http://pitchtune.sourceforge.net>

Musicmatch – mp3 ripping/encoding and CD burning software (look out for a review soon) www.musicmatch.com

The ears have it

What about Linux, then, and its standing as a musician's tool? Looking at the apps we've reviewed here, there's no doubt that developers have recognised the growing demands of users and have put plenty of effort into making free UNIX variants a viable choice in the world of sound. Also, the tradition of keeping software small – do one job, and do it well – is still alive, we're glad to see, with programs such as *Sox* and *BladeEnc* sticking to their basic tasks.

As we mentioned in the introduction, the range and variety of Linux sound-related apps results in no single program being the best of the bunch. Your requirements will dictate the tool you choose, and we've hand-picked some of the best we've found in each department.

So, if you've got some great ideas for tunes and want to implement them with the least amount of hassle,

or just fancy toying with melodies and tweaking music you've found elsewhere, *SoundTracker* is definitely your best bet. As it works with the popular MOD file format, you can pick up a whopping number of existing musical works from such places as Aminet, or simply grab a mic, record some sounds and start work on a track yourself.

On the other hand, those working with individual samples – be it for games, music or professional software – are advised to spend some time with *Sox*. Its wonderful range of supported file formats and pleasing array of effects drops it firmly in the position of being the essential sound-worker's kit. *Rosegarden* is a useful MIDI sequencer, but it seriously needs some updating. Oh, and finally, if anyone knows of a decent harpsichord sample, do let us know. Even Google.com struggled to help with that one... [LXF](#)

Hot Picks

The best new open source software on the planet!

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 CD, 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.

HotPicks at a glance

- Exim
- Xine
- atool
- Procmail
- oftpd
- Anjuta
- Kaptain 0.6
- Snarf

Hot Picks awards

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!



```

Eterm-0.8.10
Eterm-0.8.10
debian/etc# eximconfig
You already have an exim configuration. Continuing with eximconfig
will overwrite it. It will not keep any local modifications you have made.
If that is not your intention, you should break out now. If you do continue,
then your existing file will be renamed with .0 on the end.
[---Press return---]

=====
You must choose one of the options below:

(1) Internet site: mail is sent and received directly using SMTP. If your
needs don't fit neatly into any category, you probably want to start
with this one and then edit the config file by hand.

(2) Internet site using smarthost: You receive Internet mail on this
machine, either directly by SMTP or by running a utility such as
fetchmail. Outgoing mail is sent using a smarthost, optionally with
addresses rewritten. This is probably what you want for a dialup
system.

(3) Satellite system: All mail is sent to another machine, called a "smart
host" for delivery. root and postmaster mail is delivered according
to /etc/aliases. No mail is received locally.

(4) Local delivery only: You are not on a network. Mail for local users
is delivered.

(5) No configuration: No configuration will be done now; your mail system
will be broken and should not be used. You must then do the
configuration yourself later or run this script, /usr/sbin/eximconfig,
as root. Look in /usr/share/doc/exim/example.conf.gz

Select a number from 1 to 5, from the list above.
Enter value (default='1', '%' to restart):
  
```

Exim simplifies the configuration of your email system.

EMAIL UTILITY

Exim

■ VERSION 3.22 ■ WEB www.exim.org

Sending email is second nature to most of us, and we pay little attention to the underlying system that delivers and sends out our words (and attachments) of wisdom, unless paid to do otherwise. Therefore it's worth outlining quickly how this all works. Basically a mail system consists of Mail User Agent (MUA), Mail Transport Agent (MTA), Mail Delivery Agent (MDA) and the user's mailbox. When sending email, the MUA (such as *Mutt*) creates the email and sends it onto the MTA (*Sendmail* or *Exim*) which sends the mail to the destination server over the network (i.e. the Internet). From the destination machine, an MDA (such as *Fetchmail*) takes the mail and post it into the user's mailbox. Some MUAs also act as MDAs as well.

Better than sendmail

With this background, it's time to introduce *Exim*, an MTA. *Exim* sits in the same space as *Sendmail*, and is run instead of it. *Sendmail* is a very powerful program with an extended set of configuration options. This can make it difficult to set up, and its

configuration file is not this most intuitive piece of text. For home use *Sendmail* is overkill in most circumstances, and with the broadband services that are slowly coming online, you'll need an MTA that you can configure confidently, while keeping a weather eye on security.

Exim is derived from *Smail* but with extend features, including defences against junk mail (or Spam). It can be used as a drop in replacement for *Sendmail* as `/usr/sbin/sendmail` on an *Exim* system is in fact a symbolic link to `/usr/sbin/exim`. This ensures that *Exim* is compatible with programs that expect to run *Sendmail*.

Configuration of the program is controlled by the `exim.conf` file (usually located in `/etc`). This consists of four sections:

- Main configuration
- Transport configuration
- Directors configuration
- Remote configuration

The main section specifies the basic configuration, covering the set up of local domains, default domains for messages, user IDs that *Exim* is not allowed to be run under and so on.

This last option (`never_user` in `exim.conf`) allows you to list users that the *Exim* process is not allowed to run under (i.e. root). Should one of the listed users try to run *Exim*, the uid will be switched to nobody.

Pipe down

The Transport configuration section outlines how *Exim* delivers email. The application only supports SMTP, so this is the only transport available, but additionally you can configure a *Procmail* pipe within this section. Director configuration defines how to handle local addresses. Additionally, if configured, the director will run *Procmail* for users who have a `.procmailrc` file within their home directory. Lastly the Remote configuration section defines how *Exim* should handle non-local domains.

As configuration files go, `exim.conf` is not large (about 370 lines) or overly complex and is well commented throughout. With its use of descriptive syntax, the program is very easy to configure by hand. However, it comes supplied with a configuration utility, called `eximconfig`, which makes the job even simpler.

Exim can be run as either a daemon, i.e. available all the time, or can be configured within `inetd` which will then load it when an SMTP connection is requested. If you do not need access to email all of the time, i.e. if you're using a dial up account, then running under `inetd` is fine. However, there is a slight performance hit using *Exim* this way, so if you are on a network, or use a broadband link, it is best to run it as a daemon.

Exim is a very powerful, yet simple to configure MTA. It makes a good replacement for *Sendmail*, so if you have nightmares thinking about *Sendmail* and would like a simpler approach to MTAs, *Exim* is certainly worth investigating.

In a NUTSHELL

Exim simplifies the process of mail delivery and is suitable for both dial up and constant Internet connections. Its also easier to configure, and less complex to use than Sendmail.

DVD PLAYER Xine



■ **VERSION** 0.4.01 ■ **WEB** xine.sourceforge.net/

People hold onto their Windows installations for a number of reasons: access to *Cubase*, the need to play *Quake III Arena* the day it came out; the desire to watch *The Matrix* on your pin-sharp 19 inch monitor. All these are great reasons for keeping Windows, but slowly each one is being eroded.

Playing DVDs under GNU/Linux has not had the happiest of histories, what with the DeCSS debacle and subsequent legal battle. So you'd be forgiven for thinking that you will never be able to play your DVDs on your Linux box. Well, you're wrong and you can: *Xine* will open the door to what is often hyped as 'the best home cinema experience you could have'.

Xine is the DVD player we've all be praying for, and even at this fairly early stage in its development, it's looking very promising indeed. That is not to say that you can't already use this on your own system, you can, and should, to ensure that any bugs are found and fixed. This is a very stable program: we've thrown the best Hollywood can offer at it and it's still not crashed.

Not a pretty face

What *Xine* lacks at the moment is a good interface. Currently, it appears that beauty and functionality are not exactly uppermost in the minds of the developers (which, at this stage, is forgivable), it can't even display the DVD main menu screen. Instead

users are forced to select the DVD 'track' that they wish to view. *Xine* will then play the DVD from that point forward. This doesn't sound too bad, but unfortunately the tracks on almost all movies have no meaningful names (which is not *Xine's* fault), so it can be hit and miss if you want to view, say, the *Stonehenge* part of *This Is Spinal Tap*. However, this is early days yet and most of these issues should be addressed in the near future.

"Hold on" I hear you say, "surely you have the usual CSS problems?". The answer to that is yes and no. When you download *Xine* from the developer's website you will probably only be able to play a few, if any, of your DVDs. However, there is a library call libCSS (currently at version 0.1.0), which performs the decrypting, and a plugin for *Xine* (`xine_dvd_plugin.tar.gz` is the usual archive name for this plugin) which then unlocks and decrypts. The legal status of this plugin is debatable, but it is currently easy enough to find and install. There are some useful links on the website, and a search at *Google* should point you in the right direction.

Full screen

Depending on your video card and version of X, you can play DVD in full screen mode (only available in XFree86 4.0 and greater). If this is not supported by your video card or version of X, *Xine* plays the film in a



Choosing 'chapters' is hardly an intuitive process.



The Terminator: a metaphor for the destructive power of machines?



Or just a really, really violent vehicle for Arnie's acting skills?



Reservoir Dogs on Linux: Happy happy, joy joy!

window, which is the default. In addition to viewing DVD movies, *Xine* can also playback mpeg-1, mpeg-2 streams, AVI files and VCD.

The *Xine* project is a step in the right direction and is must have for anyone who wants to view DVD films (and the other supported formats) under Linux. Grab a copy of it now and enjoy your favourite celluloid moments on your favourite OS!

In a **NUTSHELL**

Xine is not perfect, but is definitely a project to watch in the future. The quality of playback is fine, and some concentration on the user-interface will make this a top quality product.



« ARCHIVE MANAGER

atool

■ VERSION 0.3.0 ■ WEB www.student.lu.se/~nbi98oli/

Atool is a Perl script designed to ease the pain of unpacking tar, zip, gzip, tar+gzip files.

We've all been there, unpacking an archive without checking the file system layout and dumping loads of files into the current directory, usually closely followed by a few chosen words. Spending hours removing all the files is not the most fun you can have on a Friday evening.

atool provides several functions in addition to the aforementioned archive unpacking. These include archive creation, listing and extraction to stdout. But it's the unpacking function that will prove most useful.

To unpack an archive, all that is needed on the command line is: **atool -x speedmgmt.tar.gz**. *atool* will then create a unique directory away from your current working directory and unpack the contents of the archive there. This avoids the horror of discovering you've dropped 3000 files into your current directory. Once everything has been unpacked successfully, the contents will be moved out of this temporary directory and into the current directory, assuming that the archive had a directory structure. If not the archive will be left in the temporary directory for you to rename or move.

```
Eterm-0.8.10
Eterm-0.8.10
jkent@debian:~/download$ atool -l speedmgmt.tar.gz
-rwxr-xr-x verreptj/we2 536860 2001-03-19 12:55:39 mgmt/mgmt
-r-xr-xr-x verreptj/we2 283 2001-03-09 14:14:46 mgmt/speedtouch
-r-r--r-- verreptj/we2 202 2001-03-09 14:14:40 mgmt/speedtch.usermap
-r-r--r-- verreptj/we2 380 2001-03-14 15:50:28 mgmt/INSTALL
-r-r--r-- verreptj/we2 1423 2001-03-09 14:14:32 mgmt/iface.h
-r-r--r-- verreptj/we2 413 2001-03-13 09:56:01 mgmt/Makefile
-r-r--r-- verreptj/we2 6334 2001-03-19 09:43:54 mgmt/LICENSE
jkent@debian:~/download$
```

atool is a small but significant addition to the hackers' toolkit.

atool comes complete with several symbolic links that can be used instead of command line switches.

These links are as follows:

- **acat** – extract archive to stdout
- **atool -c**
- **als** – list file in the archive **atool -l**
- **apack** – create an archive **atool -a**
- **aunpack** – extract files from an archive **atool -x**

This is a nice little file utility which

would enhance any hackers tool box, and which could save you time clearing up any accidental extractions.

In a **NUTSHELL**

A simple tool for managing archives which could save you valuable time if you're prone to pushing the wrong button at the wrong time. It's also small and easy to use.

EMAIL UTILITY

procmail

■ VERSION 3.15.1 ■ WEB www.procmail.org

Procmail is a mail sorting utility program and delivery agent, or in the words of the developers an 'autonomous mail processor'. *Procmail's* sorting abilities allow you to filter your incoming mail, using regular expressions to determine the action that the application should take. The default use of these regular expressions in *Procmail* is against the mail header and not the main body of the mail. The program usually uses the configuration file

`$HOME/.procmailrc` to define the filter rules that are to be used.

Procmail is extremely useful if you subscribe to lots of mailing lists, as it can be configured to move any mail received from these mailing lists to another folder. This allows you to separate your mail out logically and, therefore, makes it more manageable. It's not very difficult to configure and use in a basic setup, but can get as complex as you like. This is one application that can really expand with your requirements.

To use *Procmail* to filter only

your incoming mail, you need to add a .forward file to the root directory of your home account, unless you use *Exim* as your mail delivery agent, in which case skip this section. Using your favourite editor add the following lines to the .forward file:

```
"IFS=' '&&exec /usr/bin/procmail -f-
||exit 75 #username"
```

Replace username with your username and also verify that the path to procmail is correct, changing as required. Now when you fetch your mail it will be run through *Procmail* before being delivered. If you are using *Fetchmail* to retrieve your email another, more elegant solution, is to add the following line to your .fetchmailrc file:

```
mda "procmail"
```

This will inform *Fetchmail* to use *Procmail* as the mail delivery agent. If you use this option you can dispense with the .forward file. As mentioned earlier, if you use *Exim*, you do not need to configure a .forward file, or modify *Fetchmail*, as it supports *Procmail* if you have a .procmailrc file in your home directory.

But before you run off, there are a few things that need to be added to

the .procmailrc file. Edit .procmailrc and add the following to the file, preferably at the top of the file:

```
MAILDIR=/var/spool/mail/[username]
LOGFILE=$HOME/logs/procmail.
logfile
```

The MAILDIR entry simply defines where your mailbox is located, so change this to your mailbox location and filename. Strictly speaking you do not need to add a LOGFILE entry to your .procmailrc file. However, it is good to know how *Procmail* is performing in addition to helping when trying to add new, more complex rules. Just remember to purge this log from time to time to ensure it does not get too large. The best solution is to write a quick cronjob to purge the log on a regular basis.

With the above configuration in place, *Procmail* is not performing any filtering, so you will still receive email as before. Probably the best way to show how easy *Procmail* filters are, is to show an example rule that you could add to your .procmailrc file:

```
:O:
*^From:\ .*debian.org
debian-mail
```

This is a very simple example, which will match any messages from debian.org and then save these into the a folder named 'debian-mail'. To expand the syntax a bit, the rule checks the mail header for any lines that start with From (^From) followed by a single space (\). Then it searches for any character (.) followed by O or

more of the preceding character (any other characters in this case) followed by debian.org (*). The :O at the top of the rule indicates that this is a new rule and the trailing : after :O indicates that a lock file should be used (which is not required, but recommended).

Procmail rules can get very complicated and there is not enough room to discuss them in Hot Picks, but as a taster, have a look at this rule:

```
:O:
* 1^O ^From:([ ]S|$)
* 1^O ! ^From:
* 1^O ^From:.*<>
suspect.mail
```

This rule has several conditions within it, which are essentially looking for mail with an empty From: field and if matched places, this mail into the suspect.mail folder. Great for catching Spam.

If *Procmail* interests you, it is worth while looking into rule creation and then implementing rules to suit your own circumstances. So, go ahead and gain control back from your email hell.

In a **NUTSHELL**

An excellent way to sort your important email from a mountain of Spam. Procmail's scripting isn't very intuitive, but its complexity means the application will grow with you.

SECURE FTP

oftpd

■ **VERSION** 0.2.0 ■ **WEB** www.time-travellers.org/oftpd/

Ftpd (ftp daemon) is the staple diet of the Internet, and has a long history, not all of it good. Lets face it, *ftpd* is not exactly the most secure method of communication, with plain text usernames and passwords being passed over the Internet – you didn't know that! To be totally secure you would want to use *scp*, however this is not always practical. In this situation an anonymous ftp site is probably the best route to go and this is the area

that *oftpd* covers.

oftpd has been designed from the ground up to be an anonymous ftp daemon and nothing else. It also aims to be as secure as an anonymous ftp daemon can be. The program runs as non-root for most of the time, and uses the standard Unix **chroot()** command to hide the system's directories from external users. In other words, it is first run by root and then switches its user id to a non-privileged user one and virtually

moves the root directory to another position (chroot), effectively making a proportion of the filesystem inaccessible to the outside world.

Configuring and installing *oftpd* is straight forward enough, using the usual **.configure; make; make install** procedure. Once this has been achieved though, you need to ensure that */etc/inetd.conf* (or wherever you've chosen to install it), is changed to remove or comment out the ftpd line. Actually, unless you actually need ftp access to your machine this should be commented out as a matter of course anyway.

The next step is to create a user account that *oftpd* will be run under. This is not strictly necessary, you could use an existing account, but a new user account is by far the most secure

method, and allows for easier monitoring. To ensure that *oftpd* is running, you will need to write an initscript to launch it at system startup. The command line for *oftpd* is: **[path to ooftpd]/oftpd [account name] [ftp directory]**. For example: **/usr/local/bin/oftpd ftpserver /export/ftp &**

And that's it. *oftpd* will be providing at least a modicum of security for your file transfers. So, if you are looking to run an anonymous ftp site you should have a look at *oftpd*. It may meet all of your requirements.

In a **NUTSHELL**

If you need anonymous FTP, try this. It is a fairly early release, but it does exactly what it sets out to do.

GNOME DEVELOPMENT TOOLS

Anjuta

■ **VERSION** 0.1.4 ■ **WEB** anjuta.sourceforge.net

Does anyone really need another IDE? When the development environment in question is *Anjuta*, and you are a *GNOME* developer, the answer is probably yes.

The intention of *Anjuta* is to do for *GNOME* hackers what *Kdevelop* and *KDE Studio* have done for their KDE counterparts – create a feature rich, stable development environment with all the tools you need to design an application from scratch in the least possible time.

Anjuta organises files into projects, like other IDEs, and in common with *Kdevelop*, provides a quick-start wizard to takes you through the steps of creating a new project. Terminal, GTK, *GNOME* and Bonobo project types are supported, and *Anjuta* will set up all the necessary files for each.

In the case of *GNOME* projects, this extends to creating Glade compatible interface files to create a generic *GNOME* application, so once the project is set up, all you have to do is build it and you should get a functioning application, albeit one which doesn't do much.

One of the most impressive areas of this IDE is one you might hope not to use – the Debugger. As well as being able to set simple breakpoints, you can use advanced features like setover functions, pause the execution, or even run to the current cursor position. The debugger will examine

core files for you, to help diagnose the more disastrous runtime errors.

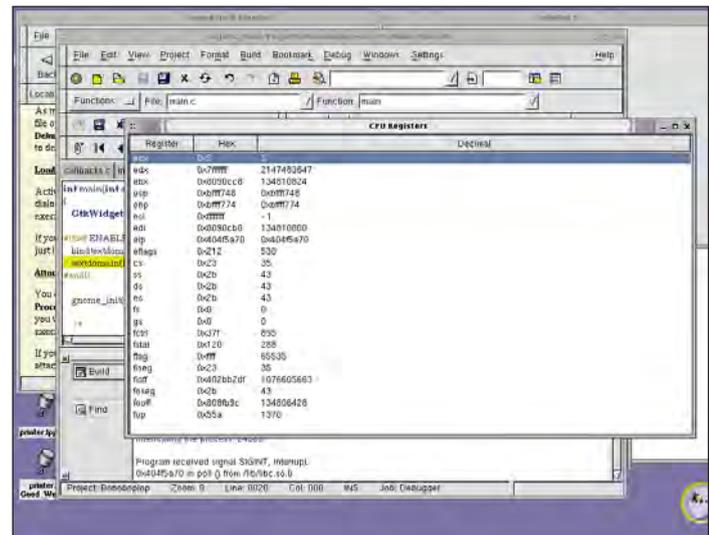
The system isn't perfect or complete yet: there are no split views in the editor, and although it often picks up instances when the files you have open have been changed by an external source (e.g. Glade) this doesn't seem to be always the case. Also, rather annoyingly, Tabs for files in projects seem to hang around even if you have closed the parent project, which can get a bit messy.

Having said that the editor works reasonably well. It has all the usual features you might expect, and a few nice extras like the code autoformatting feature, which can automatically adjust the indentation in your code to a number of preset defaults or your own custom scheme. Very handy.

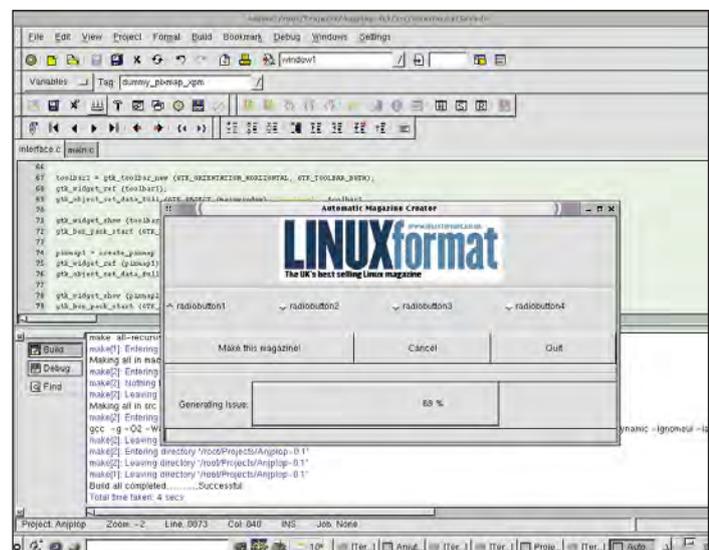
Virtually everything in *Anjuta* seems to be customisable, as the wealth of preference windows will testify, and while it could still do with a bit of spit and polish here and there, it is pretty stable and reliable.

In a **NUTSHELL**

A useful collection of development tools that show a lot of promise for the future. The interface could do with a little polish, but the debugger tool has some exceptional features.



Anjuta aim to provide a similar feature-set that Kdevelop/KDE Studio users benefit from, but for GNOME.



Simple apps can be created in minutes, but useful things will take longer. >>>

LinuxFormatHot Picks

GUI SCRIPT PROCESSOR

Kaptain

VERSION 0.6 WEB kaptain.sourceforge.net

A cunning idea this. There are plenty of console commands that offer great power and functionality, but rely on knowing all the arguments to supply. A lot of these commands have so many options, that even the most extreme console hackers can't be expected to remember them all unless they use them on a daily basis.

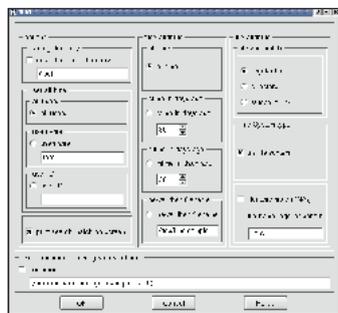
There are plenty of graphical frontends for tools like *mkisofs* and *mpg123* for sure, but *Kaptain* is a little bit different – it isn't a GUI itself, it's a program which processes GUI scripts, thus making it easy to create GUIs for commands without getting you bogged down in a ton of programming.

This approach works pretty well because, for the most part, such programs don't need fancy GUIs, just a single menu to toggle on and off some options and maybe give a few filenames etc.

Example grammar files are currently available for *indent* (the C source prettifier), *Enscript*, *find* and *grep*, and there is full documentation on the *Kaptain* website for creating more. This will require a small amount of scripting ability, but shouldn't tax the average user any more than a scheme or bash script would. For example, the script for the *grep* GUI is just 58 lines

long, and most of them are blank to improve legibility!

As you might expect from the name, this is designed for *KDE*, and requires *KDE2.1* and *Qt* to work properly. And that's its only problem: it would be nice to see a version of *Kaptain* developed that provides a choice of GUIs for people who don't use *KDE*.



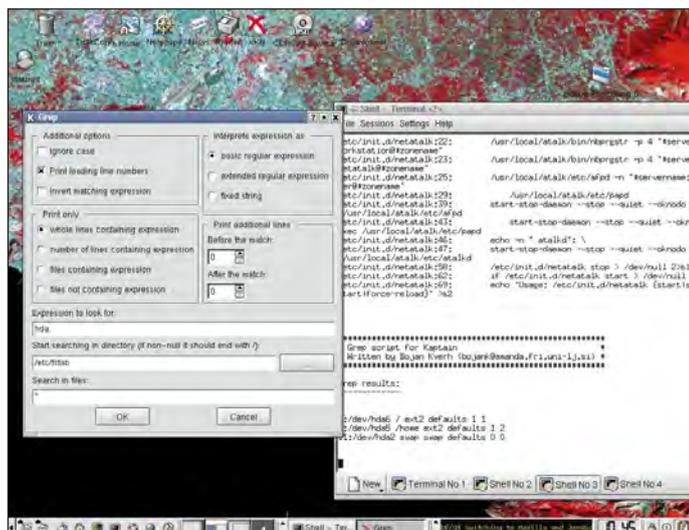
Find's major options are easily selected from this GUI.

In a NUTSHELL

A small, but useful utility that doesn't require acres of programming to get results. However, it's only for KDE 2.1 at the moment so work could be done on making it more accessible.



It isn't even that hard to create the scripts - about 50 lines for find!



A GUI for grep certainly makes it more accessible...

DOWNLOAD MANAGER

Snarf

VERSION 7.0 WEB www.xach.com/snarf

Even the most recent versions of *Mozilla* and *Konqueror* are not exactly the most effective tools for downloading data from the Internet to your PC, especially when you compare them with applications such as *Snarf*. *Snarf* is a simple little non-interactive download utility, designed to retrieve data using several different methods – for example ftp or http. However, once you start to use this tool you'll wonder how you ever managed without it.

In its most simple form you would use *Snarf* to download a web page or image to a local directory using the following command line:

```
snarf
http://www.uk.debian.org/logo.gif
The above command will download logo.gif to the current working directory. You can specify the name you wish the local copy to be called if you do not want to use the original file name by running Snarf with an outfile name, for example:
snarf
http://www.uk.debian.org/logo.gif
debian-logo.gif
```

This will copy the remote file logo.gif to debian-logo.gif on the local system. In addition to renaming the file, you can also ask snarf to send the output to stdout by using a - (dash) as

the name of the outfile. This can be useful if you want to extract data from the web in a quick and dirty fashion. When accessing an ftp site – via ftp or http – that requires you to login, you have to run *Snarf* with the following command line:

```
snarf
ftp://[username]:[password]@ftp.uk.debian.org/[file path]
for an ftp site. And:
snarf
http://[username]:[password]@www.debian.org/[file path]
for an HTTP site.
```

There is a surprising little problem (or bug) with *Snarf* when trying to access an anonymous ftp site. As most anonymous ftp sites request your email address (i.e. fred@blogs.com), *Snarf* thinks that everything after @ is the hostname you are trying to communicate with and fails. Some sites will let you log in without a full mail address, but mileage will vary.

As with most command line download utilities *snarf* allows you to continue an interrupted transfer by using the **-r** command switch. When this is specified, *Snarf* will first check if a local file of the same name exists and then continue the download. For this function to work, HTTP sites need to be using HTTP/1.1 and ftp sites need to support the REST command.

Overall, *Snarf* is a very fast and reliable download utility and is well worth a look. If it was not for the silly email password problem this would be perfect. But that's something the developers could look into. **LXF**

In a NUTSHELL

An elegant download manager let down by a single flaw. However, if you can work around the email password problem, Snarf will ease your download woes.



all you need to know about Embedded Linux cover feature

Linux is going places. **Keith Pettit** reveals why you could soon have Linux in you pocket, fridge, microwave...



Linux as an operating system is continually gaining ground in the desktop and server markets, but there is an explosive opportunity for Linux in the embedded market. Different manufacturers are starting to realise the power and potential of Linux as an embedded operating system. In a nutshell, Linux offers more potential than any other OS on the market; not only is it incredibly customisable, but it's free and there is a huge pool of developers ready, willing and able to shoehorn a complete OS into the smallest of devices. If you are reading this magazine, there is no real need

to explain what Linux is, but the term 'Embedded' might need a little explaining. The intent of this feature is to serve as an introduction to embedded Linux, point out the major players, different development and GUI environments, and discuss some of the current hurdles to world domination in the sector.

Embedded Device Definition

On a basic level embedded devices are designed for certain tasks or functions. They're also designed to be small, cheap, and easy to use. Some examples of common embedded

“The intent of this feature is to serve as an introduction to embedded Linux.”

On the CD



devices would be: ATM, PDA, cash register, cell phone, watch, VCR, TV, etc. To computer manufacturers "embedded" devices have the following characteristics or goals:

1 A specific purpose

A good example would be a hole-in-the-wall cash machine (ATM). ATMs are designed to get the customer's money from their bank account. It's not really designed for anything else; you can't take an old ATM machine and use it as a spare computer for the kids or anything. Once that machine has gone through its usefulness it gets scrapped. »

Embedded Linux



The Ericsson Screenphone, a Linux-based Internet Appliance.

2 Everything integrated

Using the example of an ATM, most have the operating system and software built onto some type of chip or FLASH memory. Later in this article we'll go into more detail about FLASH, and DiskOnChip and how it's being used. The majority of embedded devices are not designed to be upgraded – for example you can't take an old CD player from around the house and upgrade it to play MP3 CDs or DVD movies. If you could, it would take more time and effort than it would ever be worth.

3 Cheap and functional

When most of us buy a computer, we tend to get the fastest machine we can afford, get tons of storage, and go for all sorts of little gadgets to go along with it. With embedded devices, cheaper is better. You basically give the device the features it needs or will make it sell

the best, and nothing else. Who cares if your CD Player has a faster processor, just as long as it works, and if one model is a few quid cheaper...

Turn on, tune in

The basic definition of embedded seems to be expanding. PDAs and set-top-boxes are considered embedded devices, but the standard definition is becoming more and more blurred with these devices. Take the TiVo, it has all the basic components that any computer has, but it's still considered an embedded device.

Jim Barton, Chief Tech Officer/Senior VP for Research & Development at TiVo, helped to come up with a new definition for embedded devices, or at least what embedded means to the set-top market. "At first consideration, most people think of an embedded device as constituting no moving parts, or at least no hard drive. Like a palm pilot, or a VCR, these devices contain an operating system residing on FLASH, or embedded on a chip." But with the TiVo, the best selling embedded Linux device ever, it operates more like a desktop computer. It has a hard drive and the entire OS resides on that drive, just like any other computer out there. TiVo has done some optimisation to help speed boot up, but besides that it launches in pretty much the same way as almost any other machine out on the market. According to Jim Barton, "You shouldn't be aware that you're using a computer. Basically, not have the pitfalls of regular machine; booting up, starting applications with a mouse click, going through menus, etc." With an embedded device you should be able to just turn it on and not think

about it, or according to Jim: "It shouldn't be visible, but get the job done." So according to that definition a laptop rigged up to play MP3's in your car by remote control is an embedded device. You turn it on, it starts the music program right away and you use the remote to navigate through your selections. But in the end it all comes down to what the embedded device was designed for. The TiVo isn't designed to be a fast booting machine. It was designed to be turned on and stay on, record shows, and let the user play back those shows in a 'so-easy-anyone-could-use-it' interface. But for something like a PDA or Cell Phone, even if they did contain internal hard drives for these devices, you wouldn't want to boot off of them because it's a few seconds slower than booting up off FLASH, or DiskOnChip. It would be nice to have a couple gigabytes of disk storage on your PDA, but you wouldn't want to wait 15-30 seconds for the thing to boot up every time you wanted to add an address, or check on an appointment. It all comes down to what the device was designed for, and in the TiVo's case it does what it was designed to do and it performs this function very well.

