

LXF28

Linux Format June 2002

CYCLADES TERMINAL SERVER • VIDEO EDITING • MAC NETWORKING • KDE3 UNCOVERED • CROSSOVER OFFICE • CUSTOM KEYBOARDS • VOLITION

LINUX

FORMAT

www.linuxformat.co.uk

COMPACT
disc
DIGITAL DATA



DVD version also available

LXF28 JUNE 2002

THE UK'S BEST-SELLING LINUX MAGAZINE

2 CD
1 ISSUE!

MOZILLA

King of browsers!

We chronicle the epic journey to release, and the impact on open source projects worldwide **p46**



Crossover Office

Codeweavers WINE solution to office suite woes **p30**

KDE3 uncovered

Install & use from the discs – complete guide **p60**

Networking with Macs

Linux and Macs can coexist happily on your LAN **p54**

Video Editing Roundup

The top non-linear editing solutions on test **p34**



JUNE 2002

£5.99



0.6

(£5.99 outside UK and Republic of Ireland)

Another monster issue

Mozilla is more than a web browser. Most of you probably know that, but I'm sure you'll find plenty of other things you didn't know about the project, and the amount it has contributed to open source development, in our monster cover feature. It isn't every 1.0 release that would get a cover and an eight page feature, but Mozilla isn't any ordinary software, and represents more than just another web browser (good at that though it is).

The *Mozilla* project has gone on far longer than its originators intended, and ended up being more than was originally envisaged. The release candidate, which contains more code than the Linux kernel, is the culmination of an epic journey into open source development, which has thrown up many useful ideas, tools and sub-projects along the way. It may not be a model story of how to go about project managing open source development, but it has a lot to teach us. I encourage you to read and digest.

A monster in more ways than one. Mozilla analysis starts here. [p46](#)

Video editing on Linux? There's a surprising variety of software to choose from [p34](#)

Get the lowdown on the latest incarnation of the wonder that is KDE [p60](#)



On any other month it might have been KDE3 that dominated the headlines. Now in a stable release, this version appeared pretty rapidly after Qt3, and leverages the new features in the latter to create a slicker and faster desktop environment. As well as the source on CD and DVD, we have some distribution specific packages (from the respective vendors and contributors) and a four page guide for users and developers starting on page 60.

This month also sees the first of our Linux Pro 'mini' mags, which aim to provide some Linux content geared toward its real-world IT application. Do let us know what you think.

Personally, this month I have been delving into the world of video editing. It may seem straightforward, but believe me I've had plenty of battles with hardware, file formats and anything else you can think of. The good news is, hopefully you won't be so hampered if you take a look at the results in our roundup!



Nick Veitch EDITOR

LINUX FORMAT

Aims of the magazine

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

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

Meet Linux Format's team of writers...



Richard Smedley
Allegedly green fingered, our Prod Ed has a surprisingly large collection of dead potplants.



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



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



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



Maurice Kelly
Busy coder, electronic engineer and Midnight Oil fan, though he doesn't resemble anyone from the band.

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

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

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

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

Brian Long
Long time *Delphi* genius, Brian is also a dab hand with Borland's *Kylix*.

Contact us

Letters for publication:
linuxformat@futurenet.co.uk

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

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

Disc problems:
linuxformat@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 732295

More contact info on p114

CONTENTS

LINUX FORMAT

LXF28 June 2002

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

Video editing

In our roundup we wade through the maze of proprietary video codecs and standards, and investigate all the Linux apps, so you don't have to.



Linux on Macintosh

We take a look at how Linux interoperates with Mac systems – including a tour of Appletalk file and print services, and a study of filesystems.



KDE 3.0 desktop

As KDE launches its third generation desktop, to take advantage of improvements to the Qt toolkit, we show you why this had to be on our coverdiscs.

**COVER FEATURE**

MOZILLA

The legend. We investigate the birth and development of arguably the most important Open Source project ever

46





Newsdesk

Wine advances; Zaurus released; Kylix and Linux training; easy programming from IBM; Sage on Linux; Carrier Grade Linux; more office suites; new photo software; and the world's biggest supercomputer!

THE NON-EMULATOR RUNS MS OFFICE AND DIRECTX 8 GAMES

Vintage WINE success

The WINE project appears to inspire passion and ire with equal measure among Linux users, and the past few months have been a rollercoaster ride for the many projects which rely upon it.

Office politics

CrossOver Office from Codeweavers, which allows users to run Office97, Office2000 and Lotus Notes from Linux desktops without using Windows code, has been receiving a lot of press lately, which may account for a clause in the End User License Agreement of Office XP which states that the product can only be used on a Microsoft OS. CrossOver Office currently supports Word, Excel and Powerpoint with future versions offering a stable platform for Outlook, Access and Internet Explorer. www.codeweavers.com/products/office/the_real_dirt.php will tell you more about what works, what doesn't and which applications 'sort of' work.

Lindows ills

Codeweavers have also been involved in a spat with Linux startup Lindows which is aiming to bring a Windows/Linux compatible OS to the market. Lindows, essentially a Linux and WINE combination, was launched back in 2001 though the company, formed by MP3.com boss Michael Robertson, have yet to ship any tangible product. Both parties were reluctant to talk about the split – citing non-disclosure agreements – beyond confirming they would no longer be working together. Codeweavers were expected to contribute some of their proprietary



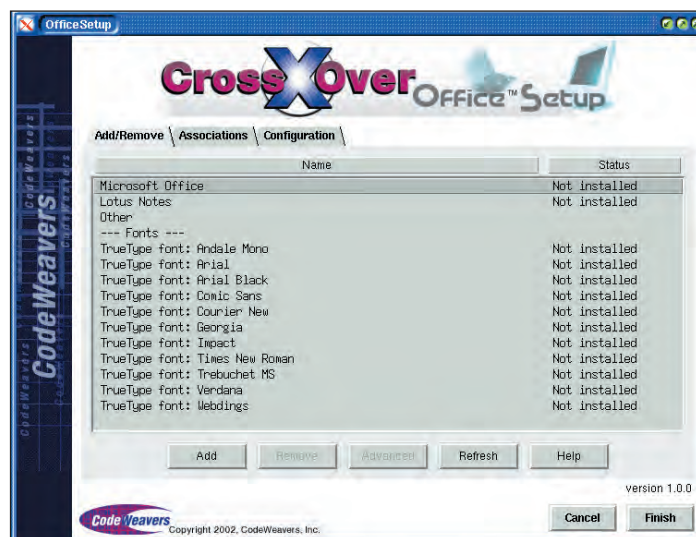
CrossOver Office reached the Holy Grail of emulators – running MS Office with no native Win32 components.

technology to the Lindows project.