Now with new PDAs like the iPAQ it's more like a small computer except it uses FLASH to store information rather than a hard drive. But unlike a desktop computer the hardware is still not upgradeable (in most cases). However, you can change the operating system from WinCE to Linux. You can also add and remove software fairly easily. Now with the PDAs further blurring the definition of what an embedded system is, the only rule that seems to be in place with embedded devices seems the idea to make it as cheap as possible, and make it so the user can just pull it out of the box at use it.

Obstacles

Embedded devices are by nature very specialised. The overall goal with embedded devices is to make it easy to use, small, cheap and specialised in whatever task it will perform. This of course pops out some very unusual devices. Look at the market today, at what is considered an embedded device and you have quite a variety. Some have colour touch



screens, some don't have screens at all, others have 200MHz or better processors and others aren't even worth mentioning. Beyond that you have a variety of different ways to power the devices, some have batteries – disposables or rechargeable – other devices are plugged into the wall, or stay in a car, other devices are floating around in space and are powered by the sun. So far, there seems to be no limit on what esoteric devices Linux could power in the future. Linux is the most customisable OS out there and that's the big reason the industry is picking up on it so rapidly, but there are still problems – or rather challenges – that need to be overcome.

First of all, Linux needs to deal with the processor. For the most part on the desktop you are just dealing with x86 type processors. In the embedded market you have four major types of processors to deal with: x86, PowerPC, StrongARM, and MIPS. Any successful OS has to be compiled to run with those different processors, and so do the individual applications that device manufacturers want to run on those systems. This used to be a major problem with development, but advances in the Linux kernel makes the job easier with every release.

One of the major things that keeps a PDA or other similar devices from taking over the desktop market is the limit in processor speed, memory and storage. If you buy a



DiskOnChip is an alternative storage system to FLASH for embedded devices.

top-of-the-line PDA you could consider yourself extremely lucky if you have 32Mb of RAM. With this limited memory comes the problem of how you are going to organise your system, and how you can compress everything – OS, applications and files – into the smallest space.

Swift startup

For other devices where the amount of time it takes to boot is critical, the only options available are to put the operating system, or parts of it at least on some sort of built in memory like FLASH or DiskOnChip. The biggest advantage of using these technologies is, of course, speed, but they're also smaller and consume less power than traditional storage media. Solid-state storage is also more robust as there are no moving parts to wear out or break. Traditionally on these types of devices, the FLASH or DiskOnChip are set to be read-only. This prevents



Embedded who's who

■ **RedHat** is the big name in the desktop and server market and it seems to be a natural move that they go into the embedded sector. RedHat offers embedded development, consulting, training and an embedded development kit (EDK 1.0). Overall RedHat seems to be just getting into the game and figuring out what their strategy is. But because of the size and stability of the organisation, they are going to be a major player in the embedded market for some time to come.

<http://www.redhat.com/embedded/>

<http://www.redhat.com/products/embedded/edk/>

■ **Lineo** was started back in 1998 and from the beginning was setup to start developing solutions for the embedded market. Lineo are trying to be the all-around embedded Linux company by providing embedded Development Kits (Embedix), Hardware, Custom Engineering and support.

<http://www.lineo.com>

■ **MonteVista** This is another company that is completely devoted to the embedded market. Probably their most well known product is their HardHat Linux distribution, which is designed for x86, PowerPC, StrongARM, MIPS and other microprocessor architectures. They are also known for helping to get the ViewML (embedded web browser) project going.

<http://www.mvista.com/>

■ **Century Embedded Technologies** are based in Salt Lake City, were founded in 1985 and is privately held. Century Software's CEO, Gregory Haerr, is the founder and chief maintainer of the Microwindows Project and founder of the ViewML Project.

<http://embedded.censoft.com/>

■ **Transvirtual Technologies, Inc.** are the developer of Kaffe (Open Source version of Java) and PocketLinux. Transvirtual was founded in 1997 by Tim Wilkinson and Peter Mehlitz. Their goal in the embedded market is to "leverage a common software architecture to deliver consistent services to all users."

<http://www.transvirtual.com>

<http://www.pocketlinux.com/>

■ **Sun Microsystems** will now doubt be a long-term player in the industry. Sun's main contribution to embedded development is the Java platform, EmbeddedJava Application Environment, and J2MW[tm] (Java 2 Platform). A version of Java powers PocketLinux.

<http://java.sun.com/products/consumer-embedded/>

<http://www.sun.com/consumer-embedded/>

■ **Trolltech** was founded in 1994. The core team of designers at Trolltech started developing Qt in 1992, and since then the project has steadily expanded and improved. The first commercial version of Qt was released in 1995. Since then, Trolltech have experienced rapid growth, and Qt is currently used in hundreds of successful software development projects world wide.

<http://www.trolltech.com/>

some common disk and write errors associated with regular hard drives. The most obvious disadvantage, at the moment, is the limited space that usually comes with FLASH and DiskOnChip type memory, and the number of times that memory can reliably be written to.

With a desktop or server you go through the regular process of booting your computer, which grabs the information off your hard drive and puts it into RAM. It's simple enough job to save data back to your hard drive. When you're working with FLASH, it is basically the same, but it



Embedded Linux

The Indrema LS600, an ambitious games console based on open source ideals.



gets a little trickier because some embedded systems are designed as read only and never store any information. Others will let you store information, but the process of saving information gets more complicated. FLASH storage is very expensive, so it needs to make the most out of the available space. You can leave the

basis; meaning that individual files and pages in files can be decompressed without decompressing the entire filesystem into RAM, like you would have to if you just compressed the entire filesystem image beforehand. When a page in a file is accessed, it is decompressed into the page-cache in

multiple cramfs images. This way you can break the file system up into major sections and hopefully, when the cramfs image needs to be changed or updated, the image that has to be re-written is not as large. This is very important because of the limits on how many rewrites FLASH memory can handle.

There is a new system out called JFFS2 (Journaling Flash File System) which may solve these problems. According to Bjorn Wessen, the original JFFS "was designed to provide a read/write flash filesystem which would be robust in case of crashes. Since it's impossible to tweak single bytes in a flash, a normal filesystem that requires a 'fsck' phase would be impossible since you can't just fix up the bytes that the crash destroyed, even if you could detect them. Therefore, JFFS is a log-structured filesystem, where changes are written sequentially to a log.

"If a crash or power-down occurs, you replay the log up to the point of the crash. You always know the order that transactions occurred, so you know that when you detect the crash-point, everything before that is OK and you just skip the faulty transaction and start writing new log entries after it.

The disadvantage to JFFS was that it wasn't compressed; so in true open source fashion JFFS2 was created, thanks mainly to the efforts of David Woodhouse and RedHat. JFFS2 has all the journaling benefits of JFFS, but adds the ability to compress data on the fly. So you could have a filesystem that was compressed much like cramfs but unlike that, it has read/write ability – not to mention the journaling capabilities. This has great advantages on devices like a PDA where space is limited. JFFS2 is still at the beta stage at present, but should be ready for general use in the next month or two.

Make the connection

Now with modern PDAs, set-top-boxes and other similar devices, there is the continual difficulty of making them work with other attached devices. For example, iPAQ has a special cradle type design, the cradle itself is a device, and there are also wireless LAN cards, external storage, etc that all connect through the

FLASH card open and in the regular ext2 drive format, but that is really a waste of space. Most people developing embedded devices use a storage system called cramfs.

RAM; there it will stay until the RAM is needed for something else and the page will be flushed until the next time someone needs it. You pay with CPU cycles for the savings in FLASH and RAM." The bad thing is, it is more of a pain to write information to the cramfs image since you have to run a utility that re-writes the entire cramfs image every time you want to make a change, no matter how small that change is.

To help get around this problem and make the system more efficient, you can make a partition on the FLASH filesystem that is in the standard uncompressed ext2 format, for example: /usr/local. This would make it possible for any user settings, or file uploads to be written to that section. But then you still have the problem that when you need to change another part of the file system, you have to re-write the entire cramfs image. So to help get around this problem you could make

"FLASH storage is very expensive, so it needs to make the most out of the available space."

Cramfs (compressed ram file system) is basically making an image of the file system and then compressing it as much as possible. It's designed to be able to read easily, but to write to cramfs you need to run a little utility called "mkcramfs." According to Bjorn Wessen, kernel hacker at Axis Communications: "The advantage of cramfs's is that it provides a read-only filesystem which is compressed on a page-by-page



Embedded Linux

cradle. As a developer you have to keep in mind all the different types of devices and all the different ways those devices will connect up. Not to mention writing the specialised drivers or software that needs to communicate with those devices.

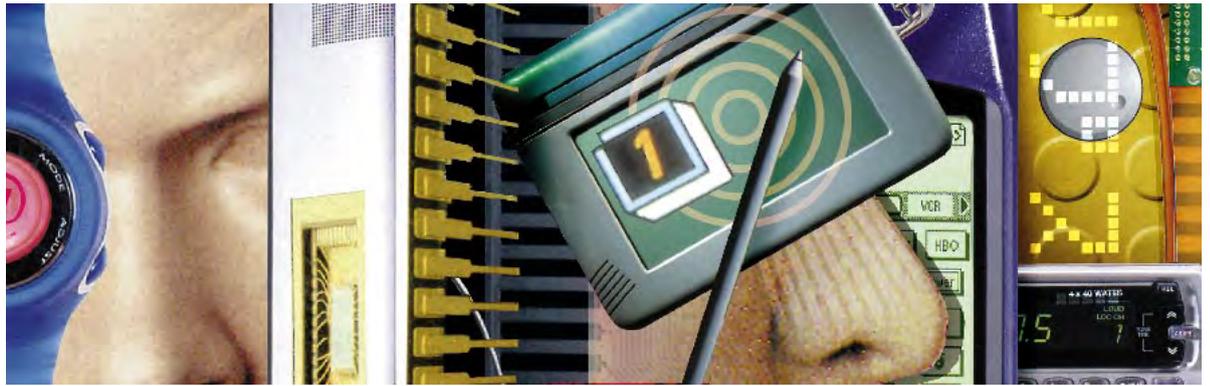
Communication is another big issue with embedded devices. For example a cell phone running Linux that's checking your e-mail has a different set of problems to deal with than a PDA connecting over a wireless LAN. Next you have the iRDA devices like your TV, PDA, or set-top-box. Linux supports all the major communication methods, but that doesn't make it automatically work. There is a lot of work needed to get the correct driver for whatever mode(s) of communication you want to use, or make available. There are no Linux distributions out there that can magically make your device communicate; the protocols for communication are there, but the drivers for that device will almost always need to be customised.

Development problems are the main issue associated with embedded devices, and that goes for embedded systems in general. But that's also where Linux is the strongest: access to the source code means the ability to mold the kernel (and most applications) to whatever the embedded device needs. But as with any new product you still want to find the cheapest overall way to develop your device, and also get it done in the shortest time possible. There are a couple of companies out there that are trying to create development kits to help port Linux to embedded devices. The main goal with these development kits are:

- Configure and build the smallest kernel possible.
- Get the stuff on the devices working (modem, ethernet connection, display, etc.).
- Put only the necessary software components on the system.

Make it pretty

Some people in the Linux community hate to admit it, but they actually use X, and some graphical applications. It's cool to say "Command Line or DIE!" or "vi is the only tool I'll ever need" but in the real world a CLI is not going to be the most popular



choice of interface for Panasonic's latest DVD player. Embedded devices usually have some sort of GUI: your VCR has one to help you change settings and picks channels to record, your TV has one to help change settings, ATM's have one to help you get through the different menus so you can get your money. There are even new cell phones that have large colour displays with graphical menus and other options. The graphical environment is vital.

Even more important than how a GUI looks, is how small it is. In the desktop world you have X on Linux and its size is typically between 30 and 40Mb. Most embedded devices don't have anywhere near that storage, so realising the need for a smaller graphical environment a few companies and developers have stepped up to try and create some fully functional graphical engines that

offer many of the features that X does, but in a space between 256k to 8Mb.

At the most basic level you really don't need X, Windows, Mac OS or any graphical engine at all. You could just start writing code and write out exactly what you want to display, and code in all the functions you need. For example let's say you want to draw a simple box. To do something like that without a graphics engine, you would have to define the video environment and drivers and colour conversion. Then figure out how to handle your mouse/stylus, keyboard, display and all that other hardware specific stuff. This is obviously a time consuming task compared to using a graphic engine where you can just say "draw box, and make it 60 pixels big". An equivalent example would be describing to a friend how to get to a local shop. Most people would just

You'll soon find embedded devices everywhere. It's called pervasive computing.



Embedded Linux

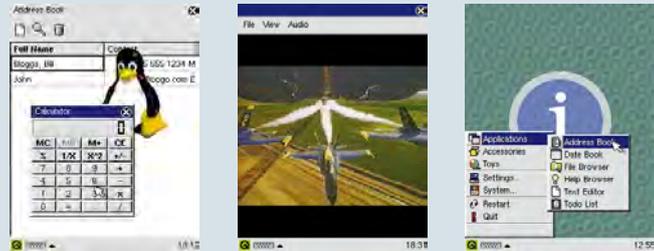
Embedded graphics engines: a quick comparison

»» Microwindows



Graphic Engine Microwindows ■ **Size** 64k-1.2MB
Important Features Extremely small, designed with X/WinCE type APIs.

»» QT/Embedded



Graphic Engine QT/Embedded ■ **Size** 800k-3MB
Important Features Fast, and easy to create/port apps using QT tools.

»» PocketLinux



Graphic Engine PocketLinux ■ **Size** 1.8-8MB
Important Features Uses Java (actually Kaffe) and XML, extremely portable, easy for developers to create apps.

»» X Windows



Graphic Engine X Windows ■ **Size** 75-1.3MB
Important Features X is very common and easy to develop for. Has extensive client/server abilities.

« say: "Take Pine Street, go left, travel 1/2 mile then turn right into the shop." That's easy enough, but if you compare that to developing a graphical application without an engine it would be more like: "shoes, 12 1/2, leather, brown, put shoes on feet, stand, left foot forward, right foot forward, repeat steps 20 times. Use left hand grab door, turn handle, open door, walk through door, etc...." The whole purpose of using a graphical engine is to make it easier for a programmer to develop applications. Also if you're using the same engine on two different machines with different hardware, your application should still be able to work (though it might need to be re-compiled), but if you're not using an engine you have

to change the code and re-define all the different hardware, a process that will probably change the entire program. The way one graphics chip draws a circle can be different than the way another chip draws it on a different system.

Most people have noticed that when they get a new graphics card they'll have to have a particular driver (or X server) to make it work. The reason the drivers are there is to get the most out of your card. They try and take advantage of the card's abilities. For example, if you're using an ATI Rage Pro, you want to use its special features when you play a games like *Quake*. The 3D acceleration, blending, fading, and all the other special features of that card need to have a tailored driver to work properly. You can run the game without the driver, but it might not look as good or run as fast. On embedded devices the concern isn't how fast the card is or what cool function the graphics chip has, but rather just getting an application to work on a device in a limited amount

of time, and still make it able to display your application in an efficient manner. Generally embedded devices operate in the 2D rather than the 3D world. This is where the Frame Buffer, and Framebuffer device comes in.

The frame buffer is an easy way to write graphics to the screen. On

Frame Buffer

Framebuffer – An area of RAM used to store the pixel data for a single screen image, or frame.

Framebuffer device – A framebuffer device is an abstraction for the graphic hardware. It represents the

frame buffer of some video hardware, and allows application software to access the graphic hardware through a well-defined interface, so that the software doesn't need to know anything about the low-level interface.



Embedded tools

MontiVista	Hard Hat Linux http://www.mvista.com/products/cross.devkit.html
Lineo	Embedix http://www.lineo.com/products/index.html
RedHat	EDK 1.0 http://www.redhat.com/products/embedded/edk/
Debian	Emdebian http://www.emdebian.org/

embedded Linux the frame buffer is a driver that can be compiled right in with the kernel. The big advantage of having the graphic engine write to the frame buffer rather than using a specific driver is the speed with which you can get graphical applications running on a new device. For example if you created your graphics engine and application using the frame buffer, all you need to do when a new device comes out is compile the kernel to run on that particular processor (assuming it's different) and maybe write a driver for the input devices. And that's basically it.

Diminutive distros

In this lineup we're going to talk about Microwindows, QT/Embedded, PocketLinux and X. It needs to be said that all of the graphical environments looked at (besides X of course) do not use X Windows to run. All of the graphic engines in this lineup use the framebuffer device to draw graphics, and that makes them very easy to port to different devices. The big issue with these engines is the size, features and language that you use to create applications.

Microwindows

Microwindows was the evolution of the Nano-X project which was started to try and make a smaller version of X for embedded devices. From that start Microwindows also included the Windows GDI, which makes it simple for people to port existing WinCE, or X applications to Microwindows. It is

also easier for programmers coming from those environments to develop applications. For development the system uses FLNX, which is a relative of FLTK (Fast Light Tool Kit). The main goal of Microwindows is to let you create applications in the smallest footprint possible. Where space is paramount, this is the way to go.

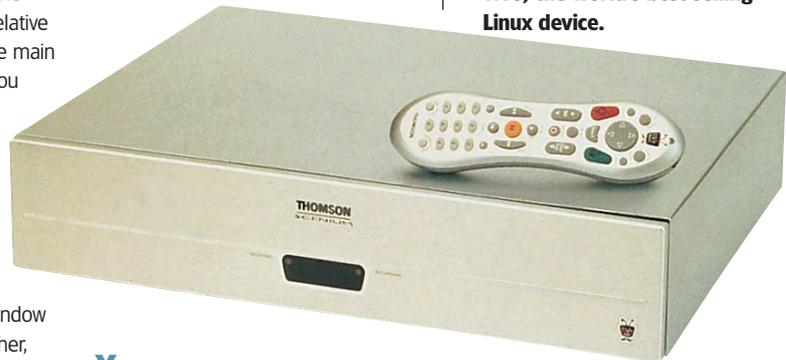
QT

The Qt Palmtop Environment is a complete Window System, Window Manager with application launcher, input methods (virtual keyboard, etc.), a commercial-grade GUI toolkit, and a collection of useful applications, all written using the standard Qt API – the same API found on Qt/X11 and Qt/Windows. But because the Qt Palmtop Environment uses Qt/Embedded, so there is no need for an X11 server, no extra client libraries, no separate window manager, no layer-upon-layer of toolkits – Qt/Embedded works directly with the Linux framebuffer.

PocketLinux

PocketLinux's overall goal is to personalise information. The developers call this a Customised Information Exchange (CIE) – the ability to provide and access synchronised and 'themed' information customised for each user's requirements, regardless of the devices being used. The unique thing about PocketLinux is that it uses Java, or rather a free version of Java called

Kaffe that was developed by Transvirtual. Using Java gives it the unique advantage of having applications that can be developed in any environment. The graphical environment is basically a browser, and applications are designed using XML or Java (Kaffee).



TiVo, the world's best-selling Linux device.

X

When we talk about X in this section we're referring to the slimmed down version of X that comes on the Familiar distribution (formerly the handhelds.org distribution). The big difference on this version besides being reduced enormously is that it is designed to run off the frame buffer rather than using an X server (or driver). This makes it easier for the Familiar distribution (with X) to run on different hardware. Credit for slimming X down to 1.2MB or below goes to Keith Packard and Jim Gettys. The big advantage of using X is its ability to have a client server relationship. So you could have applications residing on a server and pull them up on the client devices. It would also be very easy for old-hand X developers to program in this pared down environment.

We've only just begun

Embedded Linux is still in its infancy; but the Penguin is poised for a takeover. Linux is powerful, flexible and, with its open source nature, developers have far more power compared to closed source solutions. It's not a case of if Linux will completely dominate the embedded market, but when. We've starting to see the beginning of that now with devices like TiVo, new cell phones and PDAs like the Agenda. But we can confidently predict that this trickle of devices will become a flood in the next couple of years. [LXF](#)



iPaq (left) running seminal FPS, Doom.



Emulating a

Simon N. Goodwin reviews Linux-friendly tributes to Sinclair's pioneering ZX Spectrum

Sinclair's Spectrum is the best-selling British computer of all time, and introduced millions of people around the world to the joys of home computing in the 1980s. Compatible systems continued to be made and sold for a whole decade after its 1982 introduction. There are more than a dozen Linux-friendly Spectrum emulators. This article compares them, after reviewing the features that make the Spectrum both a worthwhile and challenging system to emulate.

Spectrum features

By modern standards the ZX Spectrum is a toy, yet it was uncommonly powerful compared with rival home computers on its debut, and needed only a normal TV and portable cassette recorder to make it useful.

An original Spectrum is roughly A4 in size, smaller than most PC keyboards, though with the same key pitch. The first models were expandable from 16K to 48K RAM, which filled the eight bit Z80A processor's address space alongside the 16K ROM firmware. The ROM, from Cambridge mathematicians Nine Tiles, was based on the design for the earlier ZX-80 and ZX-81, doubling in size for each update.

The conductive rubber keys had a 'dead flesh' feel and typically packed half a dozen functions on each button. It inherited the bizarre keyword-entry system which protected earlier Sinclairs against rip-off clones, but this was a mixed blessing when expanded to squeeze more than two hundred keywords and symbols on a 40 key grid. You'd hesitate to program a Spectrum in C, using four

key presses to type each curly bracket!

Most programming was done in ZX BASIC, a slow interpreter, yet a surprisingly fast way to hack together unplanned programs. Unlike the DEC-inspired Microsoft BASIC common at the time, you can stop and change ZX BASIC programs at any time and

continue with variable values, loops and subroutine scope intact. Along with syntax checking as lines are entered, this proved ideal for ad hoc program development, and an easy start for thousands of now-professional programmers.

Despite the hardware limitations there are good implementations of Pascal and Prolog, and even C, Fortran and Cobol compilers for the Spectrum.

Z80 chip

Most professional Spectrum programs were hand-crafted in Zilog Z80A assembly language, a Byzantine elaboration of crusty Intel machine code. The name was a pun on Intel, which stood for INTEgrated LOGic – Zilog stands for 'Z' (as in 'the last of') INTEgrated LOGic. Zilog trounced Intel's eight bit range, but not in bigger or embedded chips. Zilog's complications pose an extra challenge to emulator writers, though a perversely attractive one.

Unlike modern chips, the Z80 was built in random logic, without the benefit of microcode or simulations to structure its design. Extra circuits were literally taped onto the edge of the mask used to make the chip, piling on hidden states and interactions, as well as arguably useful features. The chip-maker documented 600-odd official instructions, some of them very esoteric, but in fact the chip responds to more than 800 op-codes and any decent emulator has to recognise the unofficial ones as well as the standard set. Spectrum programmers used knowledge of these secret features to show off their expertise and to make their code harder to steal.

ZX Graphics

Alongside the Z80A, Sinclair encapsulated Spectrum-specific interfaces into a similar-sized custom chip, a Ferranti logic array. This leapfrogged rivals like the Apple, TRS-80, Pet and BBC Micro, which were built from scores of standard TTL building-block parts. Sinclair's ULA cut costs and made life difficult for clone-makers – and poses more challenges for emulators.

Programs exploit undocumented ports in the ZX82 ULA to circumvent the limitations on colour positioning. Officially the Spectrum supports a wide border in one of eight colours, around a grid of 24 lines of 32 character positions, allowing just two colours chosen from 15 in each grid square of 64 pixels. This



legend

arrangement allowed high-resolution colour graphics – 256 by 192 pixels really was HiRes in those days – but the restrictions of the grid led to ‘attribute clash’ when moving images don’t exactly fit the eight by eight pixel two-colour areas. Ugly as it sometimes is, emulators have to mimic this precisely, catering for precise timing that allows real Spectrum programs to tweak the ULA’s data as the TV screen is scanned.

The key advantage of Richard Altwasser’s hardware design for Sinclair was that the whole screen consumed less than a third as much memory as it would have needed without colour restrictions, leaving more space for programs and dramatically reducing the CPU burden of display animation. Emulators work harder – one eight bit write to Spectrum video memory can change the colour of 64 pixels. True-colour screen emulation has to update a couple of hundred of bytes to get the same effect as a single byte poked to Sinclair’s screen RAM.

Storage stories

Sinclair made four models of ZX Spectrum – the original 16K and 48K models, then the reboxed Spectrum Plus, followed by the Spectrum 128 with extra RAM and three channel sound. This had a serial port, a 48K mode and new 128 mode with an editor, RAM disk and no keyword entry, but few other changes.

The Spectrum cassette interface loaded programs uncommonly fast and reliably by home computer standards, but it was still tortuous, taking five minutes to load 48K at three per cent of the speed of a modern modem. Sinclair teased those unwilling to wait with quirky microdrives – tiny disk-substitutes that churned loops of video tape. They could rival the speed of

floppies once data was located, but initial access time averaged 3.5 seconds on a good day, and an hour or never on a bad one.

Microdrive tapes were expensive and held only around 90K, so a thriving market grew for add-on interfaces for conventional 5.25 and 3.5 inch floppy drives, even though the cost of upgrading was more than the computer.

Later Spectrums were inflated to 128K RAM and 32K then 64K ROM by bank-switching – a bodge which shuffles four 16K

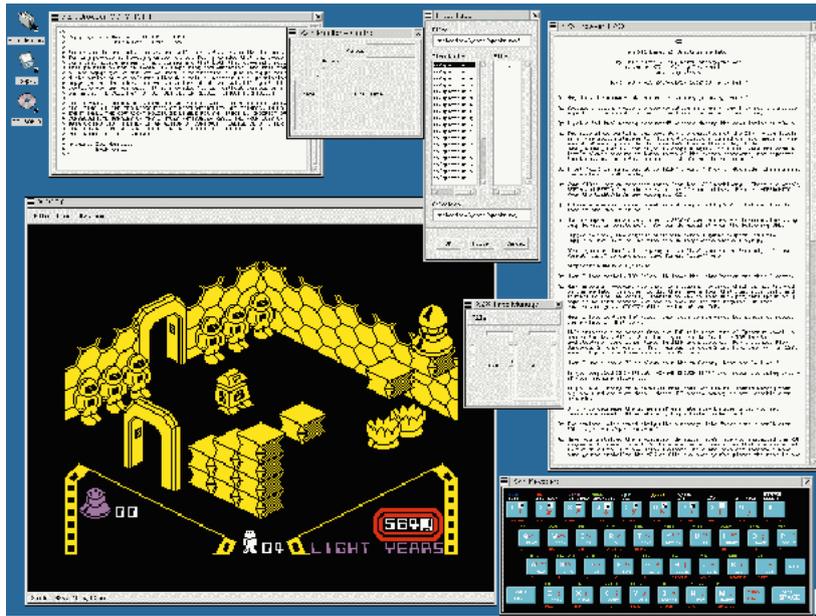
“Most classic software runs in 48k, and Linux loads it in milliseconds.”

chunks of memory in the Z80’s 16bit address space. This hampers emulation of Spectrums with more than 48K RAM. The design of a bank-switching emulator limits its speed, even on modern systems. Processor emulation is typically CPU-intensive, as dozens of new instructions must be performed for each old one interpreted. When a Spectrum writes a byte to a bank-switching port, as often happens in the 128 BASIC editor, any emulator that has to copy 32K will need a fast host to keep up.

The thousand-fold increase in CPU speed in the last two decades means that any 32bit Linux host should manage full-speed 48K Spectrum emulation, but memory advances have lagged behind those in processor cores. If you want to emulate a Spectrum 128 or other machine with bank-switched memory you’re better off with an emulator written from scratch for that job, rather than a 48K one with the bank-switching workaround. >>



DesktopEmulators



ZX3 can run Spectrum software 14 times faster than a native processor.

« Enter Amstrad

Overstock problems after the home micro market peaked led to a 1986 buy-out by arch-rival Amstrad. Alan Sugar's company made millions more Spectrum 128s than Sinclair, scrapping their CPC range to focus on Spectrums with nicer keyboards and integrated floppies. The built-in drive on their Spectrum +3 was a three inch Hitachi format. This flopped in a battle with Sony's 3.5" standard, but a normal 3.5" drive could be added externally, and most PC, Mac and Amiga drives can read that format directly via Linux.

These were cosmetic improvements. Most classic Spectrum software runs in 48K, and Linux loads it in milliseconds. The ZX legacy was taken much further by MGT's SAM, and Eastern block Hobbit, Pentagon and Scorpion clones. Software for these is more capable but less common, and runs on several emulators.

Amstrad's most generous legacy is to allow free non-commercial distribution of the ROM firmware from their 1980s home computers, legalising dozens of Spectrum, QL and CPC emulators. They even allow electronic distribution of the original manuals, making the emulators complete and easy to use.

Emulators compared

There are at least a dozen Spectrum emulators that will run on Linux, plus a couple more for MGT's SAM which has a Spectrum-compatible mode. Of those available, *XCoupé* and *SimCoupé*, do a reasonable job of 48K Spectrum emulation but their real strengths lie elsewhere. Java emulators are discussed at the end of this section.

That leaves eight emulators specifically written to run Spectrum programs under X. The best of these are *XSpecEmu*, *X128* and *XZX*, though it's worth trying others as they're all free and system requirements vary. You may need to fiddle with makefiles to get them to compile properly from source, as they've been developed over many years and not specifically for current Linux builds.

We found *XSpecEmu* by far the easiest to install. The Debian package needed no extra files beyond the basic 2.2 release, and added a direct desktop link to launch the emulator. It's a lot faster than the emulators that run 128K software, but not as fast as the French emulator which has a hand-coded CPU emulation core specifically for x86 systems.

SpectEmu

SpectEmu lacks menus. Snapshot files are selected for loading and saving by pressing a function key and typing the file name and path in a crude terminal window, lacking input history or even editing with cursor arrows.

You can play, pause and stop the emulated tape drive, and trade accuracy of emulation against speed in loading. The emulator can run at standard Spectrum speed – with impressively accurate display and sound synchronisation – or flat out, using all the CPU time it can get. This makes input tricky, but is sometimes handy and often impressive. With *SpectEmu* an AMD K6-2/500 Linux box runs ZX BASIC about twenty times faster than a Spectrum, roughly equivalent to a 70 MHz Z80. The ZX BASIC command **POKE 23561,0** limits the keyboard repeat rate. Press the letter **O** at the start of a line to enter the POKE keyword. Run the emulator at normal speed and save a snapshot with this command, then reload it into a fast emulator to have a chance of typing on an amphetamine-like Spectrum. Keyboard handling in *SpectEmu* is close to that of the original, with some additions for keys not on the original system. Like Sinclair's original, left and right shift keys have different interpretations. You can use modern keyboard cursor arrows, in place of the original shifted digit combinations, and a combination of control and function keys call up new options.

Other buttons reset, quit or give basic help, and one resynchronises the video and keyboard if they get out of step. We didn't find that a problem, though *SpectEmu* did reject some snapshots we tried, showing garbled graphics and resetting the emulator. Most emulators have similar problems, although the extent varies and files that one accepts might clobber others.

X128 and ZX

X128 is controlled with function keys that pop up menus in the original ZX font. This fits in well with the original system and allows changes on the fly. We've enjoyed using *X128* since we found Paul Hill's Amiga port of James McKay's code; it benefits from the extra speed of modern CPUs, but the Linux version needs updating to match the features on other platforms.

The shareware *XZX* favours X menus and windowing, but requires the Motif GUI toolkit, rather than GTK++ or KDE extensions customary on Linux. The 'open' version of Motif rarely features in common Linux distributions. We got *XZX 3* working with Lesstif, a Motif clone, but not all the sub-menus worked and those that did were messily rendered.

SinLinks

Planet Sinclair: <http://www.nvg.ntnu.no/sinclair/planet/>

World of Spectrum: <http://www.void.jump.org/>

■ Magazines

Crash: <http://www.mjwilson.demon.co.uk/crash/>

Sinclair User: <http://www.sincuser.f9.co.uk/>

Your Sinclair 1: <http://www.yasnry.co.uk/>

Your Sinclair 2: <http://homepages.enterprise.net/cavan/ysac/>

Your Spectrum: <http://www.users.globalnet.co.uk/~jimg/>

■ Emulator homes

SpectEmu: <http://www.inf.bme.hu/~mszeredi/spectemu/>

X128: <http://www.void.jump.org/x128/>

XZ80: <http://www.comlab.ox.ac.uk/oucl/users/ian.collier/Spectrum/>

XZX: <http://www.philosys.de/~kunze/xzx/>

■ Java Spectrums:

Hob: <http://www.emuunlim.com/hob/>

Jasper: <http://www.spectrum.lovely.net/>

JXSpeccy: <http://www.ciunga.it/jxspeccy>

DesktopEmulators

XZX is silent unless you indicate **-audio 1** for output to /dev/dsp. By default emulation is normally limited to 100 per cent of original speed, but it can go 12 times faster is with the command line parameter **-fastmode 1** to remove the brakes. XZX and X128 offer more to former Spectrum programmers than XSpecEmu, but demand Unix as well as Spectrum expertise. Both benefit from MIT-SHM extensions, standard in XFree86, to boost X redraw speed.

The rest

XZ80 was written by Ian Collier, a guru of the comp.sys.sinclair newsgroup, but stalled at version 0.1. The core is well programmed, including clever ROM patches, but XZ80 has rather slow sound and ZX printer emulation, and also lacks support for later hardware like disk drives or memory expansion.

Fuse is written by Philip Kendall, maintainer of the Spectrum FAQ, but he's not generous to it. The current version (0.3) has gained improvements in sound and is slowly losing dependence on the Allegro toolkit, but it's still rather crude and a CPU hog.

Also-rans include the officially-titled 'unnamed' Sinclair Spectrum emulator by Thomas Harte, which is as minimalist as its name suggests, Jean-Francois Lozevis's hard-coded x86 Spectrum 48K emulator, and zxsp-x, a Mac port by Günter Woik which uses the Qt toolkit.

Java options

Spectrum emulators written in Java are neat hacks as they run on any Java-enabled machine, not just Unix systems, but the extra overhead makes them relatively slow, even though they only attempt 48K Spectrum emulation.

Hob and Jasper are both written and actively supported in the UK. Hob may be the one to watch, but is so far only available to run on-line, and lacks sound. For the time being Jasper is more complete but only manages two thirds of the speed of a real Spectrum on an 500MHz AMD K6 running Netscape 4.72. You'll need a fast processor, or a compiling virtual machine, and preferably both, to make good use of this, but it does run nicely on the Linux-hosted Amiga SDK.

The programmer of /Java Spectrum/ admits it is 'not very good yet' – it runs some programs but not all, and is a Java application not an applet so it needs a full Java Development Kit, not just a browser. The author calls it a 'neat toy' and you can get the source if you want to take it further.

File formats

You need to understand emulator file formats to use the thousand of ZX files on the net. The original snapshot standard was .SNA format, popularised by Arnt Gulbrandsen's JPP emulator, named after the Sinclair tape-loading key sequence: J for SAVE and Shift P for a pair of quotes.

SNA format is based on the files generated by a Mirage Microdriver, a real Spectrum add-on with a 'magic button' that saved uncompressed 48K memory images and processor registers in a 49179 byte file. More common nowadays is .Z80 format, from the eponymous MSDOS emulator by Gerton Lunter. This supports data compression and 128K images with sound chip data, but the extensions mean that there are several variants of .Z80 format and compatibility is not assured. SPCONV converts between Z80 and SNA formats, and sometimes fixes snapshot bugs in the process.

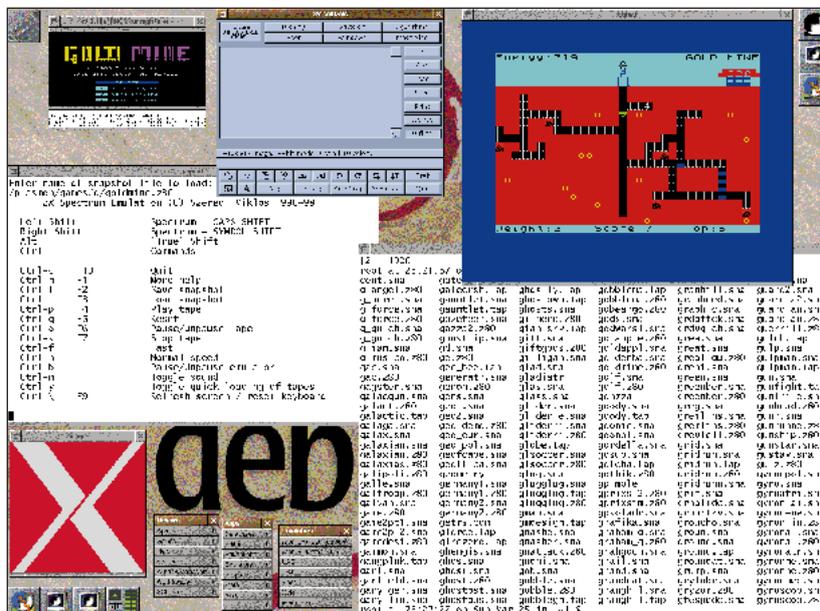
TAP is another emulation file format, designed for programs that load blocks from tape in several parts, merged into one host file. TZX is a lower-level tape format, essentially a sample of the

original audio. TZX is more verbose but more flexible than snapshots or TAPs; it can encode 'protected' data which would require a non-standard loader, though these are a severe test of emulator compatibility. VOC is an alternative raw sample format. Fixed-sized 'image' files can correspond to the contents of an add-on drive for the original Spectrum. Like TAP files these allow multi-part loading, and contain hidden directory data that Linux cannot easily store, making them more compatible with special files and programs that perform random access.

You can move files to and from these images with emulators or standalone utilities. Linux device-independence means that as long as the disk and sector sizes are appropriate you can directly transfer image files between Spectrum floppy disks and Linux devices. Don't expect this to work with microdrives, though! If all else fails, use a 9600 baud serial link to a Spectrum 128.

Emulator fuel

Spectrum emulators are widely available online and come complete with the vital ROM files, thanks to the generosity of Amstrad. We've put enough to get you going and ZX-related



utilities for Linux on the cover CD, but could have filled the whole disk and more besides with Spectrum files if we'd been willing to push everything else aside.

Instead we've listed links, most notably Planet Sinclair and World of Spectrum, and checked out CD compilations which address the popularity of ZX emulation on many platforms. If you don't fancy trawling through thousands of snapshot files on line, Weird Science (www.weirdscience.co.uk) have a couple of budget CDs which could save you time and money too at £4.99, unless your Internet connection is uncommonly quick and cheap.

Retro Gold is a combination Spectrum and C64 emulation compilation – it doesn't include any emulators specifically for Linux, but does contain thousands of well-known Spectrum game snapshots, and some ZX applications too, for programming, graphics, education, utilities, business and adventure-writing.

Speccy 99 is a Sinclair-specific compilation CD with many of the same snapshot files, hundreds of related documents, and some Linux utilities. Updates of this compilation were an annual event for much of the 1990s, and previous discs have featured similar ZX files and emulators like XZ80. **LXF**

XSpecEmu was relatively easy to install on a Debian system.

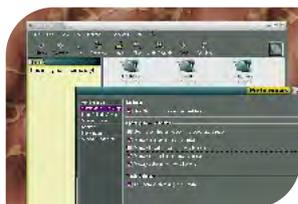
Next month

The Spectrum's arch-rival was Commodore's 64. Again it was a refinement of earlier hit systems, from the PET through the VIC-20, and sold millions worldwide.

LINUXformat Desktop

All you need for configuring, using and customising your desktop

WHAT'S ON YOUR DESKTOP THIS MONTH...



Reviews: Rune and Nautilus get the LXF treatment **50**



Winmodems that will work! **54**



Emulators explored – speccy magic **56**

Windows emulator update

Win4Lin 3.0

Those of who still feel the need to cling to Windows for some legacy applications may be relieved to hear that the Windows emulator, *Win4Lin*, is due for a radical update in the near future.

The software emulates a Windows environment, so that applications such as Microsoft *Office*, various accounts packages and other software can be run through Linux without the need to reboot into Windows. Our review back in issue 11 concluded that version 2.4 of *Win4Lin* was reasonably fast and capable of running a wide range of applications, but that it had some drawbacks in terms of ease of use.

Susan James of NeTraverse told us: "Most of the limitations discussed in your coverage of the previous version are solved in release 3.0 of the software. And there is a new graphical installer which automates the initial installation and searches for updates from our web site."

The boxed version is due for release



The new *Win4Lin* 3.0 promises better installation and ease-of-use.

in May, but you will also be able to purchase a download-only edition of the latest release direct from

NeTraverse's website, probably by the time you read this.
www.netraverse.com

Upcoming Linux events in the UK.

Not one, but two shows!

Linux users in the UK will have the choice of two venues when it comes to Linux exhibitions this year. The first of these will be the rescheduled Linux Expo, organised by IT events, which will now take place in London in July.

But this year another UK Linux exhibition is to be held in Birmingham, in September. This second event is also called Linux Expo, but it will be organised by the company that is responsible for the successful Linux Expo shows in mainland Europe (in Paris, France, and in Germany, etc.)

Both events are promising a spectacular Linux experience and will feature a huge range of exhibitors, as well as full conference sessions.

We'll be bringing you more news of

the planned events and exhibitors for both these shows in the coming months, so please check back here often, and visit our website at www.linuxformat.co.uk



The *Linux Format* crew pose for the camera at last year's Linux Expo.

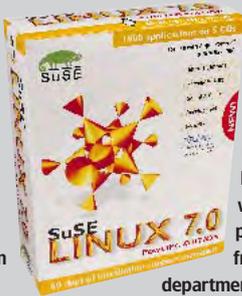
Comment

Problems with SuSE PPC to be fixed in 7.1

SuSE PPC back on track

Following our less than gushing review of SuSE 7.0 PPC, we got a reply from James Ogley at SuSE UK. We pointed out that the "Office" option didn't install all the software it claimed to, and that X configuration was tricky and didn't even work on some iMacs.

James told us: "The Default Installation with Office option was a mistake, a hangover from the i386 version. In the i386 version, this option installs *Star Office* along with the rest of the default installation, but since there is not a version of *Star Office* for Linux PPC this was not possible. This option



will be removed in 7.1.

"The problems you had with X are noted. It was known that *SaX2* only worked with a small number of PPC machines. For other machines there were XF86Config templates provided on the CD. Judging from our support

department's observations, most users got by quite well with this arrangement. The issue with bit depths however was confusing in the 7.0 manual and has been resolved for 7.1."

Other problems we mentioned with sound and security are covered by enhancements in SuSe 7.1 PPC.

www.suse.com

Browser wars good

The desktop Linux user is certainly benefiting from the browser wars. A year ago if you wanted a browser for general surfing that wouldn't fall over when it came across some Java, you were pretty much limited to the shaky and unreliable *Netscape 4.7*.

Increased competition has resulted in Linux now supporting a gaggle of fairly solid browsers. *Netscape 6*, *Opera*, *Konqueror* and *Mozilla* have all advanced faster than can be believed.

A survey on the *Linux Format* website found that *Konqueror* is the most popular Linux browser, but not by much. With over 450 votes cast in a little over two weeks, just over 27.5% of respondents chose KDE's browser as their favourite. *Opera* was just behind with 24.5%. This is an impressive result since it is still in beta (see the preview on page 26) and has only recently been released in a "free to download" form. *Netscape 6* polled a depressing 6% of the vote, amazingly losing out to *Netscape 4.7* (11%) and *Mozilla* (18.5%).

One random respondent, Peter James, was quizzed on his choice: "I had been using *Konqueror* since it



Konqueror is tops with LXF readers.



But Mozilla still has a way to go.

came out with *KDE 2.0*. I occasionally use *Netscape* as well, but *Opera* is my favourite. It still has problems with some sites, but it seems so much faster, it's like I've got a faster modem."

Other comments on the site included complaints about *Mozilla* being slow and bloated and *Konqueror* hogging resources. To participate in further surveys, cruise on over to www.linuxformat.co.uk

Installation now

If there's one thing we get more queries about than anything, it's installing software. Particularly for users new to Linux, it seems a rather complicated task. But it shouldn't be.

Every month we run the usual './configure, make, make install' instructions, and I've lost count of the number of times we've explained how to install RPMs. But, though with a bit of knowledge and practice even the most reluctant beginner can get the hang of it, is this what we should be expecting from users?

In the old days of Unix systems, building software was much harder. There was much manual fiddling around with makefiles to ensure the software you were building was being compiled against the relevant libraries on your system. The advent of *autoconf* and *automake* simplified this rather tedious process, without compromising the functionality of the software concerned. Can't we create an *auto-install* as well?

It would be unfortunate to move towards everybody creating different installers for their software. It would be much more useful to develop a widely accepted GUI-based installation standard, but very little work is being done on this. There are a few projects scattered here and there, but they rarely get the support they deserve and require. Sadly, as these tools don't make it easier to create software (as *autoconf* and *automake* do), there is no great incentive for developers to use them.

But what's the point of developing an easy to use mail program, if your target audience isn't able to install it?

Linux distributions, even the server-oriented Red Hat, have moved towards graphical installation; some, like SuSE and Mandrake, have even made installing Linux itself a relatively easy process. Don't graphical applications deserve the same treatment?



Nick Veitch

Is the Editor of *Linux Format*, and runs Linux on every desktop machine he owns.

Reviews **DesktopRune**

AXE 'EM UP

Rune

Watch out! There's a whole load of bearded, beery, belligerent battling boyos heading towards the Linux scene. The Vikings are coming! Paul Cavanagh previews Loki's next marvel.

When some bright spark at Loki came up with the idea of porting *Rune* to Linux, he/she was onto a good thing. For one thing, *Rune* utilises the *Unreal Tournament* engine, and that's one of the few up-to-date engines that Loki haven't yet managed to bring to Linux. Expect to see a heck of a lot of ported first/third person shooters being welcomed into the pantheon of Linux games shortly.

This might make decent business sense on its own, but when one of the characters in the game, better still a godly character, carries the same name as your company, you're definitely onto a winner. Loki, in case you didn't know, is a Scandinavian god who knocked about with Odin, Thor and chums. Apparently Loki was one of your nasty deities, and somehow managed to wind up the other gods so much that he got cursed, and ended

people of that region worshiped, Viking gods and warriors were the most esteemed members of society. Ragnar lives with an Odin-worshipping clan of warriors and has recently been praised by the village elders. They have decreed that he must make his rites of passage and become a warrior. As the fates would have it, at this precise time there's a distinct feeling of doom in the air, and some of the other tribes have taken to serious monkeying about involving your common-or-garden pillaging. More than that, though, reports claim that said hairy Spam-eaters have been destroying the magical rune-stones scattered about the lands. Ragnar's village houses one such rune-stone, which the villagers worship and are sworn to protect. Further, legend has it that if the stones are destroyed, Loki will be free from his torment, and the time of the end will be at hand. Phew! Frankly, plotlines



Honey! I'm home! Time to sit by the fire, drink some ale and have the missus prepare Spam, Spam, Spam, egg, Spam and Spam. With the Spam.

playing the Beta version of the game – so it would be unfair to pass any judgement here, you'll just have to wait for the full review. What I can say with certainty is that *Rune* certainly looks beautiful, with splendid sky effects, some nice rippling and splashing effects where water comes into play, and a well detailed, easily controlled playing character in Ragnar. The lack of ranged weapons could be a disappointment to seasoned fraggers who like to snipe from dark corners, but Vikings were never particularly

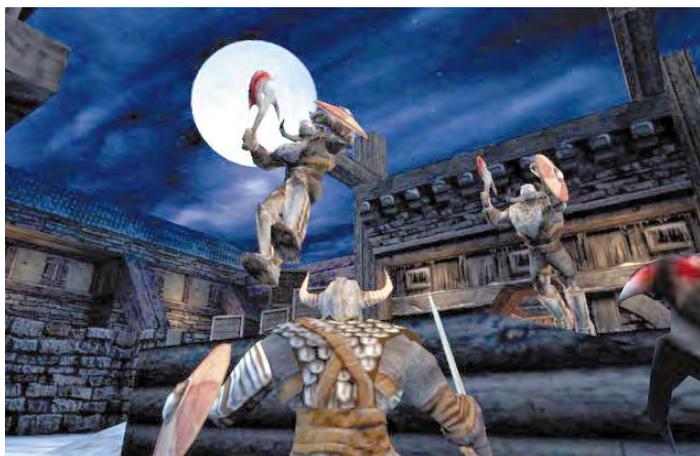
noted for their subtlety or stealth. For full-on melee action nutters, there'll be plenty of fun to be had. All of the weapons available can be used in a variety of ways, with timed multiple button presses resulting in combination attacks. Additionally, if you're running forward during an attack your attack will be more vicious, while if you

move backwards you'll perform a more defensive blow. Having the gods on your side can help too, as weapons can be magically charged adding yet more variety to your arsenal. Your Viking chum can also use a shield to defend himself, so a complex combat style can be developed over time. *Rune* comes with an online multi-player version, so players can form their own Nordic horde and start bashing up anyone who dares look at them in a funny way or spills their pint of mead.

So Viking lovers everywhere prepare! Get your horny helmet, crack open the Spam, pick the hoppy bits out of your ale and prepare to trash the neighbourhood. On the subject of food and drink, you get health points for drinking strong ale (something that I for one heartily approve of) but there's sadly no Spam to be found – these Vikings get their energy from poncey bits of fruit. An opportunity missed, methinks. But no matter! Those of you into hirsute smelly savages will be in seventh Valhalla. **LX**

Linux Format **VERDICT**

Rune looks gorgeous and its use of the UT engine could usher in a whole new generation of Linux games.



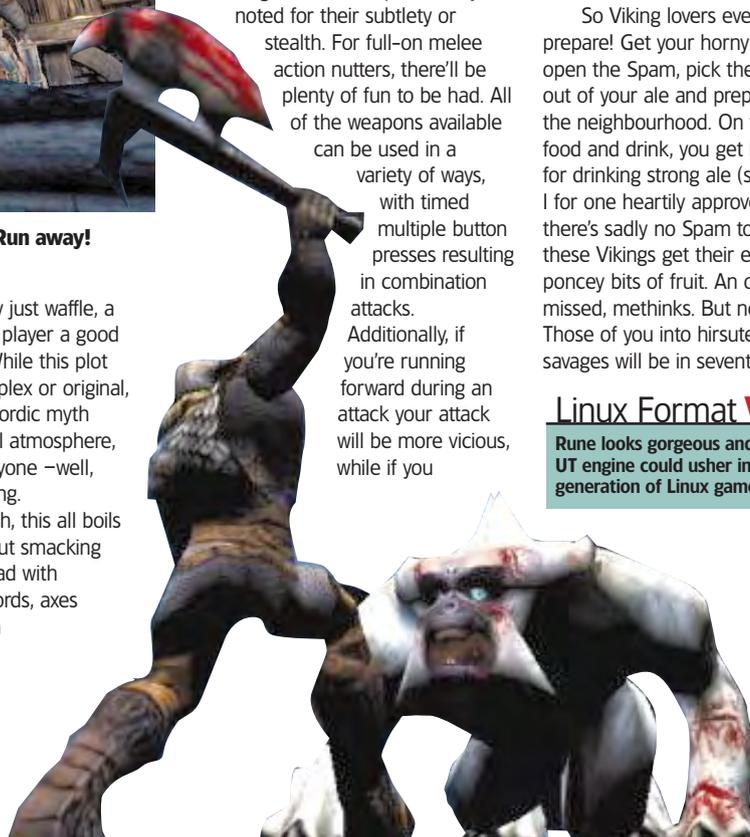
Heek! Hairy axe-wielding types descending from the sky. Run away! Run away! Run away!

up being spat at by some dirty great snake for most of eternity. The snake spit was acidic and burned away his whole chest, but as an immortal he was blessed with healing powers. So, a whole eternity of corrosion and correction. Poor chap.

So, anyway, back to the game. *Rune* is a third-person action/adventure game set in the lands of ancient Scandinavia. Way back before the Krona became a prohibitively expensive currency the

for games are usually just waffle, a device that gives the player a good excuse for a scrap. While this plot isn't particularly complex or original, the involvement of Nordic myth gives the game a real atmosphere, and let's face it, everyone – well, almost – loves a Viking.

Obviously enough, this all boils down to running about smacking people about the head with weapons such as swords, axes and maces. I've been



Reviews Desktop Nautilus

WEB BROWSER/FILE MANAGER

Nautilus 1.0

■ **LICENSE** Open source ■ **MAINTAINER** Eazel ■ **WEB** www.eazel.com

Richard Drummond examines Eazel's next gen browser for the GNOME desktop.

The GNOME desktop has been significantly lagging behind its rival, KDE, in terms of the level of integration of software and functionality ever since the release of KDE 2.0. The application that will make up the difference is *Nautilus*, Eazel's file manager-cum-browser. And unlike KDE, Eazel are not content to merely rehash the Windows look-and-feel with their browser, but are producing a truly innovative product. The first stable version of *Nautilus* has just been released, so we thought we'd give it a test drive.

Eazel does it

Nautilus has been designed – by the people behind the original MacOS no less – to make the Unix desktop accessible to new users – but remain powerful enough to be flexible for experienced users. One way that the designers have accomplished this is by implementing user experience levels. The first time you run *Nautilus* (always supposing you can manage to install it in the first place), it asks you to choose a level. Beginners and intermediate level users are shielded from some of the more confusing or dangerous functions within *Nautilus*. You can of course change your user level later on.

Once *Nautilus* proper is up and running the first thing you'll notice is how much attention has been paid to visuals. A *Nautilus* desktop is simply gorgeous; KDE 2's standard theme looks stark and clinical in comparison. The overall impression is colourful with a chunky, solid feel, cartoon-like icons and smooth fonts – it's obviously intended to appear warm and friendly to new users. Not surprisingly, *Nautilus* is completely themeable and ships with a selection of pre-defined looks.

The browser

A *Nautilus* window is simply a shell where different browsing components can be embedded to perform different operations or view the various types of



Nautilus is no slouch when it comes to good looks. If only it was a little bit more well behaved.

file you'd expect to encounter. Components supplied in this release include the file manager component for, well, managing files, a web-browser (which uses Mozilla's Gecko engine), viewers for text and images, and a novel MP3 player.

Every window additionally has a sidebar, which can be hidden if desired. This feels somewhat like the Mozilla sidebar, and, again, various components can be shown here. By default it displays information which is context-sensitive to the main browser's contents. When using the file manager, for instance, it gives details on the current directory; when viewing the Trashcan it adds an 'Empty trash' button here. The user can switch the contents of the sidebar with the tab gadgets situated at its base. Options include a tree view to navigate your

filesystem, a table of contents for the GNOME help system, a history browser – or uniquely – a notepad where you can scribble down notes to be associated with the current contents of the browser.

The *Nautilus* file manager is perhaps the most significant browser component. It firmly breaks the Windows mold, and sports some unique features. For example, you can change the viewing settings for each directory and the file manager will remember these for the next time you visit. Icons can be ordered in the browser by name or date or whatever, or you can position icons arbitrarily in a custom layout and the file manager will remember that, too. Not surprisingly, you can do the usual view as icons to get a pictorial view of your directory contents or view as a list. A

unique viewing control in *Nautilus*, however, is a Zoom level – which not only controls the size at which information is displayed in the file manager but the amount that is displayed. At the lowest magnification, each file is represented by icon only; at the highest magnification, you are told its name, size, modification date and type. Not only can you annotate directories with the sidebar notepad mentioned above, but you can add 'Emblems' to a file's icon in a similar manner to MacOS's labelling facility. Available emblems include things like "New", "Draft" and "Urgent" and serve to give a visual clue to a file's status. Another neat trick in the *Nautilus* file manager is the previewing of files (an idea that has since been copied in KDE 2.1). For text files and images, a thumbnail preview of a file's contents

can be generated and displayed as that file's icon. You can preview sound files, too: leaving the mouse pointer over a sound file's icon will play it. This feature is great if you have hundreds of sound files littering your hard drive with useful names like 'song1.mp3'.

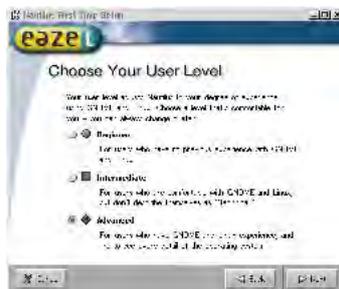
Eazel services

One of the components which Eazel hopes will appeal to *Nautilus* users – and give them a revenue stream to boot – is Eazel services. *Nautilus* users (and their should be plenty of them as the company have signed deals with the likes of Sun, Red Hat and Dell) can take advantage of two services at present: Software Catalogue and Online Storage.

Software Catalogue is a collection of commercial and open source software for Linux which, when integrated with *Nautilus*, should take the headache out of getting software onto your system. The aim is to offer one click installation.

The catalogue currently offers well in excess of 1,500 applications including staples such as *GIMP*, *Gabber*, *Netscape 6* and *GnuCash* as well as more esoteric fare such as *gPhoto* and *Gnomehack*.

The *Nautilus* online storage works in a similar way to offerings from the likes of X-Drive and I-Drive, but integrates fully with the Linux user's desktop. The system allows you to treat your free 25Mb of online storage



Nautilus is good at hiding dangerous options from new users.



Installation is easy... if you're running Red Hat 6/7.

Getting Nautilus

You can download *Nautilus* from Eazel's website, but depending on which Linux distro you are running things are not always that straightforward. If you have Red Hat 6.x or 7.0, then you're laughing. Eazel only support Red Hat at the moment and they provide a graphical installer to fetch and install the correct RPMs for your system automatically.

Life isn't so easy for the rest of us. *Nautilus* is very fussy about its dependencies. You have to have the correct versions of a host of different *GNOME* components and other libraries – and these at newer versions than most users will currently have installed. The safest bet is to install the latest *GNOME*

build from Ximian, download the source code for the *Nautilus* components and build it yourself. Instructions are provided on the Eazel site.

Users who don't want all the hassle can wait for other builds of *Nautilus* to appear. *GNOME* are supplying *Nautilus* with the soon-to-be-released *GNOME 1.4* and they support more distros than Eazel do. *Nautilus* will also be available with Mandrake 8.0 when it's when released; braver users can get it from Mandrake's Cooker and try it now. Debian users who are tracking woody can also get *Nautilus* as DEB packages from their local mirror, but Debian's build is still very flaky; expect to do some hacking to get it to work fully.



The Nautilus preferences dialog doesn't give you many options to change.

as if it were a local drive. It also offers the advantage of making your stuff available (to you) from virtually any 'net connected computer. This makes it ideal, for instance, for transporting stuff from home to work without having to carry around breakable, loseable zip disks.

Both of these services would be a valuable incentive to owning *Nautilus*, unfortunately my attempts to register for them failed due to fact that my email address was too long!

Early days

The 1.0 release of *Nautilus* is a comfortable file manager, and has innovative touches, but there is still much work to be done. It works well

as a file manager, but it feels rather spartan compared to the flexibility of *KDE's Konqueror*. Those used to the Windows and *KDE* metaphor of using a clipboard for copying, cutting and pasting files may find that *Nautilus's* drag-and-drop only method is unnecessarily cumbersome. Something else that is missing is the ability to view file archives as if they were a regular filesystem. *GNOME's* virtual filesystem component supports this – in fact it was included in the old GMC file manager – so its absence from *Nautilus* is as puzzling as it is annoying. *Nautilus's* web browser, too, is particularly primitive. It does a good job of rendering pages – it's uses *Mozilla's* excellent rendering engine

after all – but it's just a basic browser. You get no control over how pages are displayed and there's none of the pop-up menus that most browsers provide where you can choose operations that you can perform on a link or image. The only preferences option that applies to the web browser is the ability to set an HTTP proxy. In fact, the preferences editor as a whole is very simple. It affords the user a little slack to play with the operation of *Nautilus*, but nothing like the configurability that *KDE* has to offer.

More worrying perhaps than the missing functionality is the fact that the current release of *Nautilus* is big, memory-hungry and has a couple of stability problems that need to be addressed. Such worries are premature, though: *Nautilus* is still a young product. True, at the moment it is only really useable as a file manager, but it does that job very competently (once you turn off all the frills). As for issues of speed and stability, once real users start exercising it properly then bugs should be more readily identified and fixed. And once the bugs are fixed, the developers can get on with optimisations that are needed to turn *Nautilus* into the world-class product it so obviously has the potential to become. Stability will also improve when the various components and technologies it depends on, like *Mozilla*, reach a stable status.

If you fancy a change of desktop, then right now *Nautilus* is well worth experimenting with. It is not yet on a par with *Konqueror* and is not stable enough to be a full-time replacement for *GMC*. Hopefully, the *GNOME 1.4* release – which happened too late for the review – will address some of the problems, but I suspect we will have to wait for 2.0 for it really excel. [LXF](#)

Linux Format VERDICT

Installation	5/10
Ease of use	9/10
Features	7/10
Performance	6/10

Nautilus is ease-to-use, extremely good looking and has some innovative features, but performance and stability are disappointing at present.

LinuxFormatRating
 6/10

WINMODEMS FOR LINUX

Driving the LTmodem under Linux

Truculence is a trait that Linux users have come to expect in internal modems. But don't despair, **Richard Drummond** has a few aces up his sleeve.

One problem for the average home user who wants to install Linux is that most off-the-shelf PCs ship with a so-called 'WinModem' card, a type of device which isn't supported by a driver in the standard Linux kernel. Don't despair, though. This doesn't necessarily mean that you have to throw away your modem card and buy an external model. Various projects exist that are trying to implement drivers for a range of Winmodem chipsets. This tutorial covers one of the best supported of these, the Lucent DSP or LTmodem family, a chipset family used in a wide range of ISA and PCI modem cards (from vendors such as Xircom and Zoom) and which is often found built-in on the motherboards of various desktop and laptop machines.

A number of different Linux drivers for the LTmodem are available on the web, but the best solution is a half open-source and half closed-source driver derived from code written by

Lucent and originally ported to Linux for Red Hat. Binary-only versions of this driver exist for old kernels, but avoid these. The current maintainers supply Lucent's object code with the necessary glue code to build a driver for any recent 2.2 or 2.4 series kernel. We have supplied the latest version of this driver kit on this issue's coverdisc, but you should check <http://walbran.org/sean/linux/stodolsk/> for the latest updates and information. It is a good idea to read this and the Linmodem HOWTO before going any further.