Lindows has also incurred the wrath of the Free Software community for failing to include source code with its preview release of the distro. The Free Software Foundation was tipped off about the omission by one of the testers signed up for Lindows' \$99 beta programme. Michael Robertson said this was an early beta, the final distro would be fully GPL compliant.

2 can play that game

Transgaming Technologies released version 2 of WineX, the game-centric version of the non-emulator, and added a number of high-profile games to its compatibility list. The primary goal for this release, the company says, was a full implementation of Direct3D 8. "Remedy Entertainment's Max Payne – one of the most popular 3D action games ever – is TransGaming's first fully supported DirectX 8 title," the company said in a release. Version 2 also offers support for 3D sound and copy protected CDs



Now you can let yourself in for MS Office on your Linux desktop.

on Supermount systems. The latter improvement should mean games such as Max Payne and Diablo 2 work with 100% consistency.

Subscription to the Transgaming service, which allows you to download

the latest binaries and game patches, costs \$5 per month, but you can download and compile the source code yourself under the Aladdin Public License, though you won't have access to the wider features.



Trolltech's Eivind Thordsen, Andy Bush from Sun and Sharp's Mark Klein at the Zaurus launch in London.

LINUX POCKET POWER

Staying Sharp with Linux PDA



Sharp UK have just launched the commercial version of the Zaurus PDA that we featured back in LXF24. The device we gave you an exclusive detailed look at back then was the developer version, designed to seed the market with useful software in time for the main consumer launch. The strategy seemed to work, as Sharp UK ran out of developer units shortly after our issue went on sale, and Trolltech, whose *Qt/Embedded* forms the basis of the GUI on the device, recorded a huge uptake on their *Qt* programming competition (results due shortly).

The consumer Zaurus, the SL5500, is pretty much identical in terms of hardware to the SL5000D unit we looked at previously, but with the onboard SDRAM boosted to 64MB, to

enable more local storage.

Running on a 206MHz StrongArm, the Zaurus is fast enough for some demanding applications. Thanks to an updated JVM, Java applications are also incredibly fast. "Speed is not an issue anymore" Andy Bush of Sun's Vision Council told us at the launch.

The software shipping with the device has received a boost in the form of *HanCom Office*. This special embedded version features file-compatibility with common office software (such as *MS Office*), customised to run on this device.

With a retail price of £449, the Zaurus is also competitively priced against similar, non Linux devices. Check the High Street now, or wait for our full review!

PHOTO SOFTWARE

Second generation photo management

Version 2 of *gPhoto*, the open source Linux digital camera management software, is now available and the project has increased the vast number of cameras it is able to interface with. The command line tool now supports USB cameras and there are a number of graphical front ends available for it including *GnoCam* for GNOME, and *Kamera*, which is a *kio_slave* driver which allows you to

access the contents of your camera from within *Konqueror*.

<http://www.gphoto.org/>



NEWSBYTES

■ Xdrive Technologies, one of the more established online storage outfits, have announced support for Unix and Linux. Users can share and collaborate on documents held on Xdrives with *Mozilla*, *Netscape* or any Java enabled browser. This means your files will be available regardless of your OS/browser choice.

■ Caldera are attempting to remain within Nasdaq trading requirements by organising a 'reverse stock split'. This means that for every four shares a person/institution holds, it will receive one supershare. Caldera's stock has hovered around the 50 cent mark for a while, so the stock 'join' should bring the price up to about \$2. Companies are supposed to remain above \$1 per share to qualify for listing on the Nasdaq.



■ Dutch software house Not A Number have suspended development on *Blender*, the 3D design application, and instigated a corporate reorganisation. Until this 'process' is finished, the company has closed down its online store and download area.

■ In what many fear will be a rerun of the 'Winmodem' problem, Microsoft are urging PC manufacturers to adopt a 'more cost effective' software solution for wireless networking. Soft Wi-Fi would, of course, be integrated into the company's Windows product line.

■ Joy Yokley Goodreau has assumed control of the Linux Documentation Project and has begun a study to find ways of improving it. One of her major jobs will be formulating an LDP Style Guide to make writing and, perhaps more importantly, reading documentation easier.

■ Like KDE? Love *Vim*? Then why not visit www.freehackers.org/kvim and pick up *KVim*, which brings the pair serendipitously together. Also you might like to try *W3M* a Japanese text only browser that runs from an *Xterm* and, thanks to an arguably pointless patch called *W3M-IMG*, can render inline images. See <http://packages.debian.org/unstable/text/w3m-img.html>.

■ Alexander Bartolich has published a Linux Virus Writing HOWTO, which describes how to 'write parasitic file viruses infecting ELF executables on Linux/i386'. The intention is, of course, to show not only how these things are designed and propagated, but also how to detect and remove them.

David Cartwright

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



COMMENT

Translate my database!

“ Let's say I have a big *Oracle Enterprise* database running on a Sun server, and driving the central company systems. Now let's say I want to run a Web server, in a remote hosting centre, with live access to the data in the central database. Given the cost of putting another copy of *Oracle* on the website's server machines, I'd rather go for something like *PostgreSQL*, or maybe (gasp) *SQL Server*. I don't really want to be doing ODBC calls over some kind of WAN link, or a secure Internet connection, as it'll slow the website down; it makes sense to replicate the database tables between the central system and the remote location.

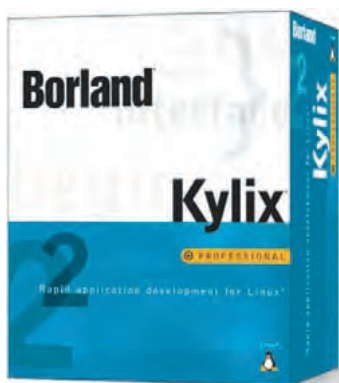
It seems, however, that the whole issue of replication between dissimilar databases is up in the air. True, there are packages that will do one-off database conversions, but I need live, real-time update. And true, I can go out and buy something that will replicate (say) *Oracle Enterprise* on Sun with *Oracle Standard* on Windows but that's going to cost me a shedload.

My plea to the Open Source movement is: let's look at issues of data interchange in real time. Plenty of sexy technologies coming along (SOAP, XML) reduce the complexity of making data source A talk to data source B, and there are potential apps out there just aching for a system that will allow them to swap data in a controlled yet generic and high-speed manner. Not easy, even deducing the structure is done differently in every database system you look at, but should the Open Database Replication Protocol working group arise, I'll be cheering.

EDUCATION OPPORTUNITY

On course with Kylix

Prospective and seasoned developers will get a chance to mine the knowledge of some of the country's foremost *Kylix* developers at a one day programming course supported by *Linux Format*, *Linux Magazine*, Borland User Group and eBooks.uk.net. The event is to be held in Hammersmith,



London on July 9th 2002. Tickets cost £66 (plus VAT) including refreshments and lunch. The day's program will cover many aspects of development from simple application building to tight integration between *Kylix* apps and SQL databases and websites.

The first session, kicking off at 10am, will be presented by LXF regular Brian Long – covering a basic introduction to the environment including visual components like edit box, buttons, memo box and using *Kylix* to launch Linux scripts/programs and to collect results.

Nigel Cohen will be using his experience as main architect of the *Kylix*-based eBooks.uk.net website to demonstrate the various aspects of database integration in the second two hour session, taking in data aware controls, SQL commands (from within *Kylix*), and SQL metadata controls

(from outside *Kylix*).

The day will be rounded off with a presentation by Allan O'Neill, core developer of the *Indy* Open Source component suite, on *Kylix* tools for collecting and responding to Internet requests with HTML. The session will include introductions to CGI scripts, **WebModule**, **HtmlPageProducer**, **HtmlTableProducer** and Actions; also

covering database updating, email client facilities and ftp client facilities.

Nigel Cohen, who is also managing the event, said the course was intended to be wholly practical. "It will be a very packed day, but also a very high output day," he said.

Find out more and book tickets at www.softwaredesign.co.uk/training/kylix/index.html

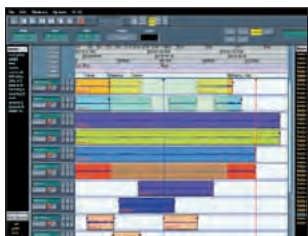
UKUUG addresses Linux developers 2002 UK Linux Developers' Conference

Another great education opportunity is the UK Unix Users Group's (UKUUG) annual Linux Developers' Conference in Bristol, 4-7th July.

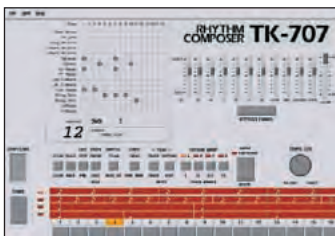
The highlights of the conference include "Running Linux on AMD's x86-64 Architecture"; The GNU Hurd; Bugzilla; "Building a Secure, Seamless &

Scaleable Wireless LAN"; the Linux 2.5 kernel; "Zope 3: Component Architecture with a Twist"; and "LEGO Programming with Open Source Tools" – as well as pre-conference tutorials on thin clients and on writing shared libraries (with *Glibc*'s Ulrich Drepper). www.ukuug.org/events/linux2002

LinuxWebWatch



You can turn your ear to anything with *Ardour*.



The Roland 707 was not as famous as its later sibling.



Slab really needs a decent MIDI editor if it wants to compete.



Taking on the hearts and minds of the music industry.

In quieter parts of the internet

No-one can hear you synthesise...

Most people don't realise the power of the PC sitting on their desk; you may watch video, create fantastic programs and imaginary worlds, but have you ever thought about the sound capabilities of your favourite OS?

For a start, this is not "just" about music, this is digital audio, and an integral part of most presentations, every film, and the occasional annoying website. So, when you look at *Ardour* (<http://ardour.sourceforge.net/>) and its ability to record 24 or more channels

of 32-bit audio at 48kHz, don't just think of how you'd mic up a drum kit, imagine capturing the raw emotion of a school play, creating your own aural adaptation of LOTR, or making bizarre answer machine messages. If you're even slightly serious about audio, check it out.

If you simply must be frivolous <http://www.vislab.usyd.edu.au/staff/chris/tk707/> contains a (mostly) stable soft drum machine based on Roland's precursor to the legendary 808.

You'll need an *ALSA* supported card and a working MIDI output to get it going, and the developers are keen on bug reports.

Slab (www.slabbexchange.org/) is yet another audio recording tool and consists of a virtual tape deck, a mixer, a wave editor and some audio processing tools. It's quite basic, but the application's Dutch developer appears to be quite committed.

Trying to muscle in on the music industry with a "new and revolutionary

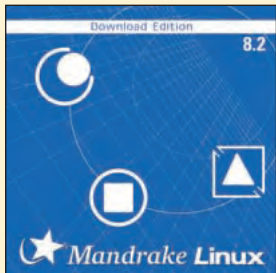
concept in the world of digital content copyrights" is a bold – and possible insane – move. But that's just what Open Music (<http://openmusic.linuxtag.org>) are attempting. The project provides a spectrum of licenses (similar in spirit to the GPL) for musicians to release their music under. There isn't a vast amount of music available on the site, but then if *LXF* readers decided to inflict their music on an unsuspecting world...

NEWSBYTES

■ The Joint Photographic Experts Group (JPEG) have announced plans for their next generation image compression technology. JPEG2000 is said to be as efficient as current JPEG standards but without any loss of image detail. www.jpeg.org

■ Xandros, the company which took control of the CoreLinux distribution when the graphics company got out of the OS business, are on the verge of their first release. Michael Bego has said the Xandros developers have been concentrating on perfecting the installation to make it foolproof. He said it would be one of the user-friendliest distros ever. Xandros is supported by Linux Global Partners, a venture capital operation focusing on Linux development.

■ Credit Suisse First Boston have completed migrating their massive worldwide financial trading system to Red Hat Linux. The system, capable of processing half a billion transactions per day, is based on high-powered Egnera BladeFrame systems and has enabled CSFB not just to reduce costs, but also increase performance. Amazon have also completed their migration away from proprietary Unix to Red Hat.

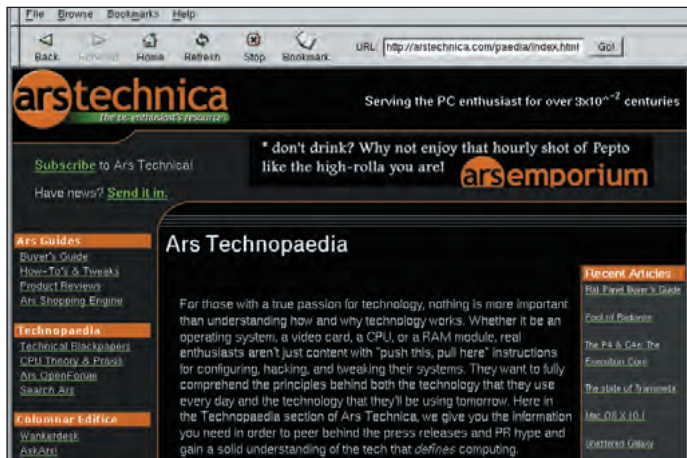


■ Mandrake 8.2 is now available in retail packs. The distro is available in three flavours, ProSuite, PowerPack – which both include StarOffice 6.0 – and Standard. There's also the usual download edition available from www.mandrakesoft.com

■ Despite the above successes and a slew of other enormous contracts, Red Hat has seen a steady drop in profits and, the newsobserver.com says, has launched a major sales offensive. "Red Hat now has 90 percent of its 630 employees working to lure corporations to Linux," the website reported.