Check your modem

The next thing to do is to verify that you do in fact have a Lucent DSP modem. This is not always easy and the fact the chipsets have gone under various names doesn't help. The ISA LTmodem family includes the DSP1641B, DSP1643 and DSP1644 versions of the chip and was often called the Apollo family. The PCI model is the DSP1645 and was sometimes called the Mars chipset. If the worst comes to the worst you can pull out the card and physically look for one of these chip designations.

If you have a PCI device, then Linux can help you identify the card. Enter the command:

```
lspci -v
```

to show what's connected to your PCI bus. You are looking for an entry similar to this:

```
00:0a.0 Communication controller: Lucent Microelectronics
56k WinModem (rev 01)
    Subsystem: Lucent Microelectronics LT WinModem
    56k
    Data+Fax+Voice+Dsvd
    Flags: bus master, medium devsel, latency 0, IRQ 12
    Memory at da800000 (32-bit, non-prefetchable)
    [size=256]
    I/O ports at b800 [size=8]
    I/O ports at b400 [size=256]
    Capabilities: [f8] Power Management version 2
```

The pair of numbers at the beginning of the entry specify where the card is installed in your machine. Above, we have "00:0a" which says it's card number 10 (0a in hexadecimal) on bus 0. Now enter:

```
lspci -n
```

and look for the entry that corresponds to your modem's card number. It will be something like this:

```
00:0a.0 Class 0780: 11c1:0440 (rev 01)
```

Ignore the class field and look at the next pair of numbers '11c1:0440'. The first of these is your card's vendor ID and the

Table 1

Vendor	Vendor ID	Device ID range
Lucent	11c1	0440 - 045c
Xircom	115d	0000 - 000f
Xircom	115d	0440 - 045c
Xircom	115d	0010 - 03ff

What's a Winmodem?

A modem is a device for converting digital data to an audio signal (and vice-versa) suitable for transmission over an analog phone line. The traditional type of modem, the hardware modem, is an intelligent device. The host computer sends and receives digital data to and from the modem and the modem takes care of all the signal processing. At the other end of the spectrum is the software modem, a device where all the modulation and demodulation of the signal is performed entirely in software. Somewhere between these two poles falls the Winmodem. The term Winmodem is often used generically to apply to any type of modem in which the work of the modem is shared

between the host computer's CPU and, usually, a hardware DSP engine (Digital Signal Processor). These devices are variously known as DSP modems, controller-less modem, host-controlled modems are any of a dozen other names. The term 'Winmodem' is actually a registered trademark of US Robotics, Inc. (now owned by 3COM).

Winmodems are popular with PC vendors because they are cheap to manufacture and easy to upgrade. The problem for Linux users is that the developers of the various Winmodem chipsets are closed-mouthed about how to program their hardware. Hence, Linux drivers for Winmodems have been slow to appear.

second is the card's Device ID and together they uniquely identify that model of card. Table 1 lists the ranges of IDs supported by the current Ltmodesm driver. Check that your card's ID falls somewhere within these ranges.

If your card ID happens to lie outside of these ranges, it may still work with the current driver, but you will need to specify its ID to **modprobe** when driver module is inserted. We'll tell you how to do this in a moment.

You can also get some help identifying ISA and non-PCI on-board devices using the **pnpdump** command. This scans for plug-and-play information and dumps it the standard output. Enter **pnpdump** and look through the listing produced and watch for an entry corresponding to your modem. We don't have access to an ISA modem here at *Linux Format* to test this with, but the following output, taken from the Linmodem HOWTO, was generated on a Thinkpad i1411 with a Lucent Ltmodesm. You should get something similar for your ISA modem.

```
(CONFIGURE ACRd119/1 (LD 0
(INT 0 (IRQ 11 (MODE +E)))
(IO 1 (SIZE 8) (BASE 0x0100) (CHECK))
(NAME "ACRd119/1[0]{LT Win Modem }")
# (ACT Y)
))
```

To take advantage of PNP configuration, you'll need to add this entry to your `/etc/isapnp.conf` file. For more information see the Plug-and-Play HOWTO.

Building the driver

The Ltmodesm driver is a kernel module that must be compiled against your particular kernel version, and to do this you need the kernel header files corresponding to your kernel. If you are using a stock kernel from your distro, then the easy way it to install the appropriate kernel-headers package. Generally, this will already have been done for you. Typically, the headers will be installed in the directory `/usr/src/kernel-headers-2.x.y`, where `x` and `y` are the major and minor version numbers corresponding to your kernel version. Make sure there's a symbolic link from `/usr/src/linux` to the directory where your kernel headers are installed. If it doesn't already exist, make it with, for example:

```
ln -s /usr/src/kernel-headers-2.2.19 /usr/src/linux/
```

If such headers aren't available – perhaps you built and installed a custom kernel image – then you'll need the full kernel source archive corresponding to your kernel version. Again make sure that `/usr/src/linux` points to, this time, your kernel source archive. You must have at least performed a **make dep** in the source tree, because this is the stage that sets up the links to the required header files for your kernel build.

Okay, now unpack the Ltmodesm archive and CD to the directory created. If you don't mind logging in as root, then `/usr/src` is a typical location for building kernel modules. You'll need root access to install the module anyway.

To do this, enter:

```
cd /usr/src/
tar zxvf ltmodesm-5.78e.tar.gz
cd ltmodesm-5.78e.tar.gz
```

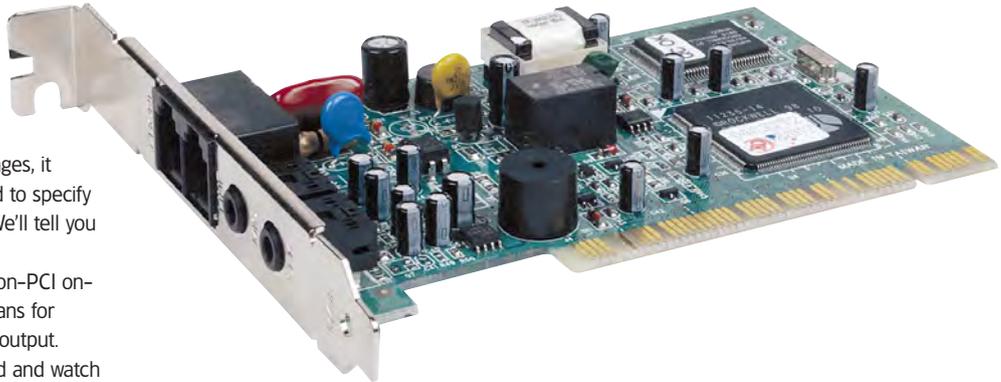
The Ltmodesm driver is now built by calling:

```
./build_module
```

and installed with:

```
./ltinst
```

This installs the module into the relevant place in your kernel module repository (`/lib/modules/2.x.y/`), sets up a device file called `/dev/ttyLTO` so that programs can communicate with it and links



Web resources

<http://linmodems.org/>

Links to information on modem chipsets and availability of Linux drivers.

Agere Systems (formerly Lucent Micro) home page.

<http://www.idir.net/~gromitkc/winmodem.html>
Database of modem compatibility with Linux.

<http://www.close.u-net.com/ltmodem.html>
Project to create truly open source driver for Lucent modems.

<http://walbran.org/sean/linux/stodolsk/Linmodem/resources>.

<http://808hi.com/56k/ltwin3.htm>
Everything you wanted to know about Lucent modems but were afraid to ask.

<http://www.lucent.com/micro/K56flex/doc.html>

`/dev/modem` to this device file. It also adds an entry to your `/etc/modules.conf` file (or `/etc/modutils/aliases` if your system uses `modutils`), so that the driver can be loaded on demand.

The Ltmodesm driver works by creating a virtual serial port – accessed via `/dev/ttyLTO` – which your Internet dialler or comms software can use to communicate with your Ltmodesm just as if you had a real serial modem attached at that port. You will have to ensure that any software that you want to use your Lucent modem does so through `/dev/ttyLTO` rather than `/dev/ttySO` or whatever it was using previously.

Problems

You should at this point have a functioning modem card. If not, there are some things you can try. First, test that the driver is being inserted correctly, by manually inserting it with:

```
modprobe ltmodesm
```

If this returns 'No such device' then the driver is failing to detect your modem card. Are you sure that you have an Ltmodesm card? If yours is a PCI card that has an unsupported device ID, then you may force the Ltmodesm driver to accept your ID. Enter the following line, substituting the values for `vendor_id` and `device_id` with values that you found above.

```
/modprobe ltmodesm vendor_id=0x11c1 device_id=0x0010
```

You can add these options to your `/etc/modules.conf` (or `/etc/modutils/aliases`) file after the `ltmodesm` driver entry that the installer inserted for you. For example:

```
alias char-major-62 ltmodesm
```

```
option ltmodesm vendor_id=0x11c1 device_id=0x0010
```

There is also a method by which you can force the driver to work with unsupported ISA chipsets. For more information see the project website. If insertion of the Ltmodesm driver is trouble free, but you get "No dial tone" when dialling out, then try changing your BIOS settings from 'PNP OS' or 'Windows' to 'non-PNP OS' or 'Other OS'. For any other difficulties please consult the project homepage. Remember that if you manage to get the driver working with a currently unsupported card, please get in touch with the developers and let them know. **LXF**

What on Earth is... FREENET?

Find out what new peer-to-peer networking systems mean to you, your freedom and your files.

» Well, you can guess my first question – What is Freenet? It sounds like an ISP or something...

No. Freenet is a peer-to-peer networking system. It uses a system of nodes around the world to store information, allowing those accessing it to search for files, publish documents or even have a poke around private websites.

» Isn't this just like the Internet/WorldWide Web?

Not really. The WWW is based on a client-server model, where people go to predefined sites for information. A peer-to-peer (p2p) network allows individuals computers to be connected directly. For

example, if I wanted to transfer a file to you using the web, I would have to upload it to an ftp site or post it on a website. Then you would have to connect to the same site to download it – both of us would be clients to a server located on the network somewhere. Using a p2p network, I would connect directly to your computer, and send you the file directly. The advantages of p2p networking is that it cuts out the middleman. As well as being simpler (in some ways) it also makes it a lot harder for a third party to control the information that passes between any two Freenet users.

The entire contents of Freenet are distributed variously throughout its nodes, making it possible to retrieve any information through the nodes you are

connected to. You can also create your own node and register it, in order to make your own files available to other users.

» Okay, now this sounds a bit like Napster/Gnutella...

Closer, but still no cigar. The entities you mention, and some others like Scour, are peer-to-peer networks, but Freenet offers some unique advantages like more security of information and (allegedly) a more efficient delivery system. When Napster went "legit" the corresponding huge increase in traffic on the gnutella network caused virtual gridlock, as clients were receiving more requests than they could handle on the upstream from a modem. The result was a



fragmentation of the network. Freenet *should* scale better, although this is difficult to test. Because of the nature of the network's privacy regime, it is difficult to tell how many nodes there are at any given time, or how many users are using them.

»» **But is it a network for distributing pirated music and so on?**

No. It is a network for distributing whatever to whomever. It would be possible to distribute pirated MP3 files, and loads more documents, images and data of dubious origin, it is also possible to publish documents free from other forms of censorship, and to do so with anonymity. Or indeed, to read what you want, without people knowing who or where you are.

Now, we're not making any judgements on this ourselves, but the people behind Freenet have made various statements on their websites and in other places, which basically centre around the idea that in the Internet age you can't have freedom of speech and an enforceable copyright system.

Basically, their philosophy seems to be that their system is designed to make free speech possible, without the interference of censorship of any kind. In order to do so effectively, the consequence is that copyrighted material can also be "traded" over the system with impunity.

We don't care whether you buy into this or not. If you have any opinions, why not write in to the mag, or contact the Freenet team (freenet.sourceforge.net). We're just here to explain what Freenet is.

»» **Woah there! That sounds like a whole different legal can of worms. What's going to stop the MPAA, RIAA or whoever suing everybody involved, as they did with Napster?**

The Napster model of a p2p system still retained some sort of client-server relationships. Although

the individual files were stored on users' systems, rather than with Napster, the legal teams acting on behalf of the RIAA successfully argued that because Napster held indexes of these files on its server, though not the files itself, it was equally culpable of copyright violations.

The Freenet model doesn't have a central indexing system for files, or even any sort of 'centre', so won't fall prey to the same arguments. In fact, it is very difficult to determine where exactly files are held, or by who, because of the way the nodes connect, and the encryption of the contents.

»» **And how exactly is that?**

Okay, say you are looking for a file. You make the request from a Freenet node. If the node has the document you want, it sends it back to you. If it doesn't, it forwards the request onto another node (from a list of nodes it is aware of) and asks that for the document, and so on...

»» **Isn't that sort of random and inefficient?**

It would be, but I hadn't finished. The node determines which of the other nodes is most likely to have the document (because it remembers which other documents came from there) and sends the request to that node. This node then performs the same request procedure, and so a chain of nodes is followed until the requested document is found, or the request times out. This isn't foolproof, because it relies on similar documents being found on the same node, which obviously isn't always the case. This is one area of Freenet which is still being worked on.

»» **And so eventually, the document is found?**

Hopefully. And when it is, something else unusual

happens. Each node may (depending on a number of factors) cache the reply it received as it passes it back along the chain. Thus, the more often a document is requested, the more copies of it will be generated on the intervening nodes, making it easier and quicker to find. So, more popular documents become more widely available than less popular ones, which is what you would want.

»» **And this system is somehow more secure?**

Yes. Each node that receives a request only knows about the node that requested it, not about where the original request came from. This makes it impossible, simply by examining one node, to determine where requests for particular things have come from.

»» **But surely if you were out to sue someone, you could request a document from a node, and then sue the owner of the node, because the document came from there?**

Well, it would be legally dubious. You have no easy way of telling whether the information was there before you requested it – the mere act of requesting it can cause it to become available on that node, which probably falls foul of illegal search laws/entrapment or many other such regulations. Plus, amusingly, because of the caching system mentioned earlier, you would probably cause a more prolific distribution of the document.

»» **Presumably, you could find out if you really wanted to...**

Few systems are perfect. With a lot of time, effort, patience and some luck, you may be able to find out where an item was requested from, and where the document/file itself was being stored, but this would >>>



WhatOnEarth Freenet

require a great deal of resources. Also, there is the element of deniability.

You see, the thing is that, through a complicated system of requests and keys, the network decides which files are stored locally on a node. The person running the node doesn't even know what they are. As all files and communication is encrypted, it makes it unlikely that the owners of nodes would be able to determine what data was being transferred. One of the reasons for this is to make it impossible for node owners to censor documents, but another is that they can truthfully claim that they are merely facilitators of the traffic going through the node, and aren't responsible for the content – in much the same way that ISPs and the Post Office aren't held responsible for what people use their services for.

What about a deliberate attack on the network, like flooding the nodes with false data?

Not possible, because each item posted has its own key. Therefore, even if you generated a new file with the same name and loaded it on to the network, it would have a different key from the original.

Well okay then, how about creating a rogue node that deliberately distributed the wrong data when asked for a file?

Again, because of the key system, this is extremely difficult. Freenet uses three types of keyfiles. The easiest one for users to remember is the Keyword Signed Key. This uses a human readable form to describe a file located on Freenet, such as `text/books/mobydick.txt` or `music/mp3/elgar.mp3`. The client translates these into a KSK file which it uses for its request. The node processing the request doesn't know what the original string was, so it can't create a file that the requesting node or client would

accept. Of course, you could try and spend a lot of effort trying out common words to generate the appropriate key, but you are again looking at a huge amount of processing time.

So how do I go about searching for a file?

This is quite simple, you merely go to a Freenet node, type in the key data, and the network will endeavour to find the file for you.

And where do I get the key data from?

Ah, well at the moment, that bit is a little difficult. In order to find out the keys for documents, you'll have to find a list of key files. At the moment there are some websites listing some of the documents available. For instance "Steve's Key Index" (thalassocracy.org/keyindex/) has a long list of files, and a CGI driven search interface.

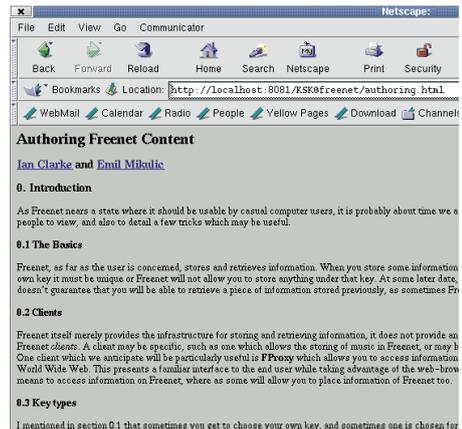
How do websites work on Freenet then?

Pretty well. All you need is to point your browser at the local proxy, and stick in the key for the main website page. Links on the page are actually Freenet keys, which can be fetched by your browser in the same way. It isn't lightning fast, but it works.

How do I get Freenet then?

You can download the latest code from the Freenet site on sourceforge (freenet.sourceforge.net). It is actually a collection of Java programs, so you will need some sort of Java runtime environment or JDK to run it. It seems to work with the major Linux Java distributions without any problems.

Freenet is available on the site in binary and source form, but as it's a java app, you don't really need the source .



Freenet isn't just about filesharing, you can also access complete websites.

And is it pretty easy to install?

Heh, heh. Not exactly. There is no installation script or anything for Linux. Unpack the archive and you'll just be left with a number of scripts and the main `freenet.jar` file. The first thing to do is run the configuration script, which is used to describe the basic setup characteristics for the Freenet server:

```
./freenet_config .freenetrc
```

then you can run the server with:

```
./freenet_server
```

What about making requests and so on?

If you want to grab files from Freenet, the easiest method is to use the built-in proxy server. Open up your favourite browser and type in the url:

http://localhost:8081

This will fire up the built-in server, which will let you enter requests for files, and also allow you to post information to the network. For more information on configuration, see the boxout on page 63 (Highly advisable!).



»» Hang on, I thought it ran on a random port for security purposes, so how come I can get to it via 8081 all the time?

The 8081 port is allocated to the proxy. This means that you can always access the Freenet server through that port. After all, it wouldn't be much good if it was so secure that even you couldn't get in.

However, the proxy is just a proxy. It knows the real port number the Freenet server is running on, and passes requests on. Because nobody knows the actual port number of the server, it makes it harder to attack it directly, or to prevent it from transferring data over networks. The only way to do this would be to follow some really Draconian policy of denying all ports except certain services you wished to permit, e.g. if you were behind a firewall.

»» When my node is up and running, will I be deluged with traffic?

Unlikely. Because of the way nodes communicate, as outlined before, nobody will know that your node exists when you first start it. You can choose to have your node listed on various "inform" pages after it has been running a certain amount of time. Also, by posting data, or requesting data from other nodes, they will gradually learn of your existence. But, if you aren't planning on running your node all the time, it is best not to broadcast your address, for the reasons outlined below.

»» The nodes I am connecting to never seem to be working...

Well, that's the problem with people who create nodes and get them listed, but then take them down again or aren't actually online most of the time.

If you haven't got a permanent connection, you want to list your node as "transient" in the config file and have it not list itself (see configuration boxout).

Configuration

There are two configuration files you need to be aware of when setting up Freenet, `.fproxyrc`, which holds configuration for the proxy server (which handles requests) and `.freetrc`, which holds the config information for the server itself.

The FProxy config file isn't too extensive, and there is only one setting you may need to change in it, right at the top of the file. By default this is:

```
serverAddress=127.0.0.1:19114
```

The last part of the number is the port number, which must match the port number defined in your `.freetrc` file

The best way to create `.freetrc` is by running the

Freenet config and answering the questions. Most things can be left at their default values but some will require tweaking to suit your circumstance:

```
listenPort=19114
```

This is determined randomly by the config script, but should match the port in `.fproxyrc`

```
nodeFile=nodes.config
```

Sets the filename of the list of nodes to try – it can be useful to store the locations of nodes you find fast/reliable in this file.

```
transient=no
```

If you are using a dial-up connection that isn't always on, you'll need to set this to "yes".

To solve your problem though, you can create a list of nodes that are reliable (check the `freenet.log` file and look for nodes that you have got a response from). This text file (node address on each line) should then be saved out (`nodes.config` is the default name) and identified in the config file with:

```
nodefile=nodes.config
```

»» There are some files in the .freetrc directory. What are they?

This is the directory where freenet stores a number of things by default. First of all there is the data shared by your node, and also files containing information about the data files being stored.

Also, freenet uses this space to store information about keys and other nodes.

»» How do I know what data is being stored on my node?

Weren't you paying attention earlier on? You don't – that's part of the system.

»» So how will I know if any illegal material is stored on my node?

You don't. If the thought of storing racist propaganda, illegal copies of Metallica songs or other such files bothers you, then you probably shouldn't be running a node.

»» Is Freenet going to be around longer than Scour etc?

Scour collapsed after it realised it would shortly face the same fate as Napster, i.e. being pursued through the courts relentlessly. The problem facing potential litigators regarding Freenet is "who do we sue". While in the case of Gnutella-type systems, they are likely to go after the high-usage targets, Freenet makes it rather difficult to ascertain who owns the nodes, where they are, what's stored on them and so on.

»» What am I likely to find in the realms of Freenet then?

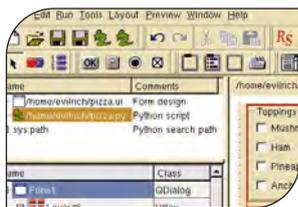
Well, sadly for all the talk of freedom of speech etc, about 99.9% of the material we found were copyright-busting music files. But do prepare to be shocked, there are some very disturbing things there too. If you are looking to offend yourself, something on Freenet will fit the bill. [LXF](#)



LINUXformat Developer

All the latest tools, info and advice for Linux hackers

WHAT'S DEVELOPING THIS MONTH...



Python tamed with
BlackAdder **66**



Using Perl to redirect
your Spam **68**



Building layout
managers in Java **73**

All change at the top

Changes at Debian

Ben Collins has topped the poll in the recent elections for a new leader for the Debian group. Ben, whose work on the Debain project has included maintaining the kernel for the distribution, beat three other candidates in the elections.

Linux Format contacted Ben about his visions for the future of Debian:

LXF: Is it true to say that you are adopting a “back to basics” policy? In your platform statements you seemed to be saying that Debian needed to get back to its roots and be about providing a stable Linux distribution that people would want to use.

BC I wouldn't really call it “back to the basics”, but more of a cleaning out the cruft policy. Right now we have a serious lack of control over our growth, both packages and developer membership. I intend to curve that in some form or another, without breaking down the structure we are all used to within Debian.



LXF What do you think the greatest challenge is for Debian during the next six months or so?

BC As always, our greatest challenge is releasing the next version. Since this is our major goal as a project, it is always the main thing we focus on. There are many things affecting this goal. We have a few new ports releasing with 2.3 (aka woody), which need boot support, and to be as stable as they possibly can (when potato released, it supported six architectures, slink supported four, and hamm, three).

Our latest developments within our archive management will help with some of these problems. The new “testing” distribution mechanism tries to make sure that only packages with less than critical bugs, and that have been in the distribution for set amount of time, will make it into the next release. This also helps to keep the separate architectures in sync with regard to which packages are available for them.

LXF What are your personal goals during your time of leadership?

BC Some of my main goals are to bring some sanity back to our organisational setup. Things like our relationship with Software in the Public Interest (SPI), which is Debian's non-profit legal face, and our internal structure have deteriorated over time. I want us to have a more active role with SPI, and a more recognised hierarchy within the separate projects within our main one.

www.debian.org

Transmeta's new distro

Midori released, other vendors unconcerned

You would think that Transmeta (better known for employing a certain Linus Trovalds, and creating the code-morphing Crusoe processor) releasing their own open sourced embedded Linux OS might cause a little consternation amongst the established embedded vendors, but according to a survey conducted by LinuxDevices (www.linuxdevices.com) that seems not to be the case.

Transmeta announced, at the time of releasing the GPLed Midori, that it was trying to make it easier for hardware manufacturers to build Internet and mobile devices based around Linux solutions. But few of the



big players in the Embedded Linux marketplace saw the move as a threat.

In fact, Lineo's Bryan Sparks actually welcomed the release suggesting it would open out some useful technology. See the survey at: <http://www.linuxdevices.com/articles/AT7124702616.html>

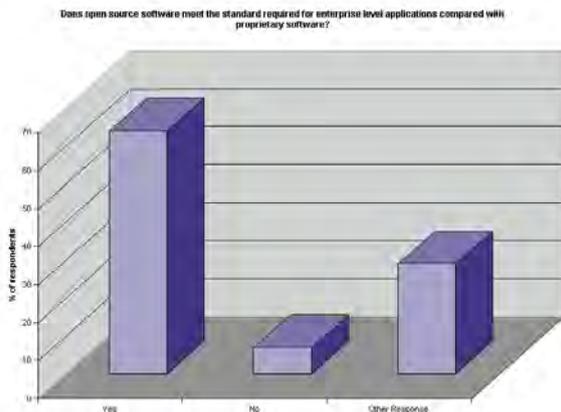
Real-Time Linux event

Dates have been announced for the third annual Real-Time Linux Workshop. The event will now be taking place on the 26th November this year, in Milan, Italy.

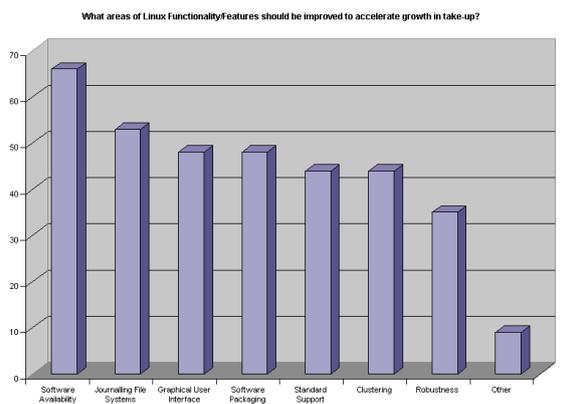
As a follow up to the previous events held in Vienna and Orlando, the focus this time will be to present updates, but also assess the “real” demands of users of real-time systems and applications. The workshop will consist of a series of tutorials on Linux in general, debugging real time applications and case studies of real time applications currently running throughout industry.

Watch this space for more details on this events as the date draws nearer.

Comment



The top 1000 ISPs put a lot of faith in the progress of Linux...



...but there are still plenty of areas which need to be improved.

Linux powers the Internet

Research confirms massive growth on the web

The use of Linux as a web server is set to increase by over 150%, according to research conducted by Idaya (www.idaya.co.uk), the software developers behind freeVSD. Further questions in their survey revealed that most respondents believed that Linux would be the dominant web platform by 2002, and that open source technologies in general would become increasingly important to this market.

The research was conducted by questioning business managers and technical heads at the top 1000 ISPs worldwide during March 2001.

The research further reveals that 64% of ISPs think open source software meets the standard required for enterprise level applications, comparable with proprietary software.

But these figures are based on anticipated improvements in Linux, and many of the people interviewed felt that there should be greater convergence among FreeBSD and Linux variants to capitalise on their strengths and remove some barriers.

Austin Delaney, founder of the freeVSD project added, "The pace of Linux growth continues unabated, fuelled by the obvious economies open source software delivers into the ISP community. The ability of the ISP to bring down the cost of services, yet rely on robust GPL applications, is putting web-enablement within the grasp of a far wider global community.

"I think everyone has realised that open source in general, and Linux in particular, is here to stay."

Standards and choice

In the free software world, one of the many benefits is the freedom of choice, and the opportunity to innovate.

That is why we have 1000 different window managers, CD players, instant messengers etc. This gives us great choice in choosing the tool we need for our purpose and not tying us down to solution. Although choice is fantastic, it comes at a price – integration. Ahhh... integration; the buzzword of the business foundry, the goal that any manufacturer strives to attain. Everyone wants their email client, web browser, word-pro and pet dog to work together in a seamless fashion.

This has caused some problems in the Linux world where we have all the CD players and window managers choosing standards that suit as many people as possible; after all we don't want to lock anyone out of the Linux community. To remedy the situation, there have been efforts such as the Linux Standards Board, and efforts to bring GNOME and KDE interoperability together. I believe these standards are the key to desktop success. Although I am a KDE developer, I would like to fire up KOffice and have it integrate with the GNOME office suite, and I would like to share attributes across applications.

I don't think the focus on integration is bringing KDE and GNOME towards the same goal... they are two very differing projects. I believe the focus should be in building a layer of interoperability and standards that both can interact with to meet this integration. While this layer will enable us to integrate between desktop standards, each desktop will still have the opportunity to learn and improve from others.

One other thing we need to battle for in the quest for integration, is quashing the image that GNOME and KDE developers are enemies. I know for a fact that mostly this is not the case. This negative image affects both camps and above all, we need to remember that we are all open source people...



Jono Bacon

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

PYTHON DEVELOPMENT

BlackAdder V1.0

■ **PUBLISHER** The Kompany ■ **WEB** www.thekompany.com ■ **PRICE** £399 (Business Edition) £79.99 (Personal Edition)

Richard Drummond finds out whether this latest Python IDE is as clever as a man with three heads and a degree from Hull university.

Python by itself is a great tool for cross-platform rapid application development. It is object-oriented, has an economical syntax and is easily expandable with bindings for your favourite widget toolkit. But combine the power and flexibility of Python with a goofy visual IDE and your productivity will skyrocket.

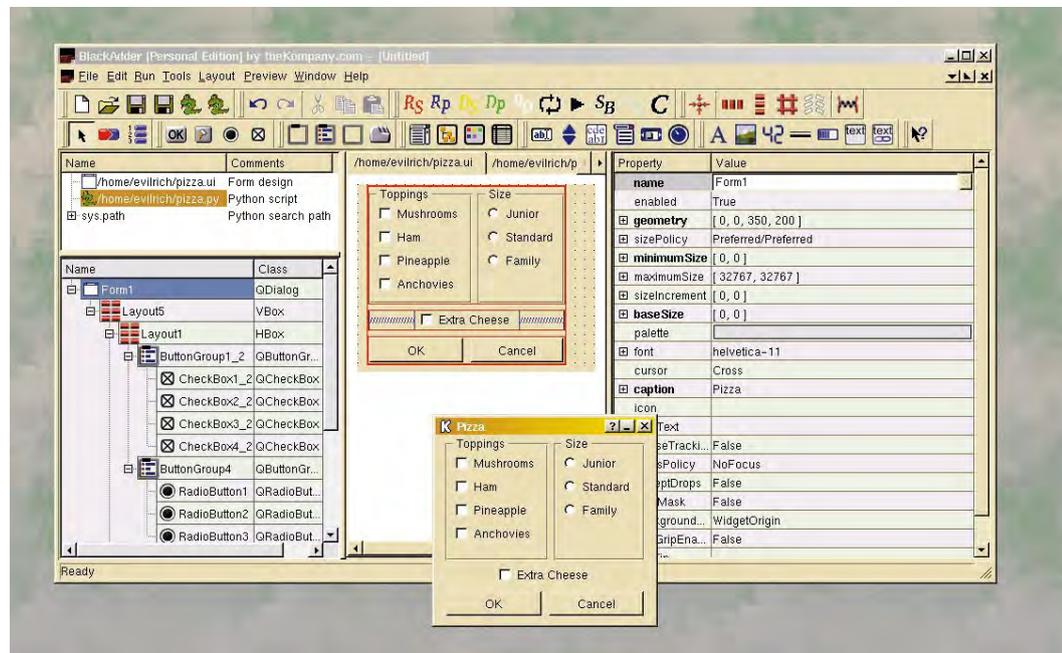
BlackAdder is just such a Python IDE and is based on the author's own *PyQt* Python bindings for Qt and Trolltech's *Qt Designer*. It features a simple project manager, a syntax-highlighting editor, a debugger and, of course, the Qt form editor itself. The full version will also include the *mxODBC* package, a module that provides access to any ODBC database engine and is nearly 100% compatible with version 2.0 of the Python Database API.