■ Linux was quite visible at the National Association of Broadcasters Show in Las Vegas, which is no surprise as almost 80 per cent of Set Top Boxes run some form of Linux.

■ Symantec have developed a 'hardened' firewall for use with IBM's iSeries, er, series of servers. The application, based on the company's Enterprise Firewall for Windows and Solaris should be available in the third quarter of 2002.



SashXB's sample applications are a good place to begin your programming journey

EASY-PEASY IDE

Easier Linux app development

Borland's Kylix has brought Linux application development to the programming masses, but IBM are hoping to go one step further with Sash XB, an integrated development environment designed for those who wouldn't know a C++ class if it slapped them.

In fact, Big Blue claim anyone with a bit of HTML/Javascript knowledge can be programming 'fully featured' Linux apps within hours. The IDE uses a host of open source technologies including Mozilla, GNOME, Glade and Gdome and is easily extendible via a number of freely available components which cover things such as interaction with the filesystem.

The download also comes with a selection of sample applications (IBM insist on calling them Weblications) including a stripped down web browser, an FTP client and a text editor – each of which has been written in between 100 and 1000 lines of code.

Although complete in itself, IBM intends for Sash to be incorporated into Eclipse, a Java-based universal development tool 'dedicated to providing a robust, full-featured, commercial-quality, industry platform for the development of highly integrated tools'.

<http://www-124.ibm.com/developerworks/oss/sashxb>
<http://eclipse.org>

UK BUSINESS APPS PROVIDER

Serious business for Linux

Sage have announced an agreement with IBM and the availability of its Line 500 business management software for Linux. Sage, the leading supplier of business and accounting software to small and medium enterprises in the UK, used IBM's Informix database to bring its software to the IBM xSeries and iSeries server ranges.

The announcement marks a turning point for the adoption of Linux by small businesses: Sage is the first significant application provider in the

UK to support Linux. It proves that Linux can do more than just web, file and print-serving, that Linux is ready for traditional business applications. Adam Jolland of IBM commented, "This is a watershed for Linux. Up until now Linux has been used for IT infrastructure, but this is an application that you can run your business on."

Sage's Line 500 will be available for Intel-based servers via IBM's channels and as an integrated package with the IBM iSeries range. www.uk.sage.com

Jono Bacon

The founder of UK Linux and active KDE developer, Jono Bacon is a student in Wolverhampton who looks like Adrian Smith.



COMMENT

Predicting the past

“ I never cease to be amazed at what people say, do, or say they do – and none amaze me more than some Linux folk. Recently I have been reading a lot of interviews with various members of the Linux community, and pretty much every interview deals with one overriding factor – predicting where Linux is going.

One thing which I have learned as I have developed as a Linux user/developer and a hard nosed cynic, is that Linux can be what you want it to be – but it cannot be forced into any box. People have predicted such things as dominance in the server market, mixed reactions in the desktop world, the toppling of Microsoft and such things. These predictions are fair and well, but do we really have an idea where Linux is now?

Linux is a system with such diversity that I feel it has been over analysed by people in the press, academic types, and by students in Wolverhampton who write for Linux mags and who look like Adrian Smith from Iron Maiden.

This overanalysis is not one that I feel should be discouraged – the problem lies in how do you get an idea of trends and usage within the Linux world, when it's use can be customised dramatically, and it's usage is difficult to trace due to its open licensing.

One thing I would like to see is people throwing caution to the wind and just predicting that Linux will be there for everyone, despite what their requirements are and despite what their politics are. A prediction can only be made if the present is combined with a knowledge of the history. I predict that Linux is here to stay... for everyone.

LARGE INFRASTRUCTURE LINUX

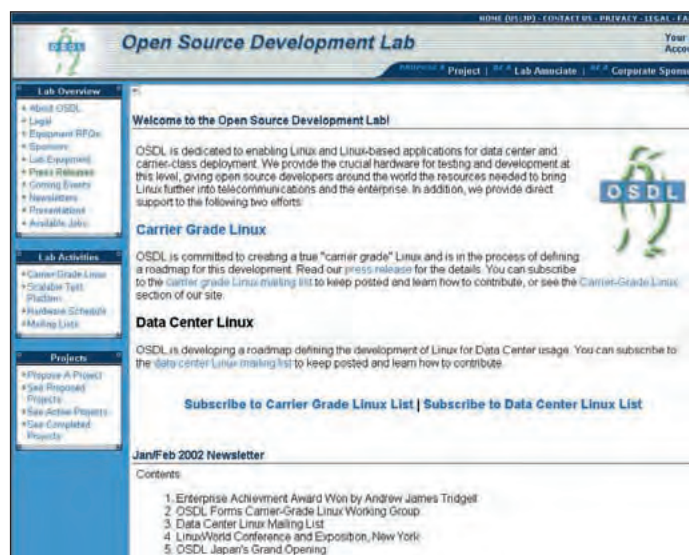
Carrier Grade Linux

A consortium of technology companies have joined forces with the Open Source Development Lab (OSDL) to define a new industry-wide specification for a Carrier Grade implementation of Linux. A Carrier Grade OS is designed for use in large scale, often public, telecommunications networks and needs to be extremely reliable – the name implies the system will have a downtime of somewhere between 30 seconds and five minutes per year.

The consortium includes IBM, Alcatel, Intel and Cisco, plus a number of well known Linux vendors including MontaVista, and their first task is to define exactly what's needed. OSDL's director Timothy Witham said IP Infrastructure currently relies upon a variety of bespoke, proprietary

technologies and that a new standard was needed to keep up with changes in the industry. "The large equipment and infrastructure providers have traditionally developed their own software platforms – which is why the OSDL's Linux-based initiative is so important, particularly in light of the rapidly evolving demands of new technologies such as multimedia communication services," he said. The group have set a furious pace so far, with the version 1.0 specification due to be published in August and products hitting the shelves before the end of the year.

Ari Virtanen, vice president of Nokia Networks, said it was important to be involved in the development effort as this Linux will be the basis for their next generation All-IP network infrastructure. "We want to ensure that Linux continues to support efficiently



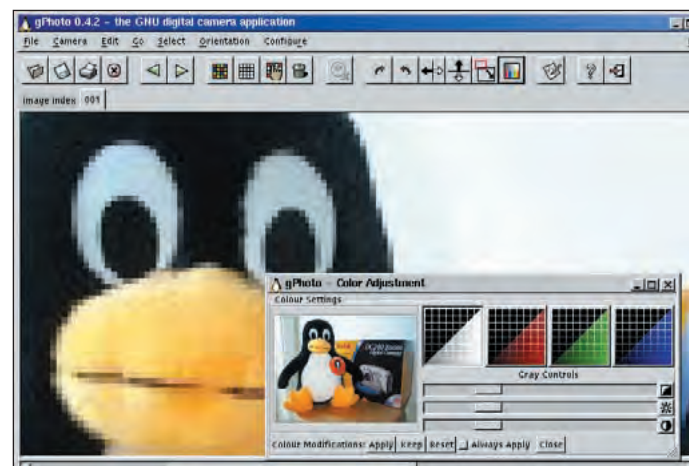
OSDL: "Defining a roadmap for the development of Carrier Grade Linux."

and in a uniform way the features required in carrier grade

environments," he said.
www.osdlab.org



ThinkFree is a cross-platform productivity suite with bundled online storage for easy file sharing and collaboration.



ThinkFree's user-based licensing is likely to win it as many friends as its MS Office file compatibility.

YET ANOTHER OFFICE SUITE

ThinkFree present another Office alternative

Joining the crowded cross-platform office market is *ThinkFree Office*, a productivity suite that combines a familiar feature-set (word processing, spreadsheet and presentations), MS file compatibility and an online storage solution in a single boxed or downloadable

application for Linux, Windows or Mac. Written in Java, *ThinkFree* is, say its developers, perfectly happy offline, but can integrate seamlessly with the company's secure online Cyberdrive storage meaning you can access the same files from any computer with an Internet connection, even a slow one.

This, allied to the user-based licensing (which enables users to run the applications on any machine at work, at home, or while traveling without the need to pay for additional licenses) may make the suite a good choice for enterprises with a large number of home workers.

There is also a Server Edition which pipes the suite onto every connected desktop PC and provides a range of file sharing and collaboration facilities.

The standard edition of the suite starts at \$49 (including 12 months of automatic online updates and 20MB storage space). www.thinkfree.com



The new system should speed up calculations by a factor of 30!

LINUX SUPERCOMPUTER

Faster and faster

The Pacific Northwest National Laboratory, a research arm of America's Department of Energy, has ordered a \$24.5 million Linux supercomputer from Hewlett Packard. Built around a staggering 1,400 Intel Itanium processors, the machine will provide a total of 8.3 teraflops of computer power – making it the most powerful Linux computer on Earth – and, HP claim, will mean that jobs which would have taken one month can now be completed in a day.

The system is due to be functional in early 2003 and will be used by PNNL scientists to study complex chemical problems in areas such as biological systems, subsurface transport, material design, atmospheric chemistry and combustion. "The new supercomputer will provide researchers from national laboratories, academia and industry with the capability to address very large and complex molecular-level computations associated with DOE's energy security

mission and environmental cleanup programs," said Office of Science Director Raymond Orbach.

HP's Rich DeMillo said the order demonstrated the company's focus on more open architectures. "This supercomputer is another validation of HP's service-centric technology vision, exemplifies the power and benefits inherent in the Itanium architecture and Linux, and clearly illustrates that there is more than one top player in the supercomputing market"

Embedded Linux News



● The Zaurus continues to attract the attention of Linux developers, with TheKompany being the latest software house to jump on the bandwagon. Highlights of the *tkc* suite (all available separately) include *tkcKapital* (a personal finance package), *tkcPainter* (a paint package) and *tkcPlayer*,

which provides facilities for playing MP3 and Ogg files on the move. The latter feature make the Zaurus the world's first portable Ogg player. <http://www.thekompany.com/embedded/>

● Lineo – who provide the underlying OS for the Zaurus – looked into the abyss recently, having to lay off staff and make a number of cutbacks, but has survived thanks to the faith of The Canopy Group, which now has a controlling stake in the company. Despite the financial woes, CEO Matt Harris claims the fund injection will see Lineo through until it reaches profitability. They may also be rehiring some of the recently laid off staff. <http://www.lineo.com/>



● Linux Labs are releasing an open source web browser – *Vagabond* – for PalmOS machines. Palm users have been without a decent browser since the demise of Digital Paths. <http://www.linuxlabs.com/vagabond.html>

Hoyt Duff

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



COMMENT

Abandoned software

“ If you are a *WordPerfect8* user on Linux you can no longer import files created with the *MS Word* format. The import filter (written by Filtrix) has a bug in it that causes it not to work when the date is a billion seconds after the Epoch (the One Billion Second bug). Corel decided it will no longer support *WP8* and the makers of Filtrix deferred to Corel.

Enter Valentijn Sessink and a *WP8* users newsgroup. Sessink created a patch that essentially lies to Filtrix about the date and gets it working. (<http://olivier.pk.wau.nl/~valentyn/wp8fix/>)

WP8 also has some other problems, like requiring *libc-5* and *ld.so-1.9.1* to run since it was compiled using links to older libs.

This episode teaches two things. First, an active and communicating user base is very powerful. Second, we will see an increasing problem with abandoned applications, often referred to as “abandonware”.

There is a site (<http://unmaintained.sourceforge.net/>) that address this subject for Free Software, but what about non-Free? I'm sure that a mechanism could be developed to make these abandoned programs remain useful to their loyal users. It would be nice to have *WP8* recompiled to work with modern libraries, and eliminate the bug in Filtrix.

But the bottom line is that with copyrighted software the owner can exercise their rights as they choose. That means that *WP* will die a slow death, kept on life support by few dedicated and technically capable users.

It does serve as an affirmation and validation of the use of Free Software, however. ”



Mailserver

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

Curphoo

As one of your avid readers and also one of the developers of the *curphoo* chat program for Yahoo!, I would just like to inform you that as of the 26th April 2002 Yahoo! have changed their login protocols. Therefore the version of *curphoo* that you have on your CD/DVD might not work.

I would like to inform your readers that they can get the latest version of *curphoo* from the website at www.waduck.com/~curphoo or from this website as of the 30th April 2002:

www.summerfield-technology.co.uk/curphoo
Charles Blackburn, Co-Developer
Curphoo

Typical. Thanks for getting in touch, and I hope the readers will respond by giving you plenty of feedback on the new version.

Upgrades again

Rather than debate the Linux development paradigm, I would like to pursue this topic from the perspective of a subscriber to *Linux Format*.