Both the Linux and Windows versions of the application are provided on the *BlackAdder* CD – the Linux version being packaged for Red Hat, Debian, Mandrake and Slackware. There are no major software requirements for installing *BlackAdder*, since Qt 2.2.4 and *Qt Designer* are statically linked with the main executable. Under Linux you must have Python 2.0 installed (it's included with the Windows version), but this should be no problem since it will be available with most distros anyway.

BlackAdder is supplied with a comprehensive array of documentation – covering both Qt and Python – and both the standard Qt examples and the Qt tutorial have been converted to be relevant to Python. Thanks to all this, it should be a fairly easy system for the beginner to get to grips with.

Follow the script

All the major components of the *BlackAdder* IDE live in single window. The project manager appears in a



The integrated *Qt Designer* lets you build frames and dialogs largely by just pointing-and-clicking.

panel on the left, and the files that you are editing – whether forms or scripts – appear in a tabbed panel on the right. Various other panels embed themselves in the window from time to time such as the property editor and object hierarchy view. Personally, I feel this way of working makes me claustrophobic – I would like to be able to tear off panels and put them where I want – but you may like it.

The *BlackAdder* project manager is fairly basic and simply provides you with a means of grouping together and managing a set of Python scripts and Qt forms. The components of your project are presented in flat list followed by a tree of directories representing Python's search path for modules. Doubling-clicking a script will open it in the script editor, while double-clicking a ui file will open the form in the form editor. Annoyingly, *BlackAdder* won't allow you to open any other sorts of files, even plain text files. You also cannot open files

without write permission – so, for instance, you can't view a standard Python module when logged in as a non-root user.

The script editor is based on the Scintilla widget and provides syntax-highlighting of Python scripts. At the moment, however, in *BlackAdder* it provides very few other features. There's no search and replace function, no undo facility and no clipboard support. These are on the to-do list however, and will hopefully make it into the release version. I hope they add a line number counter on

that list, too.

The preferences editor allows you to configure the tab size, indentation and syntax highlighting for the script editor. You can set typeface, font size and colour for each of Python's lexical elements – but one annoying aspect is that you can't change the overall size of text without going through and changing the font size for each of Python's grammatical elements.

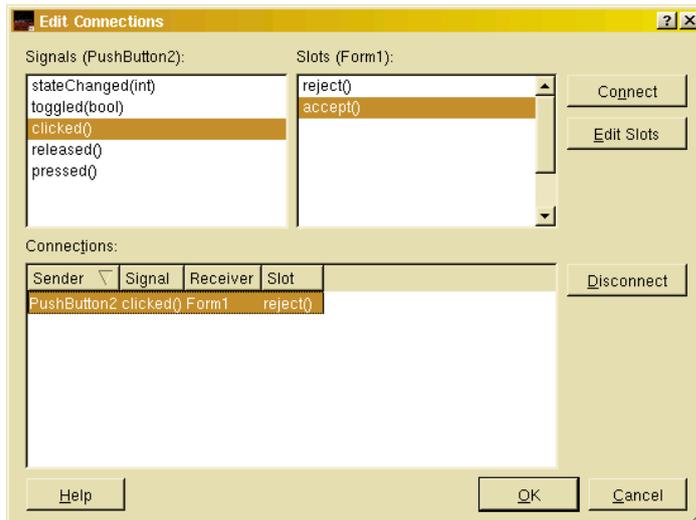
Going all GUI

As mentioned above, form building in *BlackAdder* is done with an embedded

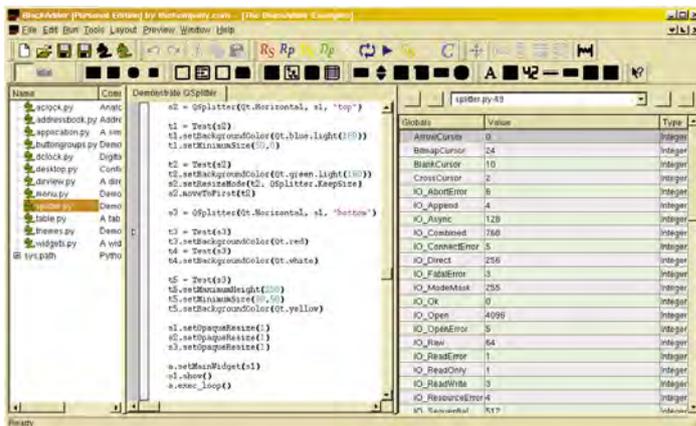
Licensing BlackAdder

The Personal Edition of *BlackAdder* is intended for home – that is, non-commercial – use, and you may distribute any of the *BlackAdder* runtime components with applications you create. The Business edition includes the right to distribute all the necessary runtime components (except *mxODBC*).

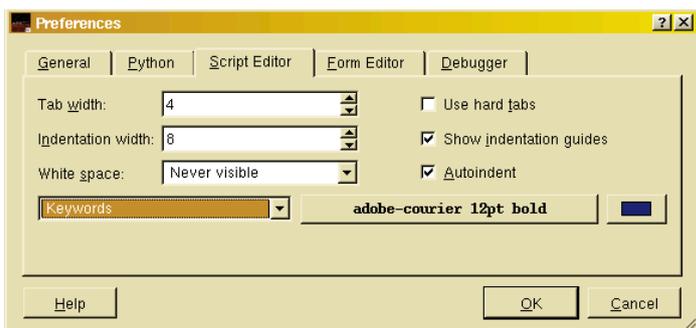
How is this possible? Aren't Qt, Python and PyQt open source? Well, a GPL'd version of Qt is only available for UNIX platforms – not for Windows. The version of Qt linked with *BlackAdder* is a non-GPL'ed version that the Kompany have licensed from TrollTech – as is the built-in version of Qt Designer.



The Connections editor lets you hook up the signals and slots in your GUI and breathe life into your application.



BlackAdder's debugger lets you single-step through your application, but its functionality is rather limited at the moment.



The syntax-highlighting is very configurable. You can pick the typeface, size, and colour used for displaying Python's lexical elements.

version of *Qt Designer*. This is a fairly sophisticated GUI designer that lets you create Qt GUIs with a point-and-click interface. I have to confess when I first tried it, that I found it incredibly unintuitive to use, but after persevering, I've grown to like it.

The form wizard supplies templates for building windows and various sorts of dialog, so getting started is easy.

The various UI elements are then selectable from a menu or the toolbar. You click the component you want, then click the form editor to add it to your window or dialog. You can then drag it out to the required size and modify any properties in the properties editor. Tools exist to group your components horizontally, vertically or in a grid. You select a set of

components and then pick a layout manager. Once grouped, components are fiddly to manipulate on the form, but the object hierarchy view provides finer control for selecting components that have already been grouped. Qt Designer supports all the major Qt components, but one obvious and desirable omission is the *Splitter* widget – the balance control that splits a container horizontally or vertically in two and allows the user to change adjust the relative sizes of the 'halves'.

Qt provides a signal and slot mechanism for communication with your GUI, and this too can be managed from within Qt Designer. Choose the 'Connect Signals/Slots' option from the Tools menu and then click a component as the signal source and drag to where you want the signal sent. A dialog pops up where you can configure that connection. The slot editor lets you define additional slots.

When you are happy with your form, you can turn it into a Python class with the 'Compile form' option. This emits some Python source code with an *init* method to construct your GUI plus dummy methods for any additional slots you may have created. You are then free to inherit that class and use it within your own code.

Squashing bugs

BlackAdder's integrated debugger lets you trace through the execution of your Python project. (*BlackAdder* only supports debugging of scripts executed on its own context, though, not external scripts run on an embedded interpreter or CGI engine like, for example, *Wing IDE* does). This lets set breakpoints and single-step through the execution of your source code. There's also a shell window that lets you enter and execute Python statements interactively.

The current values of any local or global variable on the current stack frame can be viewed on screen. The debugger displays each scope of variables as a page in the trace display and the user can switch between the various scopes with a chooser widget. This system works, but I think a tree-list where each scope is a leaf on the tree's root, like some other debuggers use, is easier to work with.

BlackAdder's debugger works adequately, but it is the weak link in the suite. With only one basic type of breakpoint and only the one single

step method, it can be frustrating to get your program to halt at some arbitrary point. It really needs the usual step over and step into functions. What's more, it's easy to get lost while debugging, because the source code display doesn't keep up with the debugger. The current line is only highlighted if you have the relevant file open and the right place.

Going forth...

On the whole, *BlackAdder* is good – but it could be better. The integrated editor, UI designer and debugger do work well together, it's just that some of the missing functions and lack of attention to detail make the whole system more awkward than it should be. I found myself frequently thinking, I want to be able to do this, why won't it let me?

Nevertheless, you can't deny that this fulfils a very desirable function. The ability to be able to rapidly throw together a GUI and produce a working program that will run on Windows or any UNIX flavour just by writing a couple of lines of Python is fantastic. It's a perfect system for prototyping and producing one-off get-the-job-done applications. When the *mxODBC* package becomes available

BlackAdder will be great for churning out simple and portable front-ends for database applications. Another major advantage over other systems is that, since Qt Designer's user interface definitions are stored as XML, it is possible to port between Python and C++ and vice-versa with relative ease – all that is required is to use a different version of the user interface compiler (UIC) to compile your forms into the desired target language.

Despite any shortcomings, *BlackAdder* at this moment is a very usable and stable development platform. I just can't wait to see it in a few update's time. **LX**

Linux Format VERDICT

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

A visual IDE for Python and Qt that is easy to use but is slightly lacking in the features department.

LinuxFormatRating
7/10

REAL WORLD PERL

Mail delivery filters in Perl



Charlie Stross finds a way of using Perl to strip the unwanted code from your email, speeding up delivery and making it accessible for all users.

On the CD



The program

You'll find Charlie's Mimex program on this month's coverdisc in the Perl directory.

A mail filter is a piece of software that lives on a mail server and does something useful to email that's in the process of being delivered. In this month's tutorial we'll look at a simple(ish) mail filter that comes in handy if your mail reader doesn't like HTML mail – along the way, we'll look at how mail is delivered on Linux, how we do file locking on Linux using Perl, parsing (of MIME attachments and HTML), how to write a filter program in Perl, and a whole load of other topics.

Making HTML mail readable

In this month's tutorial, we're going to see a simple program for filtering email. By convention, email should contain either plain ASCII text, or MIME attachments – capsules that contain data and provide information about its format. Human-readable mail is supposed to be plain text, but some badly behaved email readers send mail in other formats – HTML is most common, but Microsoft Word documents come a close second for nuisance value. If you use an HTML-compatible email tool to read mail, this won't be a problem for you, but some of us use *Pine* or *Mutt* and read mail via telnet; HTML attachments are a bane of our existence because they're effectively unreadable.

Mimex (MIME Expander) is a simple filter designed to translate HTML attachments into plain text. It does this by intercepting your email before it is delivered (typically by *procmail*, the common mail delivery agent used with *sendmail*). It reads a mail message on its standard input, and uses the module `MIME::Parser` to work out how the message is structured. If there are no attachments, it passes the message on, turning the body into a text/plain attachment along the way. If there are attachments, it calls `parse_and_translate()` recursively on each attachment (because attachments may contain other attachments, forming a tree structure).

`parse_and_translate()` checks to see if an attachment is of type text/html. If it isn't, it simply sticks a duplicate of the attachment onto an array (called `@main::stack`). If it is, it then uses another module from CPAN – `HTML::Parser` – to create a plain text copy of the HTML file. `HTML::Parser` recursively walks the tree of HTML elements in the HTML document returned by `MIME::Parser`, and every time it encounters some text it appends it to the internal variable `$body_html`.

You probably don't want to try and look too deeply into what `HTML::Parser` does – the documentation is obscure and the concepts mind-warping. When `parse_and_translate()` has extracted the plain text from an HTML attachment, it creates a new text/plain attachment labelled "This is an automatic

translation into plain text of the following HTML attachment", and saves both the text/plain and the original text/html attachments on the stack of MIME parts.

Once we've recursively parsed the MIME attachments, `@main::stack` consists of an array of attachments – and each HTML attachment is preceded by a plain text translation. We therefore replace the attachments array on our parsed message with the contents of `@main::stack`.

What happens next is highly dependent on how you deliver mail. If you don't use *sendmail* and *procmail*, you will need to modify this part of the program! *Procmail* locks mailbox-format mailboxes using `flock()`; but Perl may actually use either `lockf()` or `lock()` depending on what its configure script thinks is appropriate when you install it. You can see if Perl understands `lockf()` by running the following short script and scanning its output for `d_flock`:

```
#!/usr/bin/perl
use Config qw(myconfig config_sh config_vars);
print config_sh();
```

If we don't have `lockf()`, mimex will need modifying to call the external program lockfile (which comes with *procmail* and is designed specifically for locking mailboxes). Otherwise, we invoke the `lock()` and `unlock()` subroutines supplied by the module `File::Flock`; these let us do the file locking inside Perl, and we wrap a call to `lock()` and one to `unlock()` around code that uses `IO::File` to open our mailbox, write the message to it (as a string), and close the mailbox.

Note the error-trapping code: this attempts to save the message in a file called `~/failed.incoming.mail` if for some reason it can't write to your mailbox. This isn't perfect – it doesn't trap fatal errors caused, for example, by an incoming signal such as `SIGHUP` or `SIGQUIT` – but it reduces the risk of losing mail.

The most noteworthy point about this program is that, by extending `parse_and_translate()`, it is possible to support other file format conversions. For example, it is possible to invoke the external program `word2x` to translate Microsoft Word 6.0/95 documents into plain text – or even to replace `parse_and_translate()` with a rule-driven translator that does a variety of tasks (including virus-scanning and spam detection).

Mail boxes and delivery

Most mainstream distributions of Linux run *sendmail*. This is an SMTP server; that is, it uses the Simple Mail Transport Protocol to receive and transmit mail to other servers. When a message arrives that *sendmail* figures needs to be delivered locally, it calls

a local delivery program – and the most common one found on Linux is *procmail*.

Procmail reads a global configuration file (`/etc/procmailrc`, if present), then processes a special script in `$HOME/procmailrc` (if present). Any mail that's left over after processing the *procmailrc* scripts is then dumped in the default system mailbox.

A mailbox is either a file or a directory containing mail and laid out in a special format. The most commonly-used format, which *procmail* understands, is *mbox* format: mail is saved in a single file (usually under `/var/spool/mail/$USERNAME`), each message separated from the next by a blank line and then a line of the form: 'From somebody_or_other date_stamp'

There are other mailbox formats, notably *MH* (which stores messages as separate numbered files in a directory), but *procmail* is best at dealing with traditional *mbox* format mail files. When writing to a mailbox, the application tries to lock the file (to avoid getting into a race condition with other *procmail* processes or mail clients): on Linux it usually uses the **flock()** system call. It's also smart enough (in current versions) to cope with NFS-mounted mailboxes.

Note: if you are not using *procmail* for local mail delivery, and are not using *mbox* format mail folders, some of the assumptions in this tutorial will be wrong and if you try to use the program we're discussing you may lose mail. You have been warned!

The purpose of *procmail* is to give you some control over where mail goes. The `.procmailrc` file contains 'recipes' in the program's command language. These consist of a directive to apply a file lock, a pattern to scan the message for, and some action to take if the pattern is found. The syntax of the `.procmailrc` language is described in the man page 'procmailrc'; a bundle of examples are given in the man page 'procmailrc', and the command itself is described in 'procmail'.

A typical *procmailrc* recipe looks like this:

```
:0:
* ^TOperl5-porters
mail/p5p
```

The first line tersely tells *procmail* to use a file locking mode with a default lockfile name.

The second line says that all the header lines are to be subjected to *egrep*-style regular expression matching, starting at the beginning of the line '^', then looking for the special *procmail* macro 'TO' followed by 'perl5-porters' (TO is internally expanded to `^(Original-)?(Resent-)?(To|Cc|Bcc)(X-Envelope|Apparently(-Resent)?-To):(.{^a-zA-Z})?` by *procmail*.)

The third line indicates that if the pattern match returns true, the message is to be delivered into the mailbox `mail/p5p` in the user's home directory.

For our purposes, we want a simple `.procmailrc` file that causes all mail to be piped through our filter. Our filter takes a single argument, the name of the mailbox to deliver to: for example, `/var/spool/mail/charlie`. We can ensure that all mail is piped through our filter (saved as, say, `/usr/local/bin/mimetranslate`) like this:

```
:0:
|usr/local/bin/mimetranslate /var/spool/mail/charlie
```

Note that the *mimetranslate* filter must be world-executable, should have a valid **#!** line pointing to the installed Perl interpreter, and your home directory needs to be sufficiently readable for *procmail* (spawned by *sendmail*) to read its `.procmailrc` file. Otherwise something will fail silently, and you will end up losing mail.

It's best to experiment by creating a dummy user account and installing *mimetranslate* there. Test it out fully before you install it

Installing and using modules from CPAN

This filter does lots of stuff. In fact, it relies on external modules available from CPAN, the Combined Perl Archive Network: if it did all the necessary parsing internally it would be huge.

Before you can use the filter you'll need to install some modules – notably `MIME::Tools` (a whole bundle of modules that include `MIME::Parser`), `HTML::Tools` (for `HTML::Parser`), `File::Flock`, and `IO::Stringy` (for `IO::ScalarArray`).

The easiest way to install modules is with the CPAN module, a shell packed full of utilities for downloading, testing, and installing other modules. You invoke it like this:

```
perl -MCPAN -e shell
```

The first time you run this (as root) you will be prompted for a load of configuration information (with supplied default values): CPAN needs to use *Lynx* and *FTP* tools to download modules, wants to know which *FTP* sites to go to, and so on. After you have configured `CPAN.pm`, it will give you a command prompt.

You can search for a module by looking for regular expressions in its description:

```
i /mail/
```

performs a case-insensitive search for modules and bundles with the string 'mail' in their name or description. You can install a module once you've found its name like this:

```
install MIME::Tools
```

This causes `CPAN.pm` to fetch the most recent copy of the module sources from the primary CPAN server you configured. Once it has fetched the module it will check for dependencies and fetch any other modules that `MIME::Tools` requires in order to run. It will then unpack, build, test and, if successful, install each module in turn.

Alternatively, if you are installing a properly packaged module that isn't available from CPAN, you can download the *tar.gz* archive and install it by hand. The standard way to do this involves issuing the following commands:

```
tar xvzf My-Module-1.0.tar.gz
cd My-Module-1.0/
perl Makefile.PL
make
make test
make install
```

A properly packaged module contains a `Makefile.PL` script – generated by the program *h2xs*, which is distributed with Perl – which is used to generate a `Makefile`. When executed, the `Makefile` configures the module for local execution (compiling any *C* extensions that are needed), and can be used to run the test scripts (if any) and install the module in Perl's local `site_perl` directory tree.

on your main user account, unless you are willing to live with the risk of losing mail.

Standard input, output and files

On Linux and other UNIX systems, processes read and write data to disk by way of file handles. A file handle is a data structure that keeps track of an open file (it gives your process a handle on the file it's dealing with). Because the UNIX namespace model views most things as files – even hardware devices and terminals – newly created processes come with some standard file handles ready and waiting to use. These are standard input, standard output, and standard error (expressed as `STDIN`, `STDOUT`, and `STDERR` respectively).

Standard input is where a process usually expects to read data from: it can be attached to an actual file if we start the process using a shell redirection operator (e.g. 'myprog < data_file' opens `data_file` and passes it to *myprog* on `STDIN`), but otherwise if we start a process from an interactive terminal session the standard input is taken from the terminal's keyboard (so we can type commands or data straight into the program).

We can create a file handle using `open()` and close it using `close()`. The former takes a filename as a parameter and returns a filehandle, while requires a filehandle as a parameter.

```
open(PASS, "/etc/password"); # open the /etc/password file,
return
```

```
# filehandle PASS
```

```
close PASS;
```

```
# close it again
```

`open()` has some complex additional parameters: it checks the filename and varies its behaviour depending on whether it sees some additional characters in the name. These determine the mode in which the file is opened – for reading, writing, or appending to. For example:



Tutorial Developer Perl

```

<< open(PASS, "</etc/password"); # opens in explicit read mode
open(PASS, ">/etc/password"); # Naughty! explicit write mode,
clobbers
# the file if it already exists, creates
# it afresh if it doesn't
open(PASS, ">>/etc/password"); # append mode
open(PROCESS_OUTPUT, "cat /var/log/httpd/access.log |");
# fork() a child process (cat) and
# read its output on PROCESS_OUTPUT

```

In Perl, a file handle is a first-class object: but confusingly, it isn't identified by a symbol in front of its name (like `$my_scalar` or `%your_hash`). By convention, filehandle names are in UPPER CASE, but it's legal to ignore this. You can assign a filehandle to a scalar, so it's legal to say `$my_stdin = STDIN`; but you can't pass a bare filehandle as a parameter to a subroutine. (To get around this, either use the `IO::File` module, which wraps a filehandle up in an object and lets you pass it around freely) or read up on globbing in *Programming Perl* – but be aware that globbing is among the features due to be dropped from Perl 6.

To actually use a filehandle, there are a number of ways to get data in and out. The simplest way for using buffered *i/o* (usually on text files that are open purely for reading through or writing to) is to call `readline()`, which does exactly what you would expect – it returns the next line from the file (defined as a series of bytes until the pattern stored in the special variable `$/` is encountered – by default `$/` is a newline). In list context, `readline()` will return an array of lines (up until the end of file). It implicitly advances a pointer through the file referenced by the filehandle, so successive calls will return successive lines.

For example:

```

while (! eof(HANDLE) ) {
    push(@contents, readline(HANDLE));
}

```

is equivalent to:

```
@contents = readline(HANDLE);
```

both of these snippets read all the lines referred to by `HANDLE` into the array `@contents`.

You can send output to a writable file handle using `print()`:

```
print MY_HANDLE "this is going to MY_HANDLE\n";
```

And you can specify that input or output is unbuffered by setting the special variable `$|` to 0 (its normal non-zero value means that *i/o* is buffered).

But that's not all! `read()` provides a way of getting at a file in regular-sized chunks rather than textual records – for example, if you're trying to read fixed-size data records. (`write()` does not do what you'd expect it to do by looking at `read()`; to send fixed-size records to a file just use `print()`). And there are two entire alternative systems for reading and writing files – `opendir()`, `readdir()`, and `closedir()` for reading directory information, and `sysopen()`, `sysread()`, `syswrite()`, and `sysclose()` for doing low-level file access (using `libc`'s built-in file access routines, rather than going through Perl's buffered access system).

In the `mimex` (MIME Expander) program presented in this tutorial, we get at files using `IO::File`. `IO::File` is a subclass of `IO::Handle`, which gives file handles object-like semantics. For example:

```

my $handle = IO::File->new();
# create a new IO::File object

```

```

$handle->open("/tmp/tempfile", "rw"); # open /tmp/tempfile in
read/write mode

```

```

if (! $handle->open("/tmp/tempfile", "rw")) {
    # we failed to open the file, so we can take some
    # appropriate error-related action here
}

```

```

while (! $handle->eof()) {
    # while eof(), which tests for the end-of-file condition, isn't
    true,
    # we can read data from $handle
}

```

```

my $line = $handle->getline(); # read a line from
$handle

```

```

$handle->autoflush(0); # set non-buffered
access –
# equivalent to $| = 0

```

```
$handle->close(); # close the file
```

`IO::File` is useful because in addition to enhancing readability, it gives us the ability to pass filehandles to subroutines easily in a manner that will be supported in Perl 6: it also lets us keep track of variable scope effectively using Perl's 'strict' scope checking.

If you want to learn more about file *i/o*, you can glean a lot of information from *Programming Perl*, Chapter 16 (Interprocess Communications). And there's a whole slew of useful tips and tricks in Chapter 8 (File Contents) of *The Perl Cookbook* (Tom Christiansen and Nathan Torkington, pub. O'Reilly & Associates, ISBN 1-56592-243-3). **LXF**

Testing and debugging

90% of the work in any programming project is design; and as the military know, no plan ever survives contact with the enemy. If you get the design slightly wrong, how do you put it right? The first line of debugging support in Perl is the compiler's `-c` flag. Run:

```
perl -c myprogram.pl
```

And Perl will syntax-check `myprogram.pl` without executing it; if there are any obvious errors it will complain. Add a `-w` flag for a pedantic nit-picking examination of your code (for example, to generate warnings about uninitialised variables) and set the "use strict" pragma early in your code. If it passes the `perl -cw` test, there's nothing obviously wrong with your program that will stop it running.

The next step is to use appropriate test data. In the `mimex` example, which reads a file from standard input, the test data is an email message: several different ones are needed. We need to test that `mimex` passes through non-MIME messages more or less unchanged (in fact, it turns the body of the message into a single MIME attachment of type `text/plain`, but doesn't otherwise mangle it). We need messages with a single non-HTML attachment, two or more non-HTML attachments, and one or more HTML attachments. We need messages with nothing but HTML, and with mixed attachments.

To figure out if things are working as they should be, the programmer's first friend is the humble `print()` command. Add lots of print statements to your code if you think something is going wrong. Adding `print("myvar contains: $myvar\n")` before and after some code that modifies `$myvar` will give you insight into whether the code is working properly or not.

`Data::Dumper` is a module that dumps out a neatly-formatted rendition of a data structure or object in Perl in a form that can be re-loaded by `eval()`'ing it. Use `Data::Dumper` is your friend: you can then insert lines like `print 'my object is now: ', Dumper $my_object;` to look inside complex structures.

If things are really puzzling you, you may need to resort to a proper source-level debugger: type `perl -d myprogram.pl` to start your program under the Perl debugger. Perl's built-in debugger is way too complex to describe in full in this tutorial – though we'll look at it in a future issue. You can find a detailed tutorial on it *Programming Perl* Chapter 20 (The Perl Debugger). There are also a number of alternative Perl debuggers available – the debugger is actually written in Perl – including Perl support for the excellent `DDD`, Data Display Debugger (see <http://www.cs.tu-bs.de/softtech/ddd/>).

MIMEX – Perl program for stripping HTML code from email

```

#
# MIMEFilter — take HTML mail and insert a plain text
section
#
#
# read in mail from STDIN. Parse using MIME::Parser
# For each attachment, if it is of type text/html, parse
using
# HTML::Parser to extract the plain text. Create a new
attachment
# containing the text, and insert it before the HTML
attachment.
#
# After processing attachments, save the parsed
Mail::Internet object
# in the mbox passed as a parameter ($ARGV[0]),
remembering to
# use flock(2) so we interoperate (nicely!) with procmail.

use strict;
use HTML::Parser 3.00 ();
use MIME::Parser;
use IO::ScalarArray;
use File::Flock;

# Create and set the temporary mime attachment
output directory:
my $mimedump = "/tmp/mimedump-tmp";
(-d $mimedump) or
  mkdir $mimedump,0755 or
  die "mkdir: $!";
(-w $mimedump) or
  die "can't write to directory";

my $parser = new MIME::Parser; # create the parser
object, and ...
$parser->output_dir($mimedump); # tell it where to
store temporary files

# identify name of destination mailbox
die "No mailbox specified!\n" if (! $ARGV[0]);

my $mailbox = $ARGV[0];
my $failover_mailbox = $ENV{HOME} . "/" .
"failed.incoming.mail";

# Read STDIN into an IO::ScalarArray — this is a
filehandle tied to
# an array, which we use because MIME::Parser insists
on reading from
# a filehandle. If we don't want to do this, we could tell
# MIME::Parser to read direct from STDIN — but if we
do that, we
# couldn't insert additional debugging code.

my @file = (<STDIN>);
tie *MAIL, 'IO::ScalarArray', \@file;

# Parse the mail as a bundle of MIME parts (via the tied
filehandle) and
# ensure we're dealing with it as a multipart message:

my $entity = $parser->read("\*MAIL) or die "couldn't
parse MIME stream";
$entity->make_multipart();

@main::stack = (); # prepare a stack for the MIME parts
we're going to save
parse_and_translate($entity);

# replace MIME parts in $entity (our mail message) with
those on @stack
if (scalar(@main::stack) > 0) {
    $entity->parts(\@main::stack);
}

# now flock $mailbox and deliver the message

my $error = 0;
my $ret = 0;
my $timestamp = "From " . $ENV{USER} . " " .
scalar(localtime(time)) . "\n";
if ($ret = lock($mailbox)) {
    # mailbox is locked, so we can try and open it
    my $out = IO::File->new( ">>$mailbox") or do {
        warn "$!\n";
        $error = 1;
    };
    # if we got here, we opened the mailbox successfully,
so we can
    # now print the message to it, close, and unlock
    $out->print($timestamp);
    $out->print($entity->as_string());
    $out->close();
    unlock($mailbox);
} else {
    warn "lock($mailbox) failed: [$ret]\n";
    $error = 1;
}

if ($error == 1) {
    # try to save the message somewhere sensible
    my $out = IO::File->new( ">>$failover_mailbox") or
do {
        # our error-recovery strategy failed. Woe! Misery!
        die "failover: $!\n";
    };
    $out->print($timestamp);
    $out->print($entity->as_string());
    $out->close();
}
exit;

##### end of main program —
support routines follow #####

sub parse_and_translate {
    my $part = shift;
    my @parts = $part->parts();
    if (@parts) {
        foreach my $i (0 .. $#parts) {
            parse_and_translate($parts[$i]);
        }
    } else {
        if ($part->mime_type() eq "text/html") {
            # take HTML attachment and parse into plaintext
            my $body_html;
            my %inside;
            my $stag = sub {
                my($stag, $num) = @_;
                $inside{$stag} += $num;
                $body_html .= " "; # not for all tags
            };
            my $stext = sub {
                return if $inside{script} || $inside{style};
                $body_html .= $_[0];
            };
            HTML::Parser->new(
                api_version => 3,
                handlers => [start => [$stag, "tagname",
'+1'],
                            end => [$stag, "tagname", '-1'],
                            text => [$stext, "dtext"],
                            ],
                marked_sections => 1,
            )->parse($part->stringify_body());
            my $subject = "This is an automatic translation
into plain \n".
                "text of the following HTML
attachment:\n";
            my $new = MIME::Entity->build(Type =>
"text/plain",
                Subject => "Subject:
$subject",
                Data =>
"$subject\n\n$body_html");
            push(@main::stack, $new);
        }
        push(@main::stack, $part->dup());
    }
}

```

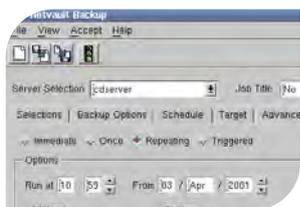
>> Next month

Next month we'll see how this program can be adapted for a variety of uses including a general purpose mail delivery agent that can do things like translate MS Word documents and PDFs into ASCII text before delivery, or check for spam.