I find that I get very limited

value from the cover CDs because a large proportion of the software on them will not compile and install because of libraries, etc., not present on my system (SuSE 7.3 Home). I know that I can get these libraries, with a bit of searching, from the 'Net but one of the principal reasons that I subscribe to the magazine is to avoid this fuss.

I would get more value from *Linux Format* if, every now and again, a CD was devoted to libraries or that the required libraries were included on the CD as required.

Mark Myatt, via email

Yes, we appreciate that. Many common libraries are included in the coverdisc – check the essentials directory. Unfortunately, if we tried to include every library for every version of every Linux distro, there wouldn't be any room for any software. Most development packages and works in progress use the latest versions, but release version software often has less stringent requirements.

What's in a name?

A few years ago I installed a Linux system on my standalone machine

★ Letter of the month

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

Linux GTi?

I have noticed quite a number of dissenting words on forums, news groups and even your 'Mailserver' section, complaining about the difficulties faced by Linux 'consumers', and have been taken aback by the response these people get. True, often those making the noises of dissent want it all but do not want to contribute anything to gaining it, but then this is what a consumer does.

If Linux is to be taken into the mainstream, to be common place within homes, schools and businesses then it must cater for the consumer. The consumer does not want to know how things work, they do not want to know what to do when things go wrong, in fact they do not want things to go wrong at all. Take a nice shiny new car; I go out and buy one from my local dealer but I have little idea of how it works, I know how to drive it, what fuel to put in it and the basic maintenance procedures but

that's it. That is ALL I want to know. I have neither the time nor the patience to start tweaking this or fixing that. It is the same for computers.

In the real world the average computer use will not stand software failing to install, they would not be willing to search out dependencies, update and install different parts of their system just to get a piece of software to run. They simply want to be able to turn their computer on and double click on their icons, and things to just work.

It is here that Linux still fails. It is not just latest releases, but updates and 'stable' releases, and commercial software as well. Don't get me wrong, I love Linux, I only ever use Linux (Except at work where I have to use some Windows machines). However, if Linux is really going to move into the mainstream of desktop computing then it must pander to the needs of the consumer. It's OK to say that

it's an "...entirely different model of software development..." but the consumer won't stand for it.

Regards, Chris White, via email

Thanks for your thoughts on this. I think your car analogy falls over though when you say the main problem people have with Linux is installing software. A car doesn't get upgraded, normally. If consumers stuck to the software provided in their distro then there would be no problem. If they want to update their software, then they can use simple tools like *Mandrake Update*, *Apt*, or the myriad of other tools now common on the desktop. All commercial software for Linux I can think of has some sort of installer, and most of them work just as well as their Windows counterparts. Trying to compile latest releases from source requires a bit more knowledge, the same way that you wouldn't try and install a new axle on your car if you didn't have the faintest clue what you were doing. Also, before you bought a

new car, you might learn how to drive before you took off up the motorway.

You may say that 'consumers' don't want to do any of this. Fine. Stick to the major distros, and only install software that comes with an installer or from the vendor updates.

I don't think that software installation is really much of an argument against Linux on Desktops anyway. Why? Because 90% of desktops are used at work, where the user has no control over the software, and won't be installing things anyway.

But to show we appreciate your efforts, please accept the letter of the month prize, a copy of *Mandrake Gaming Edition* including the *Sims* (which should give you no installation problems).





alongside Windows. I work from home so the machine is both used for work and home uses. Since then I have followed Linux closely. In November, 2001 I built myself a new machine and prepared to install my software.

First on was Windows 98 – it has to be given the nature of the work I do and the software I need. Having established that base I then prepared to install Linux. After some thought, however, I began to question why I would install Linux. All the advantages and disadvantages have been discussed fully in your magazine, but the bottom line is at present I need Windows but I do not need Linux.

In issue 26 of *LXF* Michael Roberts, CEO of Lindows, affirmed the desire of Lindows to focus on building the product but stressed the need to stand up to the bully in the playground. As far as I am concerned I would rather Lindows forget the legal battle. Even if the Lindows product were renamed “Pigs-might-fly” my guess is I, and

“Even if Lindows were renamed ‘Pigs-might-fly,’ I and many others would buy it if it enabled us to have the benefits of Linux with the software of Windows”

many others, would buy it if it enabled us to have and use the benefits of Linux together with the software base of Windows. Ultimately, that would that remove the bully from the playground – or at least put him in his place.

Keith Hulse, via email

The Lindows case is interesting, as in refusing application for an injunction, the judge determined that it could be argued that ‘windows’ was a common term used to describe GUI driven operating systems before Microsoft released their first version of Windows. So, it’s been rather an own goal for Microsoft.

However, although the name might not be as important as the content, it is sometime worthwhile striving to keep generic names generic. Personal Computer is really a generic term, and can be applied equally as well to a ZX81, Mac, or whatever, but ‘PC’



LXF26 raised many issues – and the Lindows CEO drew comments.

seems to have been hijacked into meaning an x86/Windows computer.

Open offer

I’ve just managed to get a copy of the April 2002 edition of *LXF* and noticed that you were talking about *StarOffice* and *OpenOffice.org*. I’m the UK contact person at the *OpenOffice.org* marketing project

and wanted to offer my help to both *LXF* and your readers in either answering their questions about *OpenOffice.org* or passing them on to the relevant member of the community.

My personal opinion (not necessarily that of the entire project) is that Sun is right to charge a licence fee for *StarOffice* 6. It enables them to offer significant value additions such as support and additional propriety filters. This is in fact the Free Software business model that we’re all supposed to be using, offering consulting and other services on top of an Open Sourced codebase. AFAIK Sun has always charged a licence fee to corporate users of *StarOffice*, although significant discounts are available for educational institutions and those purchasing in bulk

The LGPL/SSSL licensed *OpenOffice.org* suite is totally compatible with *StarOffice* and any other suites that will be built on top of its Open Sourced code and is always available for those who can’t afford the licence fee.

A quick correction to David Petticrew who wrote in your last letters section that there isn’t a UK dictionary for the *OpenOffice.org* spellchecker. There is. It’s available to download from the *OpenOffice.org* site at http://whiteboard.openoffice.org/lingucomponent/download_dictionary.html. For various licensing reasons we can’t bundle them in the installer. Members of the community are working on a separate installer for all of this but it’s really only editing a few config files, something that most Linux users should be fairly happy with.

Also the styles dialog box can be turned off when loading (or at least should be able to be), I would encourage him to either file an Issue in our bugtracking program *IssueZilla*, ask on a mailing list or send me an email and I’ll do what I can to help. A final point – if you could, refer to us as *OpenOffice.org* rather than *OpenOffice* in the future please. As is explained in this FAQ www.openoffice.org/FAQs/main_faq_new_p7.html#51 we don’t own the *OpenOffice* trademark.