LINUXformat Professional

News, reviews and expert advice for Linux pros

WHAT'S IN PROFESSIONAL THIS MONTH...



Review: Nick Veitch assesses Bakbone's enterprise-level backup solution. **80**



Make the most of 2.4's enhanced security features with our iptables masterclass. **82**

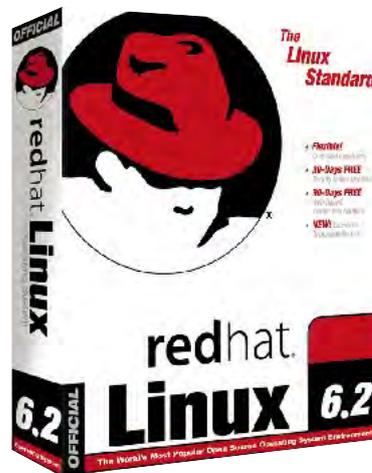
Distros beat the markets

Red Hat could soon be in the black

Despite the NASDAQ nosedive and tech stock terrors, not all is doom and gloom in the stock market. Red Hat have announced a whopping 106% rise in revenue year-on-year in their fourth quarter. They're breaking even 'per-share' and foresee profitability in the next three months.

They still lost \$600,000. Their figures were helped by cutting R&D spending and milking the profits of a recent acquisition, consulting firm Planning Technologies. But with a market capitalisation of around a billion dollars, they're looking healthy.

"We're moving from a boxed product sales into an enterprise environment", says Fiona Phelps of Red Hat Europe. A large part of their good fortune comes from deals struck in Europe with the likes of Deutsche Telekom, Lufthansa and Dresner Bank. "That's quite an achievement", Phelps noted, "considering our European



Red Hat have been looking at new revenue streams.

offices have only been open for the last 18 months".

But they may be hoping more money comes from smaller fry. Their automatic updating service, the *Red Hat Network*, designed to automatically download any necessary

bug fixes and patches every two hours, has recently ended its extended trial period. Users will still get one free license for the service, with additional subscriptions costing \$19.95 a month.

For those who don't mind thinking about what's getting installed on their system, the FTP site is still free.

Caldera, who've recently swallowed SCO Unix, also reported a sharp year-on-year rise in revenue, but have still managed to sink \$9.8 million into the red. And Corel, everyone's favourite clipart-cum-Linux vendor, has just managed to scrape into the black. But then again, they're trying to ditch their brand of Debian to focus on desktop applications for Windows and Linux.

In a year that's seen the death of many IT companies and the shrivelling of others, it's encouraging to see that those selling and supporting free software are staying afloat.

www.redhat.com
www.linux.corel.com

New Apache based server

Web server wizardry

AbriaSoft have released *Merlin Server*, a Web application server stuffed full of features. The Apache 1.3.14-based bundle includes MySQL, PostgreSQL, SSL, Perl, PHP, WebDAV, training CDs, documentation and much more, for an introductory price of \$169 – \$60 off the normal price.

Sure, you can download and configure most of these packages individually for nothing. But AbriaSoft reckons that it'd take an experienced developer 16–20 hours to get all this

up and running without their package. You may dispute that – and it'd make an interesting competition – but the 30-day telephone support and tutorial extras will make this an attractive product for business.

The real drag? Currently only Red Hat 6 and 7 and YellowDog 1.2 are supported, with *BSD, Solaris on Sparc, TurboLinux and Red Hat on Alpha planned. SuSE, Mandrake and other distributions are out of luck.



Merlin Server looks promising, but needs to support more distros.

A lite version, without the Red Hat 7 support and extras such as documentation, is available for download from their website.
www.abriaSoft.com

Comment

MySAP on IBM z900

Getting SAPpy at CeBIT

If the words Customer

Relationship Management make you go all squishy, you've probably attended one to many business seminars. But SAP, the company that practically invented the term – and boasts 30,000 installations from IBM to Microsoft and more – have long been supporters of the Open Source way of doing things.

They chose the CeBIT show in Hanover, Germany to announce their latest plans to host their e-business solution, mySAP.com, on IBM z900 mainframes running the Linux operating system. "Since early 1999, SAP has successfully supported Linux as a mature, reliable operating system

for e-business solutions", glowed SAP's Executive Board Member Karl-Heinz Hess.

While mySAP.com is actively courting the SME market with its powerful mix of e-business software, customer management and personalisation available through a Web front end, the machines needed to run their systems are huge. Currently only SuSE and TurboLinux have ported to the z900 series, formerly known as the S/390, so the CeBIT announcement should bode well for them. The first systems are due to ship in the second quarter of this year.

www.mySAP.com

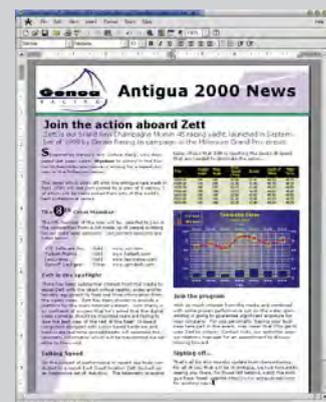
VistaSource sale

Linux office suite sold off to the capitalists.

ApplixWare has long been the leading commercial office suite for Linux. It's been around since the days before *StarOffice* was bought by Sun and rechristened *OpenOffice*, or *KOffice* was even remotely stable. But despite their longstanding Linux support, Applix have gradually moved away from the Linux desktop market.

First they split off the Linux side of their company to form VistaSource, while they concentrated on lucrative, high end corporate solutions. Now they've sold VistaSource as a going concern to a Californian private investment firm, Parallax Capital.

The sale comes as VistaSource are readying themselves to launch *AnywareOffice*, an ambitious new Java-based suite accessible through your Web browser that includes word processor, spreadsheet and presentation graphics. ASP office suites have



AnywhereOffice is accessible through Java-equipped browsers.

not proved wildly popular with users so far, and while VistaSource claim that theirs will be the first, other vendors, including Sun with the *StarOffice* suite, are having second thoughts about the viability of the idea.

www.vistasource.com

Caught in the .NET?

Microsoft's CEO Ed Ballmer recently let slip that MS .NET services could run on Linux.

Code-named 'HailStorm' and backed by commercial giants such as American Express, eBay, ClickCommerce and others, it will keep user information and preferences in a handy little XML wallet, ready for whipping out at the nearest online shop. Should the Linux community care?

Plenty of other people do. Sun have come up with their own challenge to .NET based on their Java platform. It's known as Sun ONE, and stands for Open Net Environment. The other big networking player, Novell, are planning to chip away at .NET with the imminent release of its XML-based Novell 6.0.

So it's shaping up to become a thrilling three-way fight. Each company will trumpet the 'openness' of their solution, and will attack the security and ease of use of the others. The temptation is to grab a beer, sit back and enjoy.

But it's not that simple. Swapping information between applications in XML files is not a bad idea. You may not like the idea of storing all your personal habits in a Microsoft-inspired file so a site selling airline tickets can decide the class you want to fly, but all joining hands and chanting spells against this insipid 'personalisation' is not going to stop it.

As always, there will be competing standards. Unfortunately, consumers care about usability, not standards. If HailStorm works, your site will at least have to offer it as an option. Either you'll run Microsoft's port of their .NET programs or wait, losing business, until someone engineers an Open Source alternative.

Perhaps the most frightening thing about .NET is its name. It's not an operating system or an application, it's a strategy. With a Microsoft product sitting between the operating system and the browser-based application, Linux will lose the two things it's most respected for: stability and security. We can hope it's a flawed strategy, but we can't ignore it. We need our own.

Got a hot news tip? Send it off to kynaston@yahoo.com.



Matt Kynaston

Matt is a freelance writer and web developer.

Reviews Professional Netvault

NETWORK BACKUP SOLUTION

Netvault

■ **WEB** www.bakbone.com ■ **PRICE** £1,000

Nick Veitch investigates Bakbone's claims that backing up isn't hard to do.

Do we really have to sell you the benefits of keeping regular backups of all your data? For the home/small business user, such precautions can prevent unpleasant and inconvenient surprises – for larger corporations, data security can mean the difference between being in business or not being in business.

Netvault is squarely aimed at the latter marketplace, with a modular and scaleable solution for backing up data of all kinds on multiple platforms. The basic idea is that you populate your network with the client software, and then install a *Netvault* server on a machine. Backup devices can be located anywhere on the network, just as long as they are connected to a machine running the client software. Now you can create your normal backup jobs, sit back and let the server get on with it.

For larger networks any number of servers can be installed, and all of them will still be able to access the same devices. *Netvault* also understands SAN networks and dynamically shared devices, so there's probably no need to alter your established network configuration to use it.

Application Plugins

You can, as we have pointed out, select individual files and directories, and even raw devices for backup. But *Netvault* also offers application specific plugins. For example, the Oracle plugin will automatically find your Oracle database files and back them up – how handy!

Various types of data are supported using this method, which is a lot simpler and more foolproof than making sure all the vital files are selected from a filesystem listing.

The only downside with the plugins is the number of applications supported this way: Oracle, DB/2, Sybase, Informix, Lotus Notes, NCR, ODS, MS SQL, MS Exchange. Can anyone see the odd one's out? Like MySQL and PostgreSQL?

Devices

Netvault supplies plugins to support a wide range of typical backup devices, though setting these up in the first place may be a little bit cumbersome. Most are recognised straight away, and once they are set up, they're set up.

The backup devices don't need to be on the server (or servers) you have selected to deploy *Netvault* on. In fact, they can be attached to any client machine reachable through the network. What's the point of having a client co-ordinate local backups

through a server? There are a number of good reasons – the device can also be used by other clients on the network, you will have logs of all the backups, made on a central database and so on.

Pseudo-devices can also be created on clients. This involves sending them the plugin for creating such devices, then specifying various options. Using a pseudo-device, you can set up areas of disk space which will behave like a normal backup device. Pseudo-drives will actually be

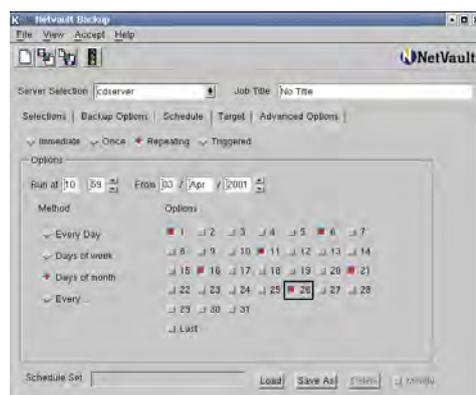
created on the target machine and will take up the allocated amount of space regardless of what's on there – that is, if you create two 100Mb drives, you'll find that two files of 100Mb have been created. Although it would be nice if this space could be allocated dynamically, at least you are sure the target will actually have the space available before the backup is started.

Jobs

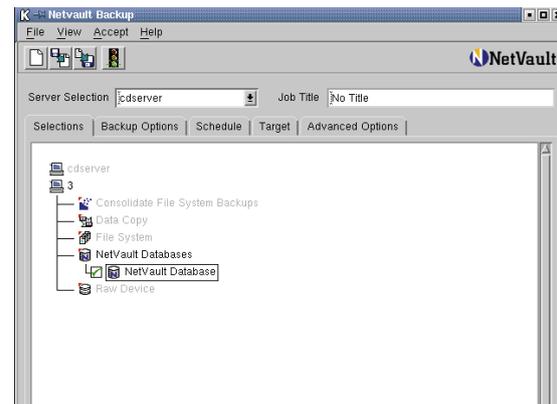
Netvault caters for various types of jobs, and using its application plugins, various types of data backup. The procedure is quite straightforward, even if the details can be complicated. To start a backup job you merely have to indicate the client and the data to be backed up and set a time. You can choose an immediate run, a timed backup, a repeating backup or a triggered job, and the usual choice of full and incremental backups is available. Finally, select the target



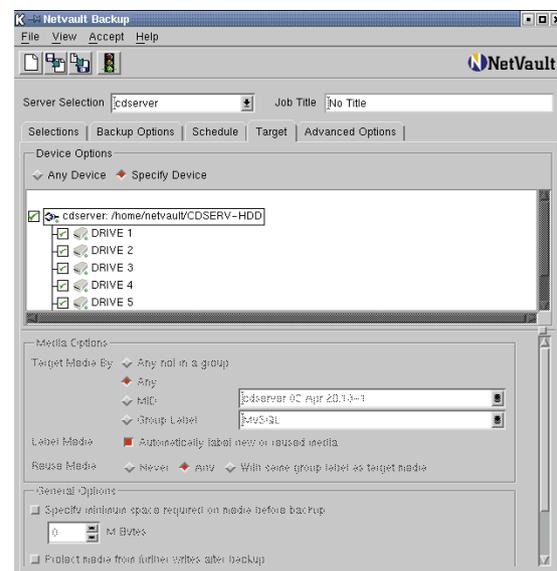
Online help provides a quick reference to all the GUI tools – just click on the relevant area.



Jobs can be given a variety of schedules, including weekly or running on particular dates.



Choose the data you wish to backup – from the filesystem, a raw device or one of the plug-ins.



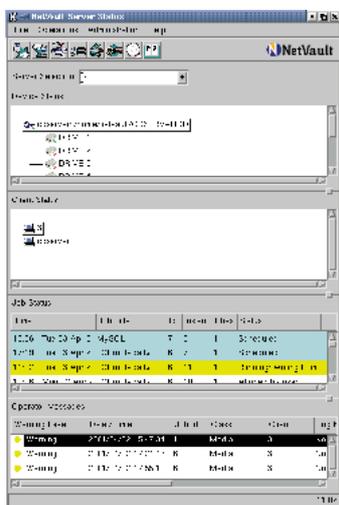
Backups can be media or device targeted.

media (or select ANY if you aren't particularly bothered which device will be used for the job) and everything is ready to go.

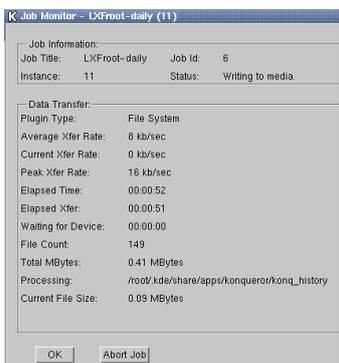
Scheduled jobs are added to the list of jobs waiting to be enacted, which you can view at any time in the job management section. Jobs which repeat (e.g. a daily backup) automatically respawn when they are called. However, unless you give the jobs specific names like "Daily mailserver backup", you can't actually tell from the jobs list which items are continuously scheduled and which ones aren't.

This is a pity, because it would be nice to be able to view the waiting jobs by type, or perhaps even by target – imagine you knew some part of the network was going to be down over a particular period – finding all the jobs queued (on all the servers mind you) that might be effected could be problematic.

Overall the feedback from the backup jobs is a little limited as well. The message "backup failed:media not found" could be a little more specific –



The status display will inform you of what exactly is going on.



A backup in process - the status display will even time each job.

Linux requirements

NetVault only claims to run on Red Hat. In common with other commercial solution providers, Bakbone would rather only stake their reputations on one flavour of Linux, and Red Hat is it.

Although we can't categorically say it will support other Linux flavours, we can tell you that during testing we managed to install both client and server software on Mandrake, SuSE and even Debian distributions, and it still seemed to work fine.

Triggered backups

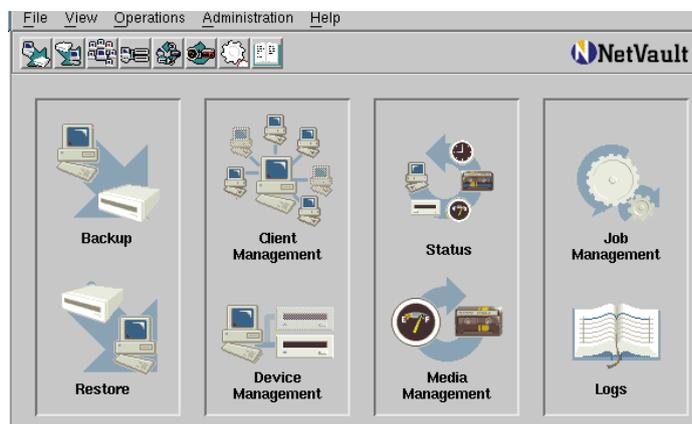
As well as some other helpful utilities for propagating device information throughout the network, formulating reports and so on, the *NetVault* utilities directory contains a program called *nvtrigger*. This is used in conjunction with triggered backups, and simply passes a string to the *NetVault* server, which it tries to match up with waiting jobs.

This is incredibly useful, as you can use the trigger to execute a backup job only after specific other conditions have been met. As well as being able to run the trigger manually, it would easily be possible to incorporate the command into scripts, causing backups to be triggered automatically after any number of conditions had been met.

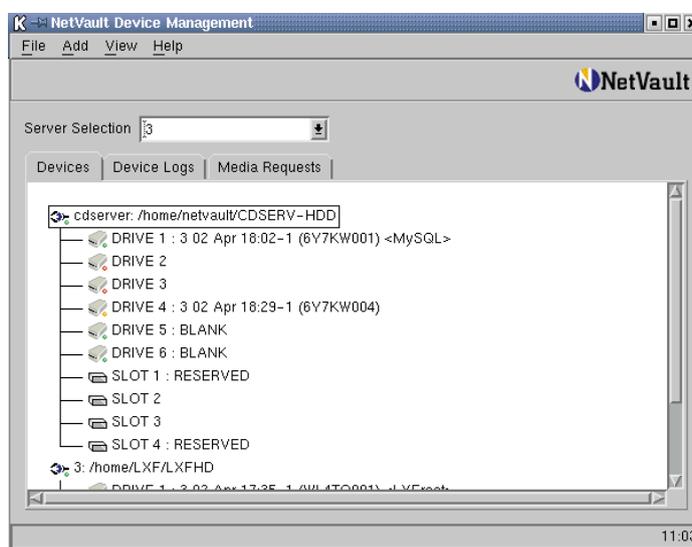
Help!

was the specified device not found on the network, was the target dataset not available or full? You'll have to root through the individual job processes to find out for sure.

Bakbone have done a pretty good job on providing help information. The online help generally consists of clickable images from the *NetVault*



The main GUI window allows access to the subsections of *NetVault*



Virtual devices can be created using disk space on clients or servers.

interface – just click on the area you need help with and you'll be taken to the relevant part of the documentation. Some areas of the documentation are a little bit sketchy though, and you may need to wade through several links to find the piece of information you are after. There is a 412 page pdf format administrators guide too, which contains quite a bit of detailed info, but again, finding the information you need to use the software can sometimes be a problem.

No problems?

While backup solutions such as BRU are fair enough for medium sized networks, *NetVault's* modular system will work much better on a more disparate network, or one which has evolving backup demands. As there can be multiple servers on the network accessing the same devices, and it supports SAN and shared drives, *NetVault* makes client/network configuration even more flexible.

The main downside to the application is how complicated it can become to set up. The GUI makes this easier, but many of the screens are not particularly intuitive. Adding devices that are automatically recognised is fairly easy, but creating a pseudo-device from spare disk space is a little convoluted. The process isn't helped by the sheer number and interconnectedness of GUI windows *NetVault* employs.

While BRU etc may look simplistic or feature-lite compared with *NetVault*, you can be pretty sure your backup tasks are going to work, and individual jobs are quite easy to set up. In many cases *NetVault* is probably guilty of giving you too much information. However, once it is setup, the system is able to deal admirably with complicated networks and convoluted backup requirements. With plenty of reporting tools and monitoring options, media management and extensive logs, you should be able to determine just what state your backups are in at any time. **LXF**

Linux Format VERDICT

Documentation	7/10
Ease of use	6/10
Features	9/10
Performance	8/10

A thorough and competent backup solution for demanding networks.

LinuxFormatRating
 ■■■■■■■■ 8/10

Tutorial Professional IPTables

INTERNET SECURITY

Mastering IPTables



Security chief **David Coulson** shows you how 2.4's new Iptables features can be used to keep out unwanted crackers.

When Linux 2.4 was released, most people focused on what it would do to help the average Linux user and talked about the USB support, firewire, PCMCIA and DRI. While these are great additions to the kernel for the majority of people, often one of the major improvements over 2.2 was overlooked, even though it applies almost as much to Joe (and of course Jane) User as it does to a hardened network engineer. This is, of course, the inclusion of the 'netfilter' system into the kernel, which provides packet filtering

different, it's worth understanding exactly what each of them do and how we should view the organisation of the firewalling code in the kernel.

Netfilter is the system compiled into the kernel which provides hooks into the IP stack which loadable modules (iptables is one) can use to perform operations on packets. As netfilter uses modules for the filtering, you can use an ipchains module to provide exactly the same capabilities as the kernel level ipchains code in 2.2, or even the module for ipfwadm from 2.0. Netfilter is there all of the time, as long as it is compiled in, whether or not you are using any firewalling modules at all.

IPTables is split into two parts; The user-space tools and the kernel-space modules. The kernel-space modules are distributed with the main kernel, and you compile them as you would any other module, be it sound drivers, a filesystem or USB support. There is the main ip_tables module, as well as modules specifically for NAT, logging, connection tracking and so on. These modules perform the appropriate function on the packets which they get sent by netfilter, depending on the rules which they have in their rule-list, or chain.

The user-space iptables code comes in the form of a binary called 'iptables', which is distributed separately from the main kernel tree, and is used to add, remove or edit rules for the modules. This is comparable to the ipchains binary in 2.2. Often, when referring to iptables, it is assumed to mean the iptables binary, and we will continue to use such a standard here.

Configure, compile, install, reboot

Ideally, you need to have a machine which is already running a 2.4 kernel, or have the knowledge to install 2.4 on a machine currently running 2.2., as the required updates to make sure 2.4 runs without problems are outside the scope of this article. This machine is going to be for mission critical routing, so the use of the latest bleeding edge kernel is not really necessary; all we need is something which is stable, secure and is not going to corrupt our filesystems.

Aside from all of the other options which you may or may not need, there are numerous settings under 'Networking Options' which don't directly pertain to iptables, but are applicable to many features of it. Firstly, we need to select 'Network Packet Filtering', which basically enables the use of netfilter, although unless you're intending to become a netfilter developer, you won't need the debugging option. You will probably also want to enable 'IP: advances router' and 'IP: use netfilter MARK value as routing key'. We next need to compile

```
technoir@echaid:/usr/src/linux (pts/3)
technoir:/usr/src/linux# more /proc/net/ip_conntrack
udp      17 158 src=10.1.1.1 dst=10.1.1.1 sport=50488 dport=739 src=10.1.1.1 dst=
=10.1.1.1 sport=739 dport=50488 [ASSURED] use=1
udp      17 164 src=10.1.1.4 dst=10.1.1.5 sport=50493 dport=689 src=10.1.1.5 dst=
=10.1.1.4 sport=689 dport=50493 [ASSURED] use=1
tcp      6 64197 ESTABLISHED src=10.1.3.3 dst=212.108.212.228 sport=1329 dport=6
699 src=212.108.212.228 dst=194.222.178.203 sport=6699 dport=1329 [ASSURED] use=
1
udp      17 148 src=10.1.1.4 dst=10.1.1.5 sport=50487 dport=689 src=10.1.1.5 dst=
=10.1.1.4 sport=689 dport=50487 [ASSURED] use=1
udp      17 19 src=10.1.1.4 dst=10.1.1.5 sport=50468 dport=689 src=10.1.1.5 dst=
=10.1.1.4 sport=689 dport=50468 [ASSURED] use=1
tcp      6 50 TIME_WAIT src=10.1.1.4 dst=10.1.1.5 sport=40530 dport=5666 src=10.
1.1.5 dst=10.1.1.4 sport=5666 dport=40530 [ASSURED] use=1
tcp      6 431989 ESTABLISHED src=10.1.1.5 dst=198.186.203.68 sport=7715 dport=5
000 src=198.186.203.68 dst=194.159.178.188 sport=5000 dport=7715 [ASSURED] use=1
udp      17 9 src=10.1.1.4 dst=10.1.1.2 sport=50492 dport=111 src=10.1.1.2 dst=1
0.1.1.4 sport=111 dport=50492 use=1
tcp      6 431999 ESTABLISHED src=10.1.254.2 dst=131.211.28.48 sport=1034 dport=
6667 src=131.211.28.48 dst=194.159.178.189 sport=6667 dport=1034 [ASSURED] use=1
udp      17 38 src=10.1.1.1 dst=10.1.1.1 sport=50471 dport=32798 src=10.1.1.1 ds
--more-- (0%)
```

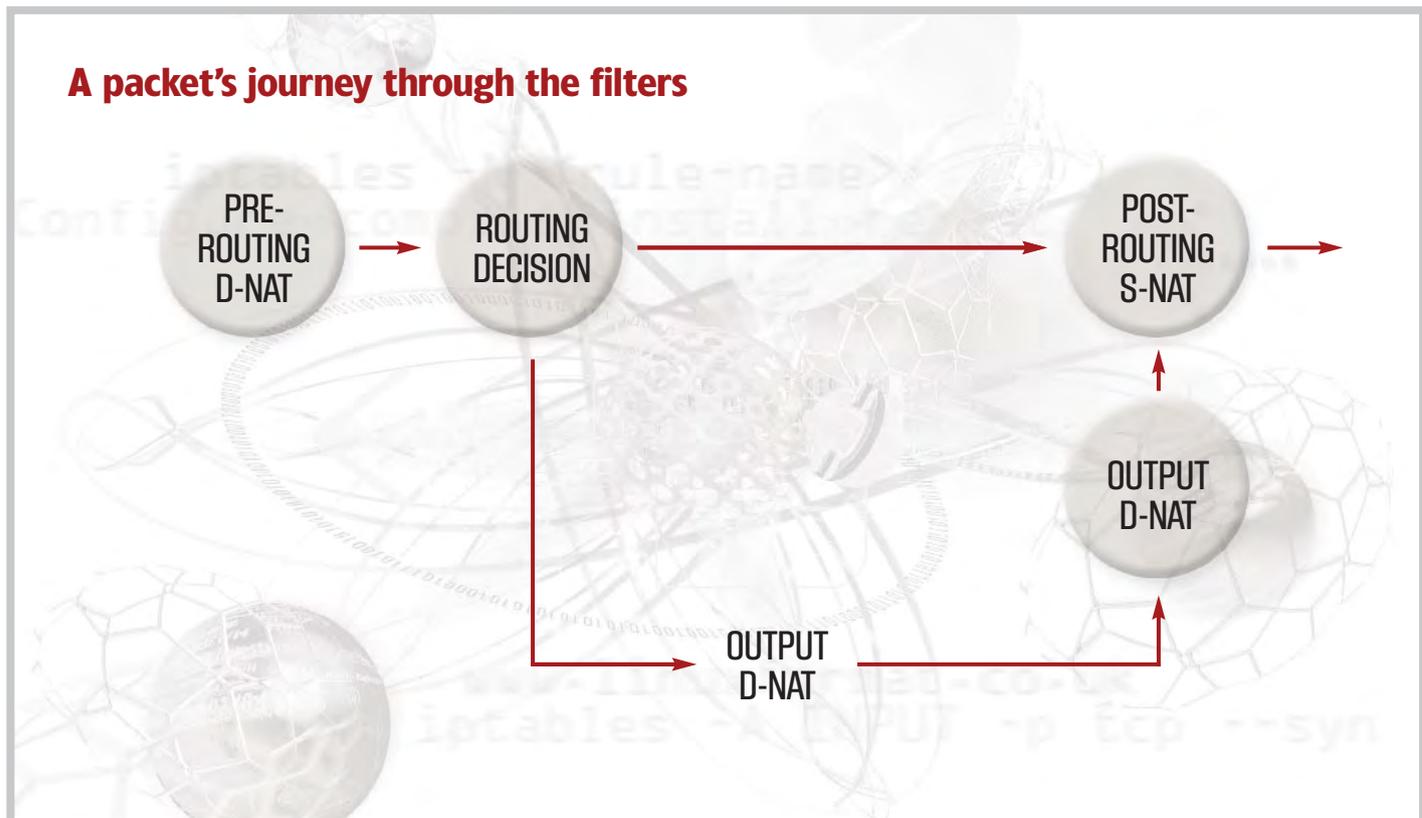
Conntrack means you can find out which connection specific packets are associated with.

and other more advanced IP features. Along with 'netfilter' comes 'iptables', which is the 2.4 equivalent of ipchains, and provides a user-space interface to the filtering, Network Address Translation (NAT) and mangling modules.

We're going to look at building 2.4 with support for netfilter and iptables, then building a production level router out of it. For those of you who just have one machine, and use it to connect to the Internet, then many of the same rules apply. The Internet is one giant, generally unrestricted, network which any reasonable person would have reservations about putting any sort of machine on, never mind their own Linux system.

Netfilter or iptables?

Often when referring to the firewalling code in 2.4, it will blindly be referred to as 'netfilter' or 'iptables', without any justification for using the specific name for it and, given that they are both very



some modules which netfilter can use, with the 'IP: Netfilter Configuration' sub-menu. Everything there needs to be selected as **m**, apart from the 2.2 and 2.0 support, unless you specifically need to use ipchains or ipfwadm on the machine while you learn to use iptables.

As with any kernel rebuild, **make dep && make clean && make bzlibo && make modules && make modules_install**, then reboot, assuming you are using lilo.

IPTables

Once the new kernel is up and running happily, we can go ahead and compile the userspace iptables tool. You can download this, the latest release being 1.2, from <http://netfilter.kernelnotes.org/>. It is basically a matter of doing **make; make install**, as root, and everything is sorted out. You will need a configured 2.4 kernel available for iptables to compile against, so if you've not yet built 2.4, or have deleted the source code, you might want to take a few steps back and have another go.

Next we need to load the `ip_tables` module into the kernel using **modprobe ip_tables**. A lot of the other modules are loaded automatically as we use the various features, but both `ip_nat_ftp` and `ip_contrack_ftp` need to be loaded manually, and we will look at their usage later.

Filtering

As with ipchains, iptables has three lists of rules – or chains – for filtering. For those of you who are confused about moving from ipchains, they have exactly the same names, but have to be in upper case, so there is INPUT, OUTPUT and FORWARD. INPUT applies to all packets destined for the local machine, OUTPUT for packets which originated locally and FORWARD for packets which are sent to our machine, but are not actually for it.

We can, if we choose, create our own chains to organise our

rules into different groups based on other rules. We create a rule with **iptables -N <rule-name>** and delete it with **iptables -X <rule-name>**. After this, they behave just like the three default chains, and we can flush them with **iptables -F <rule-name>** or list their rules with **iptables -nL <rule-name>**.

Using iptables we can perform three actions on the chains which alter their rules. We can either add, insert or delete rules, using **-A**, **-I** and **-D**, respectively, followed by the chain name. So, if we wanted to add another rule to the end of the INPUT chain we would use **iptables -A INPUT**. Not much use so far, as we need to specify which packets we want the rule to apply to. Matching source and destination IPs and ports is the most straight forward things to do. If, for example, we want to block all connections to port 23, over tcp, to a local machine we would do:

```
iptables -A INPUT -p tcp --dport 23 -j DROP
```

-p sets the IP protocol used, be it TCP, ICMP, UDP or one of the other more unused protocols, and **--dport** specifies the destination port of the packet. We can, of course, use **--sport** to specify a source port, but that is rarely used as connections usually use a random source port, unless they are from a specific service, such as NTP or BIND which has packets coming from a specific port.