Please feel free to contact me or any member of the *OpenOffice.org* project if you have any questions, we’re here to help! Nick Richards, *OpenOffice.org* nick@nedrichards.com

Thanks for clearing that up. We hope to include a wider comparison of the various Office software suites in a future issue, where hopefully we’ll be able to include a comprehensive table of what features are available and which are not.

Codebreaking

In the March 2002 issue Mr John Briggs tried to dispute the statement that “breaking the Enigma code was the first use of computers in cryptography.”

Actually he is both right and wrong at the same time. What he failed to notice is that there were several Enigma machines (or cipher systems). It is true that Polish national Marian Rejewski broke the Enigma code before WWII, but that Enigma machine was not the same that was broken by the brilliant minds of Bletchley Park workgroup. And Rejewski did not do cryptanalysis entirely by hand. He actually devised mechanised “computers” called *bombes* that were used to test possible settings of Enigma machine. But when Germans increased the number of cipher wheels Rejewski’s method had reached its limit. But Rejewski’s methods were later used and developed by Bletchley Park cryptanalysts as Enigma machine was updated several times during the War (also, German Navy had a different version than German Army and it was even harder to break).

In response to this, Alan Turing studied the previously decyphered messages to find usable patterns. Later he had constructed a set of electrically linked scramblers that



OpenOffice.org – the new name for OpenOffice (not a big change).

eased the decyphering process, but the process still needed human intuition as the test keys had to be guessed (by deducing some passphrase, for example). Although it had shortcomings (as it did not decypher messages without guessed keys) it was a major breakthrough.

So, in a sense, these Turing “computers” did not actually break the code, but were merely an important tool to aid the human mind. Army Enigma was fairly easy to break as their system was simpler (and Army operators used easily guessable phrases) but in order to break the German Naval Enigma code they had to steal the codebooks from German Navy. So, in a way, the original statement is right, if we think of Turing’s machines as computers, which they certainly were (although they bear little resemblance to a modern computer). Their “programming” was crude and very limited, and they were built only for one purpose.

Even if they had no GUIs,

modern supercomputers are built only for calculations).

We could also dispute whether the computer has to be electronic device – I think the power source or computing parts has no relevance in this matter. Modern computers are just an example of miniaturisation with electronic devices, but we could build these devices in some other way, and still it would be a computer.

Mika Laaksonen, Turku, Finland

I knew there would be a reader somewhere who knew even more about this subject. Thank you very much indeed for your detailed response. At least that’s one ‘enigma’ that’s been solved.

USB answers

Lately I have noticed that you seem to be avoiding some of the questions that are being put forward from troubled readers.

My way of thinking, is that you are there by the grace of your readers and questions posed are not there just to annoy you. Most of these are genuine sticking points



Our USB answers in LXF26 plugged into the right spot.

get to grips with Linux, you are going to go the extra mile to help us the readers.

Often you tell us what should happen but you seem to forget what can go wrong, it’s not very often that you get ways of sorting the problem! Myself, I spent more than six months trying to get my LS120 working, only to find that it was simply a Directory needed in the main root file. And where in all the man pages was that simple piece of info? I don’t know, I never found it! I rest my case!

Please, a lot of people out there use USB now and a lot of people need help to get it working. I am one of them! Is there a program that tells you what is connected where? i.e. printer on /dev/ttyUSB0, etc.

R Faulkner, via email

Er, I’ve had a look at the issue you mentioned, and all the questions were answered as far as possible.

Sometimes an exact answer isn’t possible because exact information isn’t given, but all the answers seem pretty specific and accurate as far as I can see. Are you saying you can’t

make sense of the answers?

Versions 7, 7.1 and possibly 7.2 do seem to have problems with USB in general, mainly due to the kernel. As the answers state, usually these are solved by installing a 2.4.10 kernel or better.

The short answer to your question is that it depends on your distribution, which you have neglected to tell us, though these days most printers are mapped to /dev/usb/lp0, etc. There is no quick way of knowing where any USB device is connected in terms of the device file. ttyUSB is used for serial converters only, /usb/hiddev for HID devices (joysticks, etc), /dev/usb/scanner0 for scanners. A list of the minor numbers and associated device files is given at www.linux-usb.org

And to prove it...

Thanks for publishing my letter in LXF26. Following your advice I have switched to 2.4.10 and everything now works fine (including USB). I managed to get sound working by installing and compiling ALSA against the new kernel. This resulted in the creation of the

“We could dispute whether a computer has to be an electronic device these devices could be built some other way and would still be computers”

multimedia or gaming capabilities they were machines that could compute much faster than human mind. We should remember that most computers before 70’s were built only for calculations and simple data-processing. Modern generic computers are just an advanced form (and many

for a lot of people, LXF26 has a lot of requests for help on USB but none of your answers make any sense, nor are they of any use to those who are trying to get to grips with Linux! Your mag is looked upon by many to be Britain’s leading authority on Linux and if you want people to try harder to

Helpdex

BY SHANE COLLINGE

shane_collinge@yahoo.com





/lib/modules/2.4.10/misc directory and lots of snd*.o modules which did the trick.

I ditched my old Winmodem in disgust some time ago and now use a Multitech MT5634ZPXI 56K Voice Internal PCI controller-based modem which works very well. Getting the modem to work under 2.4.10 was a simple matter of deleting /dev/modem (which used to point to /dev/ttyS2) and replacing it with a new link to point to /dev/ttyS4. I found this by using `lspci -vv` to determine which interrupt my modem was using and then `setserial -a /dev/ttySx` to find which ttyS device was connected that interrupt.

I use *Vmware Express* to run some old Windows apps that I still need. In order to get *Vmware Express* running under 2.4.10 it is necessary to download and install the following patch: <ftp://platan.vc.cvut.cz/pub/vmware/vmware-ws-1142-for-2.4.7.tar.gz>

A big thanks to SuSE for making this information available in their support database too: http://sdb.suse.de/en/sdb/html/fhassel_express_73.html

I noticed at letter entitled "Yet more USB" from Richard Russell in LXF26. This letter struck a chord with me because I have seen exactly the same error messages on my system running both SuSE 7.1 and 7.2. I upgraded to 7.2 because I hoped it would fix the problem but it didn't. I found that USB was functional using the standard single processor SuSE 7.2 kernel but then what's the point of having a dual processor setup if you can't use both processors! My advice to Richard would be to install one of the standard Linux kernels (2.4.10 worked for me).

SuSE documentation is excellent and you will find plenty of help on how to go about installing a new kernel in the various handbooks supplied.