Those who migrated from ipchains will be familiar with the difference between DENY and REJECT. However, the people who wrote iptables thought that DENY and REJECT sounded like the same thing, so there is now DROP and DENY. DROP literally drops the packet without making any effort to clean up afterwards, whereas DENY drops the packet, then returns an ICMP packet to the source of the packet to tell it that the connection was denied.

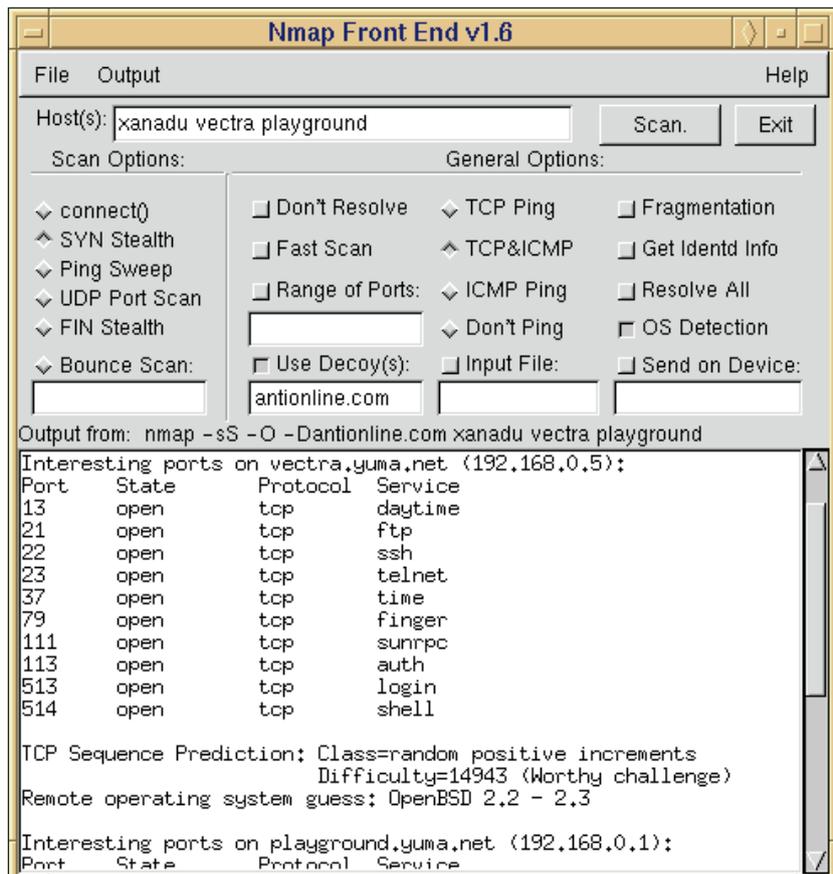
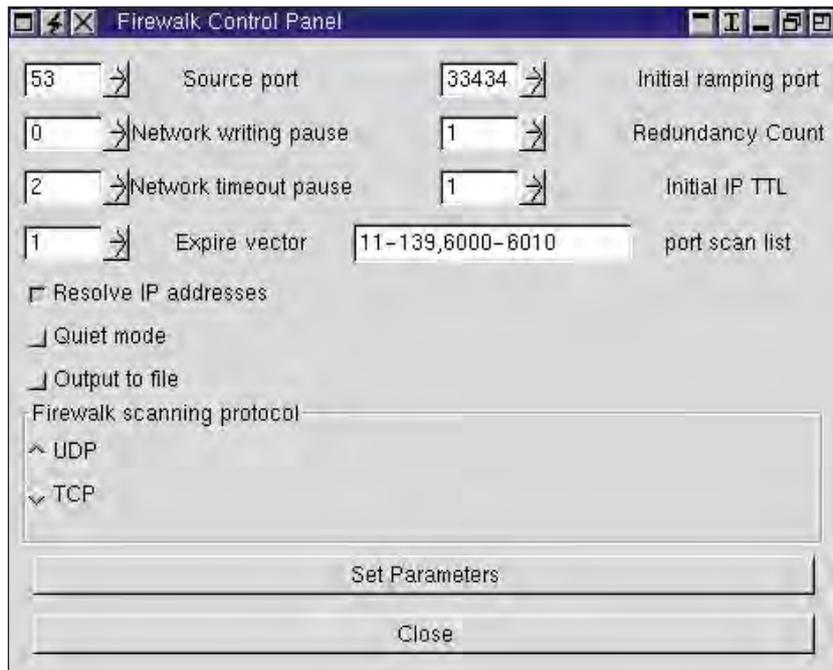
Describing source and destination IP addresses is often used to distinguish between trusted and unknown networks, and there are a number of different ways to refer to IPs and network

When a packet reaches a junction, it is examined to see if any action should be taken or if it should move onto the next chain.



Tutorial Professional IPTables

◀ addresses with iptables. Within the rule, we can use **-s <ip-addr>** and **-d <ip-addr>** to set the source and destination IPs which must match for the rule to be used. We can either use a normal IP, such as 10.1.2.4, a hostname, such as mail.domain.com or a network address 10.1.2.0/24 as an example. In the latter



There are a variety of tools and utilities to test the robustness of your network security – using tools like nmap you can find out what crackers can tell about your server.

case, it will use either the common slash notation or a proper network address/netmask identifier, and an IP is of course really a /32. If we neglect to use either **-s** or **-d** it will use 0.0.0.0/0.0.0.0 which will match any packet.

Often, we want to drop all internal traffic coming in from a remote network, such as the Internet, and this can be done with a combination of the **-d** flag, and **-i** which refers to the input network device, such as ppp0 or eth0:

```
iptables -A INPUT -d 10.0.0.0/8 -i ppp0 -j DROP
```

INPUT will only understand **-i**, and OUTPUT **-o**, as neither will have a device of the opposite type, but a rule in FORWARD can use both **-i** and **-o**, as it is not unlikely that a packet will come in one interface and go out of another, depending on the routing.

Previously, we would check for the SYN flag, which is usually indicative of a packet which is going to start a new connection, in order to prevent incoming connections to a machine.

Unfortunately, this is not a particularly secure way to do it, as it is fairly straightforward to create software which starts connections with malformed packets, and even if you can't do that, there are plenty of things you can download off the Internet which will do all the hard work for you. IPTables has a far better option, in the form of connection tracking modules. Every time a new connection is created, either locally, or by routing through our machine from somewhere else, ip_conntrack catches it and stores the details, so it can use the information to see which connection specific packets belong to. Now, rather than just checking for the SYN flag, we can check to see if it matches a currently established connection, which is much neater.

There are four types of connection which can exist. NEW corresponds to packets which are being used to create new connections. This is, of course, done by checking the connection tracking list, rather than checking any packet flags, so will apply to new connections being routed through our machine. ESTABLISHED relates to packets from a known connection, and RELATED applies to packets related to a active connection, such as an ICMP reply, or via the use of ip_conntrack_ftp, active FTP sessions. Last, but by no means last, is INVALID which should be dropped and are malformed or unrecognised packets.

All this matching is done with the **-m** switch, and for connection tracking, or stateful matching, we use **-m state** followed by a **--state** option, then list the packet types we do, or don't, want to match.

If we, for example, wanted to drop all NEW or INVALID packets coming in ppp0 we would use:

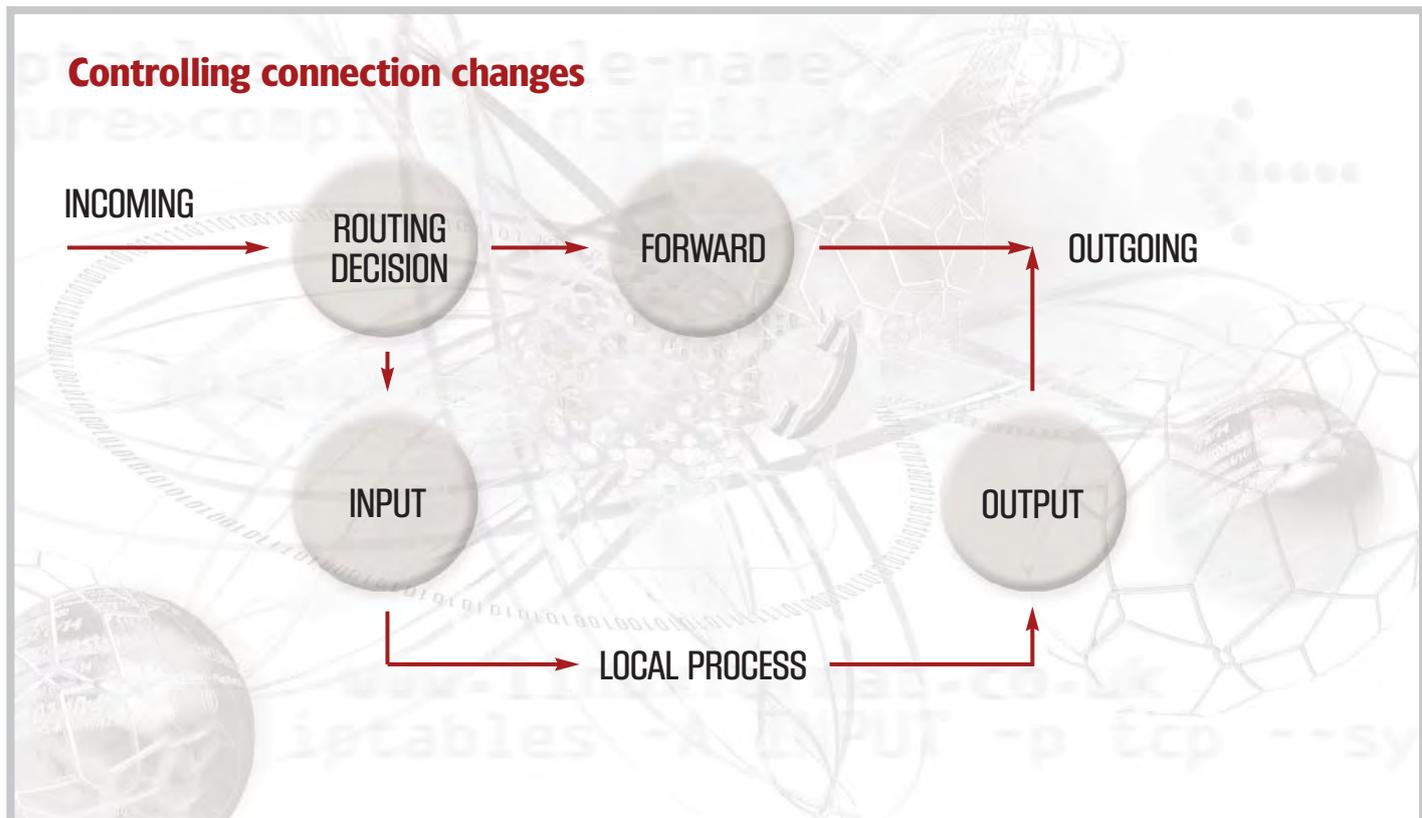
```
iptables -A INPUT -m state --state NEW, INVALID -j DROP
```

Sickeningly easy, isn't it? We can have it match packets which do not match a specific state:

```
iptables -A INPUT -m state --state ! INVALID -j INCOMING
```

Further info

If you want to find out more about firewalls and the sort of attacks they are subjected to, there are plenty of places on the Internet (surprisingly) where you can find out more info. One port of call should definitely be www.packetfactory.net, which is home to a great many network tools and information. Home of the Hack FAQ and a few more tools of the hacking variety, the Nomad Mobile Research Centre can be found at www.nmrc.org. If you want to test just how secure your new firewall is, you might want to check out nmap. This utility uses a variety of port scanning, fingerprinting and various other methods to help spot any vulnerabilities in your server. Find it at www.insecure.org.



Which would match all non-INVALID packets, then start to match them against rules in the INCOMING chain.

Matching specific flags of TCP packets is still important, so we can still check them using the `--tcp-flags`. This is slightly different, as it takes two options. Firstly, it needs a list of all flags it should check, then a list of the flags which should be set. If we wanted to perform a check for a 'SYN packet', that is, a packet with the SYN flag set, we would do;

```
iptables -A INPUT -p tcp --tcp-flags SYN, ACK, FIN SYN -j DROP
```

This translated into English says: drop all packets which have the SYN flag set, and the ACK and FIN flags not set. All other TCP flags are not checked. The above is provided as a single option `--syn`, so:

```
iptables -A INPUT -p tcp --syn -j DROP
```

is exactly the same.

The `-m` flag can match numerous other things, such as source MAC address, which is useful on a network where you only want trusted physical machines accessing services, but the most important match is the rate limiting, which is very useful for log messages, or limiting the connections on a machine.

`-m limit` is followed by `--limit`, which sets the rate limiting. If we use `--limit 1/s` it will allow one packet every second to match the rule. You will, however, notice that this doesn't work straight away. As default, it will allow the first five packets straight through, which is not always what we want. The `--limit-burst` option specifies the number of packets which can match the rule before it starts to limit the packets;

```
iptables -A INPUT -m limit --limit 5/m --limit-burst 10 --syn -j ACCEPT
```

This will allow the first ten packets to pass without any interruption, then limit to one packet every twelve seconds, or five per minute. Every time we hit a limit time, but a packet has

not passed, one is added to the current burst, so if we had ten packets, then none for a whole minute, it would allow five packets to pass through before starting to limit them again. After two minutes of no packets, the burst will be back to the beginning and will allow the first ten packets through again.

This is especially useful for logging, as we don't want our logs being filled up with loads of repeated information. Unlike ipchains, logging is done with a **LOG** target, much like **ACCEPT** or **DROP**. However, unlike the others, even if a packet matches a defined LOG rule, it will continue to transcend the chain, so you would normally have:

```
iptables -A INPUT -i ppp0 -m state --state NEW -m limit --limit 1/m --limit-burst 0 -j LOG
```

```
iptables -A INPUT -i ppp0 -m state --state NEW -j DROP
```

in order to log, then drop, all incoming connections on ppp0.

LOG will take two optional arguments, `--log-level` allowing you to specify a syslogd level, such as debug, info, etc, and `--log-prefix`, which lets you set a textual prefix to the log entry, up to twenty-nine characters, so you can easily distinguish which rule threw up the log entry.

NAT

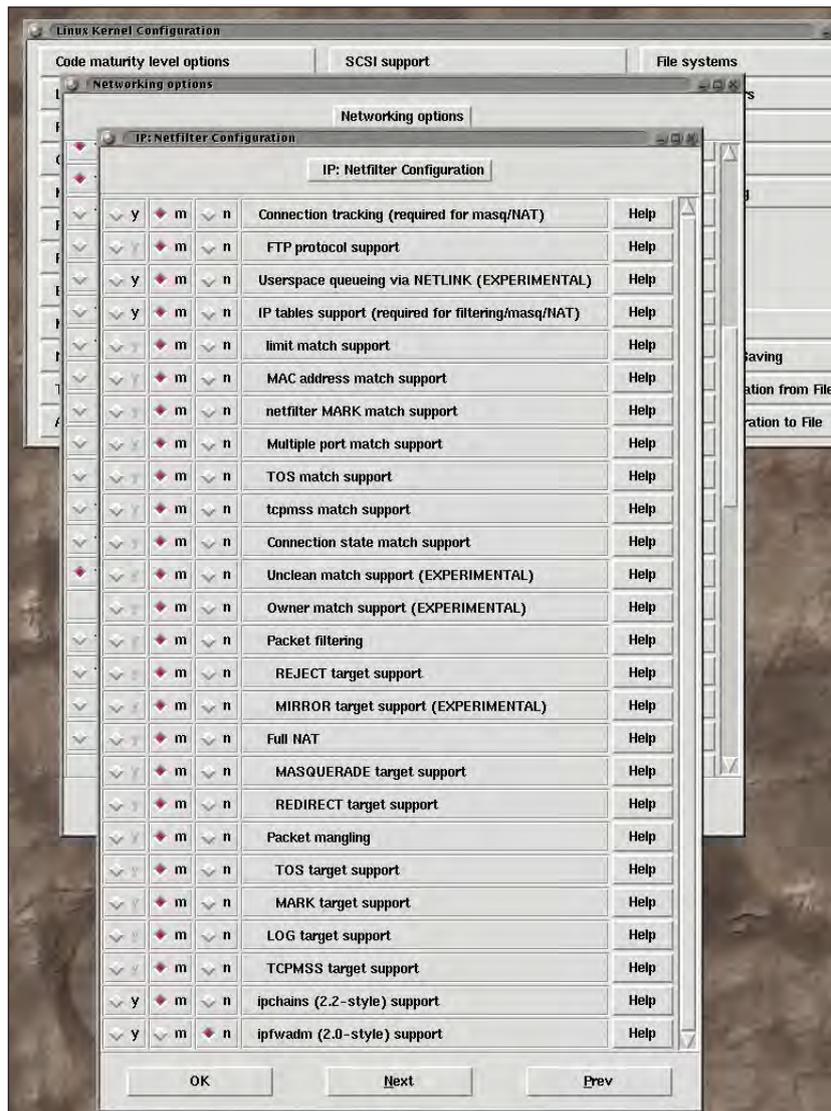
So far, we've looked at rules, chains, targets and matches, but as it is called iptables, there must be a table of some sort in there. Well, unknown to us, we've been using a table all along, although it is actually the default, so we didn't notice. 'filter' is the table used to filter packets, and contains the three chains, plus any ones created using the iptables utility. There are two other tables, nat and mangle, which also exist, and we're going to look at the nat table first, as it contains some of the most important changes to the 2.4 kernel over 2.2.

Firstly, we can list the chains in the nat table with:

```
iptables -t nat -nL
```

Each rule is examined against the three chains until a match is found.

Tutorial Professional IPTables



You will need to enable the new features of the kernel before you use them!

« which will throw up PREROUTING, POSTROUTING and OUTPUT. These three chains, much like the chains in filter, only apply to certain packets, although they are a little broader. Packets pass through the PREROUTING chain when they enter the machine, whether they are destined for the local machine or for somewhere else, before any routing decisions are taken by the kernel, so it doesn't know where they are going. OUTPUT corresponds to any packet originating from the local machine, and POSTROUTING to any packets leaving our machine, after the routing decision is taken, but did not originate locally.

We can, if we really wanted to, DROP, ACCEPT or LOG packets using these chains, but they are mainly used for Network Address Translation, or NAT, features. You might, at first, think that it does not apply to you, but IP masquerading is a type of NAT, so if you intend to share a network connection, it may be worth paying attention.

There are two varieties of NAT, source NAT or destination NAT. It doesn't take an experienced network administrator to work out that source NAT changes the source address, or port, of packets and destination NAT changes the destination information. Because of the way NAT works, source NAT, or SNAT, only works in the POSTROUTING chain, and DNAT in the PREROUTING or OUTPUT chains.

The most obvious reason to use NAT is to traffic packets from a public network, such as the Internet, onto an internal LAN, then back out again. We might want to have all SMTP connections to our router from the outside world forwarded onto our mail server on our LAN, and to do this we would use DNAT:

```
iptables -A PREROUTING -p tcp --dport 25 -i ppp0 -j DNAT --to 10.11.2:25
```

And, that is it. It will track packets coming in, and going out, so the outside world does not notice anything odd is going on, even if there are not masquerading or SNAT rules for the internal machines. The only caveat is that all packets for the connection must pass through the router, so you can't traffic packets from the mail server via a different machine to the outside world, as it will not know that it should reverse the DNAT rule when something leaves the network.

SNAT is used to hide internal IPs behind public IPs, which is not quite like masquerading. We might want to have all packets coming from 10.1.1.4 to correspond to a specific external IP:

```
iptables -A POSTROUTING -s 10.1.1.4 -o ppp0 -j SNAT --to 192.168.1.2
```

Of course, 192.168.1.2 is not a public address range, but it is just an example. What you can do with SNAT depends on the IP allocation, if any, from your ISP, so you may only be able to SNAT onto a single IP, but you can have many SNAT rules for a single external IP.

IP Masquerading is a form of SNAT, except that it is more interested in the interface than the IP address:

```
iptables -A POSTROUTING -o ppp0 -j MASQUERADE
```

will masquerade all packets going out of ppp0, just like in 2.2, but it is worth knowing what the difference is. Masqueraded connections are handled just like SNAT, until the interface goes down, at which point all connections are dropped, and you have to start again. Obviously, if you have a dynamic IP, you won't want old connections with the wrong IP address hanging around, as they won't do anything useful, but if you have a static IP then you suddenly lose the ability to resume TCP connections when you redial, as the router won't remember how it SNATed them the last time.

Problems are encountered when we SNAT onto an IP belonging to an interface which the packet will not be going out of, as SNAT only changes the packet, not the routing, in the POSTROUTING chain. This can be combated by changing the routing for specific packets, which we will cover next.

Mangle

The mangle table is a little strange, as it is used to change packet properties, which won't have a direct effect on them. As with nat, mangle has the same three chains which we can use to set packet properties, specifically marking them for later rules. Marking packets is especially useful if you want to use something outside of iptables, such as iproute2, to perform an operation on a specific packet, but it cannot match all of the options we need. If we wanted to mark all packets heading for a SMTP server with the number 1 we would do:

```
iptables -t mangle -p tcp --dport 25 -j MARK --mark 1
```

You might wonder exactly what the point of that is, but it is the first step in performing per-packet routing, as iproute2 can be used to setup routing tables based on 'fwmark', rather than the traditional destination IP as 'route' does. This is quite handy if you have a quick, but unreliable DSL connection, and a slow and stable ISDN line, but want all of your mail heading across the ISDN line.

Application

So, we now know most of the theory, but what about using it in practice? If we just have a single machine connected to the Internet and don't want to allow any incoming connections, but want to masquerade our LAN behind it we just do:

```
iptables -A INPUT -m state --state NEW,INVALID -i ppp0 -j DROP
```

```
iptables -t nat -A POSTROUTING -o ppp0 -j MASQUERADE
```

Not forgetting to do:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Notice that we don't need any ugly rules for active FTP, as we use `ip_conntrack_ftp`, which stops the incoming FTP connections being tracked as NEW. We might want to allow some ports, such as 113 and 25, for `identd` and SMTP mail delivery. We just do:

```
iptables -I INPUT 1 -p tcp -m multiport --dport 113,25 -j ACCEPT
```

`multiport` is a matching option which can be used to specify multiple ports within the same rule, either `--dport` or `--sport`.

To have this happen every time we reboot, we can use a combination of `iptables-save` and `iptables-restore` to save and restore the rules, or we can just write a bash script, or pop it on the end of `/etc/rc.d/rc.local`, depending on the distribution. Usually, if you're just starting out, it is best to use a script, as you can change it, rerun it, and you can be 100% sure that if you reboot the machine, it will end up how you have it now.

However, this isn't much use if you've got a couple of hundred machines behind a firewall and want to run proper web and mail servers, and allow internal machines to access the outside world transparently.

The simplest way to do this is to setup a selection of 101.x.0/24 networks, and put different classes of machines on them, such as front end servers, back end servers and workstations, as we will want to apply rules to each class in order to secure the network. The actual internal structure of the network depends on the services used, but it is not best to plug backend servers onto a hub along with a load of web servers. The public IPs will all be allocated to the public interface on the router, so the internal machines need not care which IP they are using, let alone how anyone gets to them.

Firstly, we need to take control of the IPs we are going to use, which is nothing more than making sure the public side of the LAN, which will probably consist of little more than a router for the line, knows where to send packets destined for the IPs we have. The quickest way is to setup IP Aliases for the network interface which faces the outside world, `eth0` in our case, so we might have `'eth0:mail'` as the interface for the mail server's IP. We could, instead, use `arp` to publish the NICs MAC addresses relationship to the IP with `arp -Ds 192.168.1.2 eth0 pub`, where 192.168.1.2 is the public IP. Which ever method is chosen, it will have to be performed whenever the machine is rebooted, so it should be inserted within the iptables setup script.

We will want all packets going to 192.168.1.2 to head for our mail server 10.1.1.2, and anything coming from 10.1.1.2, that is new connections, to look as if they are from 192.168.1.2:

```
iptables -t nat -A PREROUTING -p tcp -i eth0 -d 192.168.1.2 --dport 25 -j DNAT --to 10.1.1.2:25
```

```
iptables -t nat -A POSTROUTING -s 10.1.1.2 -o eth0 -j SNAT --to 192.168.1.2
```

All packets coming from our workstation network should only come out of one IP, which is easily done with another SNAT rule:

```
iptables -t nat -A POSTROUTING -s 10.1.3.0/24 -o eth0 -j SNAT --to 192.168.1.1
```

We can also drop packets from the PREROUTING chain, which makes it easy to drop all incoming connections to any machine, which does not have a specific DNAT rule, and as we're performing operations based on interface, rather than IP, we don't need to explicitly allow 10/8 traffic from being routed through our machine.

We hit a problem with this sort of setup, as if 10.1.3.2 hits our public IP for the mail server 192.168.1.2, the router translates it to 10.1.1.2, so the mail server gets a packet to 10.1.1.2 from 10.1.3.2, which won't travel back through our router in order for the DNAT to be reversed. This is quickly and easily combated with a SNAT rule, which will make all internal connections to any of our public IPs look as if they are coming from the router, and the DNAT will be reverse correctly:

```
iptables -t nat -A POSTROUTING -i eth1 -d 192.168.1.0/24 -j SNAT --to 10.1.1.1
```

Assuming our router has the internal IP of 10.1.1.1 on `eth1`. We can extend this further, to force all of the workstations to use a squid web cache which lives on 10.1.1.3:3128. A simple DNAT rule:

```
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -s 10.1.3.0/24 -j DNAT --to 10.1.1.3:3128
```

Squid needs a couple of options to perform correctly as a transparent cache, but those are well documented at <http://www.squid-cache.org>.

Conclusion

By now, we should have some variety of network running with iptables, and once you've spent time working out all its eccentricities, you'll wonder how you ever got along with ipchains. Even for a single machine connected to the Internet with a 56k modem, as a large proportion of people are, the sheer simplicity of its use makes it very difficult for even the most inexperienced user not to make the effort and having a go, assuming they can get 2.4 up and running in the first place. Simply, iptables offers many, many features which ipchains is technically incapable of, and when my own network is sitting on the Internet all hours of the day and night, I'm not going to pick second best. **LXF**

Netfilter GUIs

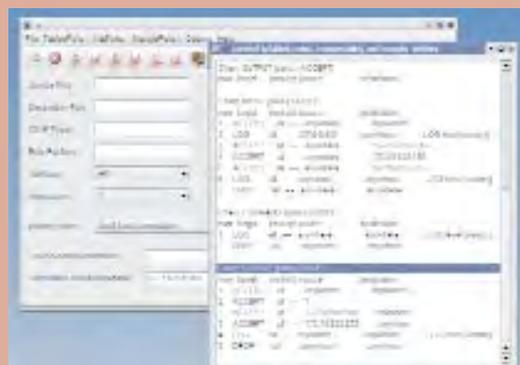
Once you understand how iptables works, you can use one of several GUIs to speed up the management of your firewall's rules. A good option, if you're running *KDE2*, is *knetfilter*. This is available as part of the *KDE2.1* distribution or you can obtain it from the project homepage at <http://expansa.sns.it/knetfilter>.

As well as allowing you to add, change and list your rules and policies, *knetfilter* provides integrated GUIs for *nmap* and *tcddump* for the easy monitoring of your network.

Another option is Solsoft's *NP-Lite*. This is not an open-source tool, but a freely downloadable version of Solsoft's commercial *NP* product. It is graphical firewall manager, implemented in Java, and translates a visual representation of your firewall policy into appropriate rules for iptables. See <http://www.solsoft.com/np-lite> for more details and expect a review next issue.

knetfilter is a useful addition to your security arsenal.

Finally, on a router you probably don't want to run X, so you won't be able to use either *knetfilter* or *NP-Lite*. A curses-based alternative is *ipmenu*, available from users.pandora.be/stes/ipmenu.html. It offers a menu-based interface for managing your firewall rules, and since it runs in a console, will work over a telnet or ssh link.



Coverdisc

Paul Ravening is your guide through the wonders of this month's jam-packed Linux Format CD. From firewalls to Hot Picks, there's something for everyone.



Important notice

Before you even put the CD in your drive, please make sure you read, understand and agree to the following: The Linux Format CD 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 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.

On the CD



Wherever you see this logo it means there's related stuff on the CD

>> Read me first

There's a job I've been putting off for ages now, and that's tidying my spare room. It's not that I'm lazy or un-houseproud or anything. It's just that the spare room seemed, well, spare. Seeing as it's been full of junk since I moved in two years ago, it was a huge task, and one that could be avoided no longer. Down among the duvets and old copies of *PC Plus*, I found my old 486 PC. Now I'm at a bit of a loss as to what to do with it. If past performance is anything to go by, it'll probably just end up gathering dust in the 'redundant technology' section in my house, along with my Spectrum +3 (though I might check out page 56 to relive my youth with that particular device) and my Psion 5.

However, there are better uses for your old kit. If you've got a redundant PC knocking about the house, and would like a bit more security on your system, look no further than SmoothWall, the excellent open source firewall distro. If you've been reading Linux Format since the year dot, then you might remember SmoothWall. We ran a version on the cover CD a while back. So many of you got in contact with the developers with suggestions, that we felt we should offer you the latest release, in essence designed by Linux Format readers. Well, in a way.

For those of you who don't know, SmoothWall is a self contained compact Linux distribution designed for

one task – to turn the host machine into a simple, low maintenance, easy to configure firewall system for home or small office networks. It's also Britain's most successful open source project.

Of course, there's more than just a complete firewall system on the CD. We've also got the latest version of Cincom *Smalltalk* 5i.3. This is a pure object-oriented language, which was developed at the Xerox Palo Alto Research Center. It is based on very simple concepts and a pretty straightforward syntax. This makes the system easy to learn. But don't think power has been sacrificed for the sake of simplicity. Oh no, this is also an extremely powerful language thanks to the robust class libraries included and the completely flexible nature.

We've also got *KDE 2.1*, which you non-DVD users missed out on last month, Hot Picks and more.

As usual, we welcome your suggestions for software. Email paul.ravening@futurenet.co.uk, with the subject 'Software Suggestions'

Thanks and Happy Browsing.

Paul Ravening
NEW MEDIA EDITOR



>> LXF14 May Disc contents at a glance...

SMOOTHWALL

A fully self-contained distro designed with one thing – and only one thing – in mind: to protect your computers from prying crackers. Read the warnings dotted all over the place before you use it. See page 91.

CINCOM SMALLTALK

Smalltalk is a very competent object-orientated application development system that claims to offer exceptional power without recourse to arcane syntax. See page 92.

KDE 2.1

We featured this on our highly popular DVD last issue, but we thought CD users should be able to get in on the act too. This new version includes KDevelop, anti-aliased fonts and a clutch of other improvements. See page 94.

GPHOTO

One problem many Linux users have is getting their digital cameras to work properly. However, GPhoto could be the answer to your prayers. See page 95.

PLUS: HOTPICKS:

As usual we've grabbed the best the Open Source community have to offer this month and shoved them onto the CD, to save you the cost of a huge download. Our Hottest Pick is Xine, an extremely promising DVD application that will bring Hollywood (or even Doctor Who) to your desktop. Or something. Check out the Hot Picks reviews starting on page 34, and then get installing.

SmoothWall 0.9.8

Dust off one of your old machines and put it to good use with the latest version of *SmoothWall*, the personal firewall system.

SmoothWall is a self contained compact Linux distribution designed for one task: to turn the host machine into a simple, low maintenance, easy to configure firewall system for home or small office networks. It also offers services such as IP masquerading, so any network device can use the *SmoothWall* box as a gateway to the Internet. The client machines do not even have to be running Linux, as all access to the settings and controls is via HTTP, any machine with a web browser can access it. All you need is a TCP/IP network to connect whatever machines you wish to use the *SmoothWall* box.

Warning!

Before you consider whether you want to install *SmoothWall*, please be aware that you will need to have a PC box dedicated to it. This software is not intended to be used on a dual booting system, and doesn't like playing with others. The idea is simply to take an old and otherwise redundant piece of hardware and turn it into what amounts to an Internet connection and security appliance. If you only have once machine, there isn't much point in installing it. Installing *SmoothWall* will wipe EVERYTHING from your machine. Hope that's clear enough for you.

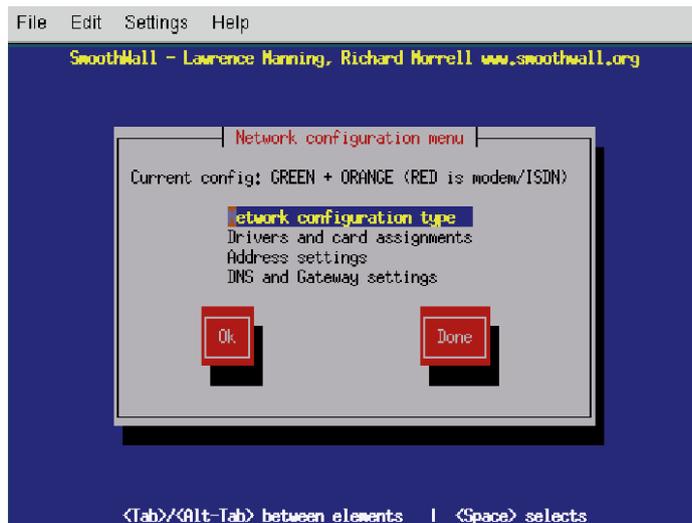
Installation

Step 1.

Proceed with great caution. *SmoothWall* will not happily co-exist with ANY other operating systems. It will trash them all. Completely! This product is meant to be run on inexpensive, outdated hardware, to use old bits of kit that are no use for anything else. The idea is to make a firewall appliance that you can just turn on and use

Step 2.

As long as step one is understood, then we'll carry on. The easiest way to install *SmoothWall* is to install it directly from the CD. You can, if you must, install it over a network via FTP



SmoothWall gives you warning before killing all your files.

What's new

So what's brand new in this version?

Have a look at this little lot:

- Auto probing and setup of ISDN hardware
- Auto probing and setup of up to three ethernet cards (DMZ support)
- Easy Port forwarding setup
- Proxy Server with caching facilities for small networks
- Updated DHCP server
- DNS server automatically setup
- Customised SmoothSSH Java client built in for system admin
- Revised installer
- IPSEC extensions
- FreeSWAN VPN Binaries and compiled code ready to VPN "out the box"
- Full graphical firewall and activity logging



You can configure *SmoothWall* from any web browser-equipped machine whether you're running Linux or not.

in which case check out the installation text in the /docs folder on the CD. If your machine can boot from CD, then pop your cover cd in the drive and turn the machine on. If it can't, you're going to have to make a boot disc, and that's where step three comes in.

Step 3

If you need to make a boot floppy, here's how to do it. Stick the *Linux Format* CD into a machine running Linux, and pop a blank floppy in the drive. Next open up a console and navigate to your CD-ROM drive with something like `cd /mnt/cdrom/` then type:

```
dd if=/path/to/cdrom/images/boot-0.9.8.img of=/dev/fd0 bs=1k count=1440
```

When the floppy disk has finished, slide it in the floppy drive of the soon-to-be-*SmoothWall* box.

Step 4

Insert the CD into the drive of the intended *SmoothWall* machine and turn it on. You will see a few messages scrolling past, then a message from *SmoothWall's* creators (it's important, please read it).

Step 5

Press return to continue booting Linux and go into the install process. You'll be asked what install medium you want to use. Choose CD and hit return. The installer will then ask you to insert the CD. Just hit Return, it's obviously inserted already. The installer will then automatically partition the first hard drive it finds. At this point, ALL data on this drive will be lost.

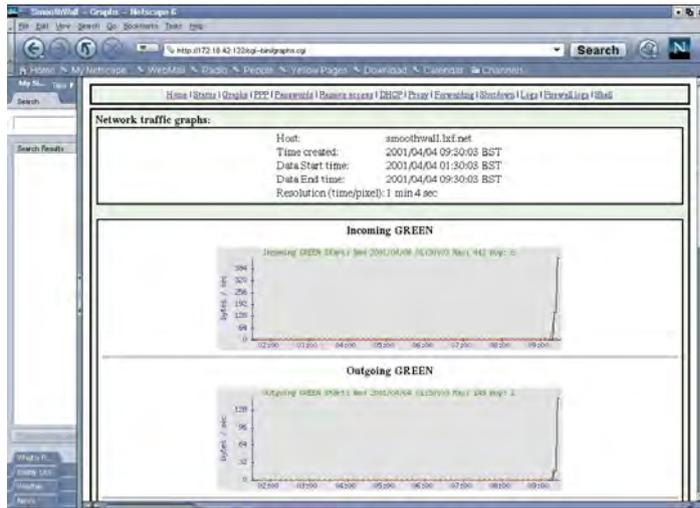
Step 6

After the basic partitioning, the installer will attempt to configure your network card. *SmoothWall* automatically recognises a large number of network cards, but if yours isn't detected then you'll be asked to enter its details yourself.

Step 7

Now you will need to setup your network information. We're just going to go over basic modem connection here, for more in-depth help, have a >>

LinuxFormatCoverdisc



SmoothWall provides a number of monitoring tools.

- ◀ look at the manual in the /docs folder. You will need to enter an IP address which is appropriate for your network.
- If you already have a network set up, use an IP address which fits with that. For example, assume our network is

using the 192.168 network prefix (which is one reserved for Class C internal networks), you might want to choose an IP address of 192.168.0.1 for the SmoothWall router. The network address would be 192.168.0.0 and the netmask would be 255.255.255.0. This would give you space for another 253 devices on the network from 192.168.0.2 to 192.168.0.254.

Step 8

Now the installer will ask you to set the root and admin passwords for the box, much like any distro. The admin user will be able to do pretty much anything, including reconfiguring the network settings on the box

Step 9

That's it, remove the CD and click on OK to reboot. When you get the LILO prompt, press return and SmoothWall will be up and ready for action.

Now it's installed, there's no need to keep the CD-ROM drive attached to the box, or even the monitor, you can probably use these more efficiently elsewhere. All other system configuration can be run across the network from any machine with a standard web browser.

Configuring

Once your *SmoothWall* box is up and running, you'll need to configure it with a few details (such as the phone number for your ISP, passwords etc.) *SmoothWall* uses an HTTP server to allow you to change these settings via the network using your favourite browser on any operating system.

Step 1

On a machine networked to your new firewall, run a web browser and point it towards <http://192.168.0.1> (or whatever IP address you gave the *SmoothWall*

Cincom Smalltalk 5i.3

Smalltalk 5i3 is a pure object-orientated language that enables you to create and distribute your own applications.



Are you interested in programming languages? Do you develop your own applications? Do you want to learn something simple and cool? Or do you need a software development environment that helps to build Internet application servers in a critical time-to-market environment? If your answer to any of these questions is yes, then try Cincom *Smalltalk*.

Smalltalk is a pure object-oriented language, which was developed at the Xerox Palo Alto Research Center (where the first real GUI was born). It is based on very simple concepts and a straightforward syntax. This makes it easy to learn. *Smalltalk* is also an extremely powerful language because of the robust class libraries included in *Smalltalk* environments and the completely flexible nature of a *Smalltalk* application.

How it's developed

An important difference between *Smalltalk* and other development environments is that *Smalltalk* is not developed in the same manner as traditional applications. In a traditional

development environment, developers use text editors to produce code that is then compiled, linked, and tested. In a *Smalltalk* system, however, developers work from an "image" which is a collection of available objects. These objects can be extended and new objects can be added to the system. However, unlike the traditional environment, the developer is free of the edit, compile, link, and debug cycles. This allows the developer to rapidly prototype and test ideas. *Smalltalk* developers typically work with browsers that allow the various objects to be viewed and modified. The collection can be readily searched to find objects, relationships, and implementations. This allows developers to be more productive and to have a better understanding of their source code.

Do it yourself

Cincom *Smalltalk* is the Object Orientated development environment from Cincom Systems Inc. For more than 30 years, Cincom has specialised

in managing customer relationships through adaptive e-business information systems. Cincom *Smalltalk* combines *VisualWorks* and *ObjectStudio* as two different environments in one bundle.

VisualWorks is a robust *Smalltalk* Integrated Development Environment (IDE) that allows you to build Internet application servers or client/server systems with instant cross-platform portability across Linux, Windows, Macintosh, and a variety of UNIX machines. (The CD only contains the Linux platform, other platforms can be downloaded from <http://www.cincom.com/smalltalk>).

VisualWorks ships with a number of open source components. The XML and XSL frameworks are used for source code representation within the environment. The *PostgreSQL* database and *StORE* connection is a community-developed component that extends the reach of *StORE*. *WikiWorks* is a tool for building user-editable web sites, and is used internally at Cincom for team communication. *DoME* is a modelling

tool that supports UML, and *JUN* is a graphics modelling toolset that supports OpenGL. The latter five components have been developed by members of the *VisualWorks* open source community.

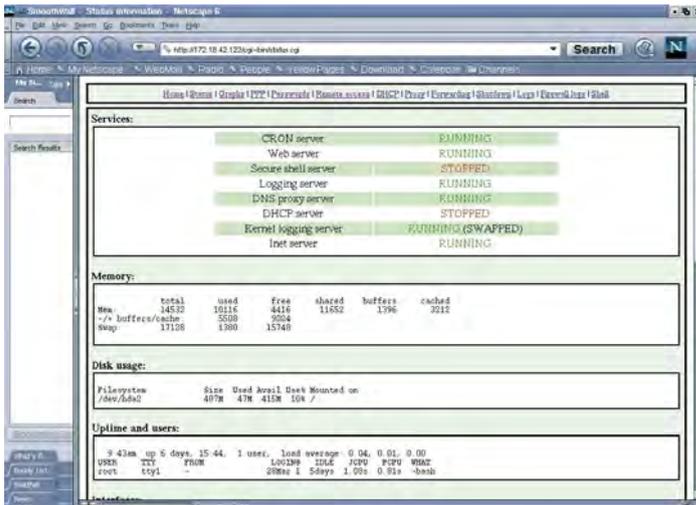
ObjectStudio is a complete, object-oriented rapid application development suite of tools for designing and assembling customised applications. *ObjectStudio* isn't included on the CD but can be downloaded from <http://www.cincom.com/smalltalk>

Have a go! Discover how simple it is to get going.

Getting Started

For new users, the *VisualWorks* environment has many features to ensure a smooth passage into the world of *Smalltalk*.

The welcome.pdf file is a good place to start. This can be found at www.cincom.com/visualworks/documentation.html. This explains all the other documentation available with *VisualWorks*. To read these documents, you will need Adobe Acrobat Reader



Everything in *SmoothWall* is done remotely through a web browser.

box) You should find yourself on the main configuration screen. You shouldn't need to touch the

SmoothWall box anymore, as everything – from setting up addresses to dialling out is done here.

Step 2

Go to the PPP setup page. You will be asked to enter your user name and password: use the "admin" name and password that you set for this. Then you'll get to the PPP page.

Step 3

Enter the configuration for connecting to your ISP. These will include the modem speed and port, the phone number to dial and your ISP user name and password.

Step 4

If you are using an unmetered line you may want to select persistent; this will automatically redial and re-establish the connection should it ever fail.

Step 5

Next select your authentication procedure. Many services use PAP or CHAP these days and entering your username and password and leaving the default settings should suffice.

Step 6

If your ISP uses script based authentication, the standard script used by *SmoothWall* should do the job in most cases (select it from the drop down menu). If it's a little more complicated, you can supply your own scripts. For further details see the configuration.txt file on the CD.

Step 7

Next you need to set up the DNS type. If your ISP supports automatic DNS configuration, you can just click on the appropriate button. Otherwise you'll need to enter the DNS information in the boxes below.

Step 8

Click on save and return to the main screen. There will now be a Dial and Hangup button on the main screen which will allow you to connect and disconnect the dialup network remotely from the webpage.

from www.adobe.com. Documents that are especially useful include:

- The Installation Document – A very valuable document for both new and more experienced users to ensure that your *VisualWorks* setup is correct.
- The Frequently Asked Questions Document – This document is good for troubleshooting. There are a number of ways for new users to become familiar with *VisualWorks*:
- The Walk-Through Document – A good place for new users to start. It is a short document that introduces the *VisualWorks* environment and enables you to quickly build a GUI-based application. It can be found in the "doc" directory of your *VW* installation.
- The Online Tutorial – This document assumes you know nothing about *Smalltalk* but a little something about programming.

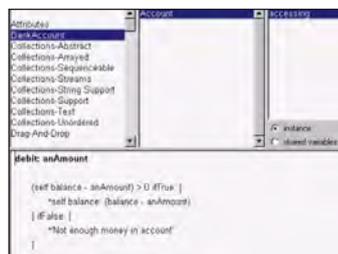
The tutorial will take you step by step in building an actual *Smalltalk* application:

www.cincom.com/smalltalk/tutorialform.html

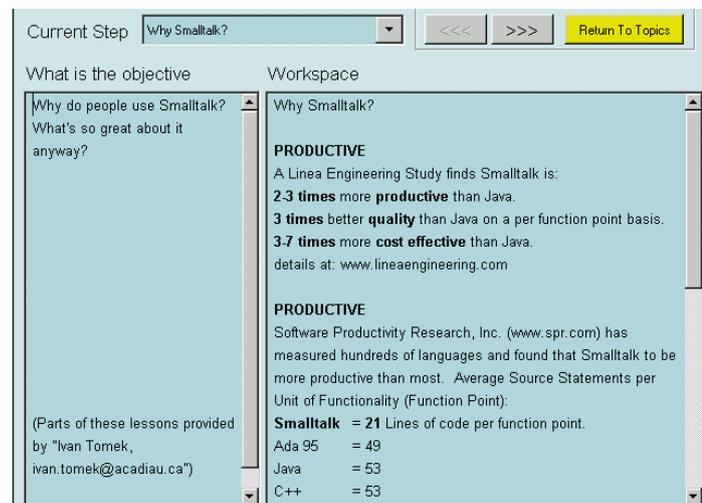
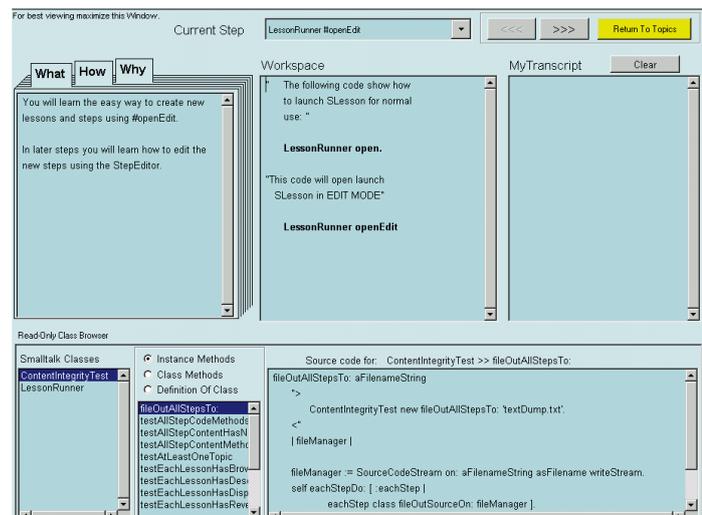
- The Lesson Browser – Included in the *VisualWorks* environment, the Lesson Browser walks you through the class library. Installation instructions can be found at: www.cincom.com/smalltalk/lesson-browser.html.
- The Application Developer's Guide – This guide describes the *VW* environment, tools, base libraries, and their usage. This is appropriate for developers of all levels, since it covers the fundamentals of the basic system.
- The Help Menu – When you select Help->Load Help in the *VisualWorks* Launcher, the Help Menu loads and opens automatically. From then on, select Help->Help Topics to reopen the help browser.

Contact Details

For comments/questions:
smalltalkmarketing@cincom.com
 For interest in commercial usage:
eurosmalltalksales@cincom.com
 Please visit the Cincom web page:
www.cincom.com/smalltalk and their page for Linux Format readers:
www.cincom.com/smalltalk/linuxformat



The system browser enables you to view different types of data



Why do people use Smalltalk? Be sure to check out Cincom's page on the history of the language, and use the provided lessons as seen above. >>>

« KDE 2.1

New, improved, better looking and finally here...

The KDE project has finally announced the release of *KDE 2.1*, the latest version of the powerful and easy to use Internet-enabled desktop for Linux. The second major release of *KDE 2* showcases a number of real improvements in terms of stability, performance and features. It can now lay claim to being a mature, solid, intuitive and, above-all, complete desktop for daily use.

KDE features *Konqueror*, a state of the art web browser that also doubles up as an extremely competent file manager, FTP client, picture viewer, SMB share browser...the list goes on. There have also been some great enhancements to *KMail*, *KDE*'s email client. The multimedia architecture has also made great strides, and this release includes the new media player *noatun* which has a modular, plug-in design for playing the latest audio and video formats.

The *KDE* package now features *KDevelop* for the first time. This is a central component of *KDE*'s powerful development environment. *KDevelop* is an outstanding IDE/RAD tool which will be instantly familiar to developers with Windows development backgrounds.

This release marks a leap forward

in Linux desktop stability, usability and maturity and is suitable for enterprise deployment. In short, *KDE 2.1* opens the door to more widespread adoption of the Linux desktop and will help provide the success on the desktop that Linux already enjoys in the server space. It significantly raises the bar for Linux desktop functionality, usability and quality in every aspect of the operating system.

How to install

To build and install *KDE 2.1* – if an RPM for your distro is not included – follow these instructions. Open up a console and for each package follow the commands listed here, substituting 'packagename' with the name of the package each time. Note that the packages should be built and installed in the order *kdesupport*, *kdelibs*, then *kdebase*. After that, the order is not especially significant.

Unpack a package with:

```
tar xvzf packagename.tar.gz
```

then change to the newly created package directory with:

```
cd packagename
```

and configure the package with:

```
./configure
```

By default the compiled packages will



The control centre can be used to configure the look and feel of KDE 2.1. Be sure to check out the anti-aliased font displays.



The internationalisation of KDE 2 is still continuing, with nearly twice as many languages supported that there were under KDE1.

KDE 2.2

The future's bright, the future's cog-shaped

Even though the dust has hardly settled and you might not of even got *KDE 2.1* integrated into your distribution yet, the distant rumblings of *KDE 2.2* can be heard far off in the distance. Yep, the *KDE* team have announced a revised schedule of future releases. *KDE 2.2* will hopefully be in final release on 16th July, a little later than originally planned. But then, it should be worth the wait. And I quote, "*KDE 2.2*, unlike *KDE 2.1* will come with new functionality and improvements in many areas of *KDE*". Part of the delay is due to wanting to get some much requested features into the next version, including IMAP support for *KMail*, better database support and a new printing subsystem. And with a bit of luck and a following wind, we'll have it on Issue 18 of *Linux Format*. Oh the anticipation!



be installed in `/usr/loca/kde`. If you want them to be installed elsewhere then use supply the `--prefix` option as a parameter to the configure script. Some packages (notably *kdebase*) have some special configuration options that might be applicable to your installation. Type:

```
./configure --help
```

to see these options. Finally, build the package with:

```
make
```

If you are logged in as root then you can install a compiled package:

```
make install
```

Otherwise issue the command:

```
su -c"make install"
```

If you encounter any problems during the build process please consult the latest version of the KDE Compilation FAQ which can be found at www.kde.org/compilationfaq.html.

Using KDE

Even though you can use most of the *KDE* applications simply by calling them, you can only benefit fully from all of them if you use the *KDE* window manager and its helper programs.

In order to make it easy for you, a simple script called *startkde* is provided. This script gets installed in `$KDEDIR/bin`. To use it, edit the file `.xinitrc` in your home directory (make a backup first) and remove everything that looks like calling a window manager. Insert *startkde* instead. Now restart the X Window System. If you use *kdm* or *xdm*, you will have to edit the file `.xsession` instead of `.xinitrc`. And if there's no `.xinitrc` or `.xsession` in your home directory, simply create one with just one line containing *startkde*. Doing this should finally present you with a shiny new *KDE* desktop.

gPhoto

gPhoto could get your digital camera talking to Linux for the first time!

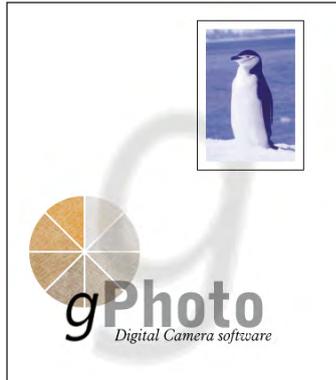
Digital cameras have the balance

of quality, features, and price that appeals to users looking for a fast and convenient way to get images into their PC. You can spend a lot more or a lot less, but you won't like the quality at the low end or the prices at the high end. With digital cameras, the best place to be is in the middle.

No other image-capture option for your computer matches the fun, immediacy, and convenience of a digital camera. Unlike with a scanner, you don't need a pre-existing image: A digital camera lets you capture what you see. And unlike film-to-disk services, development is instantaneous: You don't have to wait hours or days before you can use an image on your computer.

Operating a digital camera is also inexpensive, because the only recurring cost is batteries. The camera's storage is reusable once you download a batch of images. And since you don't have to pay for film processing, you can go wild.

Unfortunately, many digital camera manufacturers refuse to share their camera communication specifications with their customers and millions of



users who run free operating systems. This is a big obstacle to development of free photo applications, and it blocks free software developers from writing free programs, that support the same features as the proprietary digital camera programs for Microsoft Windows and MacOS that usually are shipped with the camera. It renders the camera quite useless if you don't run one of those proprietary operating systems. Luckily, a solution for all us Linux users is here.

gPhoto is a free, redistributable digital camera software application for Unix-like systems. It supports 105+ different digital camera models from

various vendors. See the boxout on this page to see if your digital camera is supported.

At the time of creating this issue's *Linux Format* cover CD, the version you have was the most recent. However, in the interim, the beta version of 2.0 has been released. You can download it from <http://www.gphoto.net/download.html>

Getting an interface

gPhoto 2.0 comes with a command-line interface that you can use to download pictures and control your camera. If the command-line seems a little counter-intuitive when dealing with your digital snaps, there are a couple of graphical frontends that can be used with *gPhoto*. You can choose from the following: *GnoCam* for the *GNOME* interface or *gtkam* for *KDE*. *GnoCam* depends on several very-recent *GNOME* packages. You can get it from the *gPhoto* Sourceforge CVS repository using the following method:

```
# cvs -d
:pserver:anonymous@cvs.gphoto.
sourceforge.net:/cvsroot/gphoto login
```

Press enter when prompted for a password.

```
# cvs -d
:pserver:anonymous@cvs.gphoto.
sourceforge.net:/cvsroot/gphoto -
z3 co gnocam
```

Next, compile and install it:

```
# cd gnocam
# ./autogen
# make
```

Become root then type:

```
# make install
# cd ..
```

gtkam

gtkam is also in the *gPhoto* Sourceforge CVS, and the instructions for getting it are similar:

```
# cvs -d
:pserver:anonymous@cvs.gphoto.
sourceforge.net:/cvsroot/gphoto login
```

Press enter when prompted for a password.

```
# cvs -d
:pserver:anonymous@cvs.gphoto.
sourceforge.net:/cvsroot/gphoto -z3
co gtkam
```

Then, compile and install it:

```
# cd gtkam
# ./autogen
# make
```

Become root and type:

```
# make install
# cd ..
```

You can now run *gPhoto* at the prompt by typing in: **gtkam**.

Now you just need import the pics and fire up *The GIMP* (or some other image editor) to get creative on your precious holiday photos! **LXF**

Is your camera supported?

If it's listed below, then it is.

Agfa ePhoto 1280	Epson PhotoPC 500	Mustek VDC-3500	Olympus C-400	Olympus D-400L Zoom
Agfa ePhoto 1680	Epson PhotoPC 550	Nick Click	Olympus C-400L	Olympus D-450Z
Agfa ePhoto 307	Epson PhotoPC 600	Nikon CoolPix 100	Olympus C-410	Olympus D-460Z
Agfa ePhoto 780	Epson PhotoPC 700	Nikon CoolPix 300	Olympus C-410L	Olympus D-500L
Agfa ePhoto 780C	Epson PhotoPC 800	Nikon CoolPix 600	Olympus C-420	Olympus D-600L
Apple QuickTake 200	HP PhotoSmart 215	Nikon CoolPix 700	Olympus C-420L	Olympus D-600XL
Barbie	HP PhotoSmart 315	Nikon CoolPix 800	Olympus C-800	Olympus D-620L
Canon Digital IXUS	HP PhotoSmart C20	Nikon CoolPix 880	Olympus C-800L	Panasonic Coolshot KXI-600A
Canon IXY DIGITAL	HP PhotoSmart C200	Nikon CoolPix 900	Olympus C-820	Panasonic Coolshot NV-DCF5E
Canon PowerShot A5	HP PhotoSmart C30	Nikon CoolPix 900S	Olympus C-820L	Panasonic DC1000
Canon PowerShot A5 Zoom	Hot Wheels	Nikon CoolPix 910	Olympus C-830L	Panasonic DC1580
Canon PowerShot A50	Kodak DC120	Nikon CoolPix 950	Olympus C-840L	Polaroid PDC 640
Canon PowerShot G1	Kodak DC220	Nikon CoolPix 950S	Olympus C-900 Zoom	Relisys Dimeria 3500
Canon PowerShot Pro70	Kodak DC240	Nikon CoolPix 990	Olympus C-900L Zoom	STV0680
Canon PowerShot S10	Kodak DC260	Olympus C-1000L	Olympus D-100Z	Sanyo DSC-X300
Canon PowerShot S100	Kodak DC265	Olympus C-1400L	Olympus D-200L	Sanyo DSC-X350
Canon PowerShot S20	Kodak DC290	Olympus C-1400XL	Olympus D-220L	Sanyo VPC-G200
Chinon ES-1000	Kodak DC3200	Olympus C-2000Z	Olympus D-300L	Sanyo VPC-G200EX
Directory Browse	Konica Q-EZ	Olympus C-2020Z	Olympus D-320L	Sanyo VPC-G210
Epson 3000Z	Konica Q-M100	Olympus C-2040Z	Olympus D-330R	Sanyo VPC-G250
Epson PhotoPC 3000z	Konica Q-M100V	Olympus C-2500Z	Olympus D-340L	Sierra Imaging SD640
	Konica Q-M200	Olympus C-3000Z	Olympus D-340R	Sony DSC-F1
	Minolta Dimage V	Olympus C-3030Z	Olympus D-360L	WWF

EDITORIAL

Editor Nick Veitch
nick.veitch@futurenet.co.uk

Art Editor Julian Jefferson
julian.jefferson@futurenet.co.uk

Reviews Editor Richard Drummond
richard.drummond@futurenet.co.uk

Production Editor Andrew Channelle
andrew.channelle@futurenet.co.uk

CD Editor Paul Ravening
paul.ravening@futurenet.co.uk

Editorial Contributors

Charlie Stross, Mike Saunders, David Coulson, Jono Bacon, Matt Kynaston, Jon Kent, Paul Cavanagh, Keith Pettit, Biagio Lucini, Chris Howells

ART CONTRIBUTORS

Photography Rick Buettner, Louise Parker, Gavin Roberts
Illustration Paul Bateman, Chris Winn

ADVERTISING

Senior Advertising Manager Brian Ainge
Deputy Ad Manager Michelle Blackwell: 020 7317 2602
Display Sales execs Lorien Dorking: 020 7317 2630
Matt Dalton: 020 7317 2622

Classified Sales exec Tom Denning

ADVERTISING DESIGN

Point of contact/Team Leader David Mathews
Designers Dan Yeo, S Hobbs, M Stapleton, C Stenner
Supervisor Sarah Orchard

MARKETING AND PROMOTIONS

Marketing Manager Gill Stevenson
Marketing Exec Liz Britton
Subscriptions Manager Julie Sewell

PRODUCTION

Production Manager Lou Reffell
Production Co-ordinator Ian Wardle
Ad Designer David Mathews
Group Production Manager Judith Green

MANAGEMENT

Publisher Sheena Pittaway
Group Publisher John Weir
Circulation Director Sue Hartley
Managing Director Mike Frey

DISTRIBUTION AND CIRCULATION

Circulation Manager Richard Jefferies
richard.jefferies@futurenet.co.uk

Distributed by Seymour Distribution, 86 Newman Street, London W1P 3LD
Tel +44 (0)207 9076000

Overseas Licences

Tel +44(0)1225 442244 **Fax** +44 (0)1225 732384

Contact Details

Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel +44 (0)1225 442244 **Email** linuxformat@futurenet.co.uk

Subscriptions and Mail Order

Phone +44 (0)1458 271178. See page 88
Email 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 a non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance, in writing.

All CD-ROM demos and reader submissions are supplied to us on the assumption they can be incorporated into a future cover-mounted 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 2001

GNU/LINUX is a trademark of Linus Torvalds, and is abbreviated to Linux throughout for brevity.

Future Publishing Ltd. is part of The Future Network plc.

The Future Network serves the information needs of groups of people who share a passion. We aim to satisfy their passion by creating magazines and web sites that offer superb value for money, trustworthy information, multiple ways to save time and money, and are a pleasure to read or visit. This simple strategy has helped create one of the fastest-growing media companies in the world: today we publish more than 130 magazines, over 45 magazine web sites and eight specialist web networks from offices in seven countries. The company also licences 52 magazines in 26 other countries.

The Future Network plc is a public company quoted on the London Stock Exchange (symbol: FNET).

Chairman Chris Anderson

Chief Executive Greg Ingham

Finance Director Michael Pennington

Tel +44 (0)1225 442244

www.thefuturenetwork.plc.uk

BATH • LONDON • LOS ANGELES • MILAN • MUNICH • NEW YORK • PARIS • ROTTERDAM • SAN FRANCISCO • TOKYO • WROCLAW



Next month

Stallman

Founder of the GNU project, libertarian and gcc author, Richard Stallman talks to *LXF* about GNU/Linux



»» Databases explored, explained and compared

An extensive look at the business of serving data

»» Is your scanner driving you inSANE?

Scanning for Linux – we show you how.

»» A real double-header

Run X on multi-headed displays? It can be done!

»» Sun's JDK AND FORTÉ on the CD, Plus loads more

Including KDE 2.1.1, Compupic graphics software, vital updates, the latest kernel and Hot Picks...

LINUXformat

Issue 15 on sale Wednesday 23 May