Steve Roper, *via email*

It was good of you to get in touch and update us on your progress. I hope other readers will benefit from the advice you have given.

Cracking tools

Just reading the April issue of LXF and felt I had to write with regard to the letter entitled "Cracker snacks". I just wanted to say that I agreed completely with your response.

As many good security books have said, "Security through Obscurity" is never a good idea. It is better to have this kind of thing on the CD so that administrators have access to tools that they otherwise might not look for, whereas crackers on the other hand will obviously find these tools elsewhere anyway.

I therefore don't believe you have made a "mistake" at all by including these tools on the CD. Keep it up.

On another note, with regard to the hot-topic "Are Sun right to charge for Linux versions", I would like to say that I think it depends a small amount on how much they charge, but otherwise I think it is a good thing.

I once worked alongside an 'IT Consultant' who was always playing down the importance of Open Source,

etc., and cited one of the reasons as being that because you didn't pay money for it, you can't hold anyone responsible. He said that for example with Microsoft, you can always ultimately go back to Microsoft for support issues. Personally, I always thought that this argument was weak. However, I am sure he isn't alone in his thinking.

I therefore believe that if people like this can see someone like Sun offering a product which you have to pay for, then they will get a nice warm feeling and it can only be a good thing all round. Well, it's just my opinion anyway and you did ask!

Darren Parkinson, *Bedfordshire*

Thanks for your vote of confidence. We don't wish to encourage cracking at all – we spend a great deal of time trying to keep such types out of our webserver, which seems to be a popular target! But we find tools like the ones mentioned help secure that server, and I'm sure others do too.

The question of support is a very valid one. We get loads of questions every month about how to print from *StarOffice*, or how to add new fonts, etc. If Sun charge for

the product, they not only owe it to their customers to help with these things, but will have the funds to do so properly.

Hoyt mail

Hoyt Duff's column in April issue (LXF26) says: "Mail clients that can't handle HTML are simply broken".

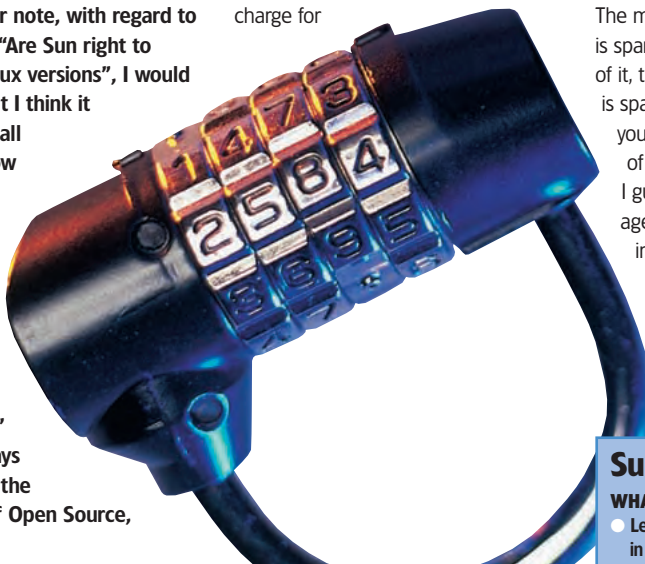
I can't agree with this. I use *Agent*, which doesn't render HTML mail, and I wouldn't have it any other way. HTML mail is conveniently shown as a blank mail, with an HTML attachment, making it easy to recognise and delete.

Most HTML mail which I receive is spam and I don't like the idea of the renderer following the **IMG** tags (which often have identifying strings attached to the URL via a **?**, thereby telling the spammer that I have read the mail).

If I receive HTML mail from a known or wanted source, I reply asking them to use plain text as well as or instead of the HTML. If they are incapable or unwilling to do that, well there's always the 'phone.

Jim Hatfield, *via email*

The majority of the HTML email I get is spam also. Actually, come to think of it, the majority of all the mail I get is spam (the disadvantage of having your email address printed in tens of thousands of mags every month I guess). Your point about the mail agent following links which can indicate you have read the message is a good one, and I guess we should ensure that where an HTML client is used, you can stop it accessing such URLs. **LXF**



Submission advice

WHAT WE WANT:

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

WHAT WE DON'T WANT:

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

WRITE TO US AT:

Linux Format, Future Publishing, 30 Monmouth Street, Bath BA1 2BW or email: lxf.letters@futurenet.co.uk

Reviews

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

LXF verdict explained

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

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

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

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

Value for money: Does it have a competitive price?

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



10 The close to perfect product.



8-9 Good, but has a few niggles.



6-7 Does the job, but needs work.



5-4 Average.



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

The Top Stuff Award

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

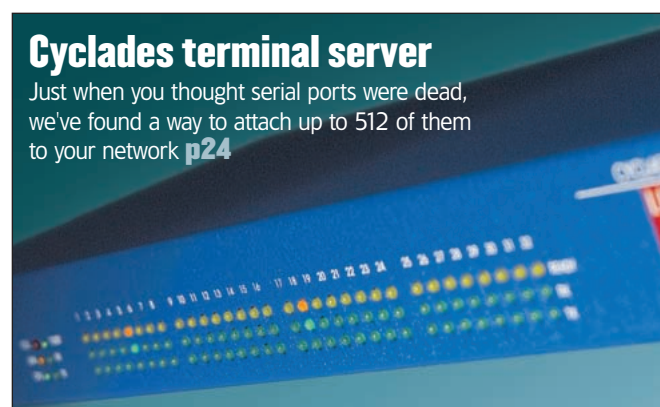


THIS MONTH...



Return to Castle Wolfenstein

The long awaited sequel to *Wolfenstein 3D* is on the shelves and runs on Linux. Find out why it's been worth the wait **p18**



Cyclades terminal server

Just when you thought serial ports were dead, we've found a way to attach up to 512 of them to your network **p24**

Volution

Manage hundreds of Linux boxes remotely and simultaneously — then work out what to do with all of your extra free time **p22**

Rapid Builder

Capture your X display and play it back as a simulation. **p26**

VMWare

The top PC virtualisation technology goes from strength to strength, with improved serial port emulation **p28**

Crossover Office

The Holy Grail of *Wine*-dom has arrived with successful running of *MS Office* without the need for any native parts of Win32 **p30**

Books

Javascript: The Definitive Guide, *Java RMI and Free as in Freedom* — a biography of Richard Stallman **p32**

COMING UP SOON...



SuSE 8.0

It's big, it's green and it arrived in the post this morning. Expect a review next issue

OpenOffice 1.0

Sun's open source MS Office-killer has just hit 1.0. We'll be putting it through its paces

Creatures

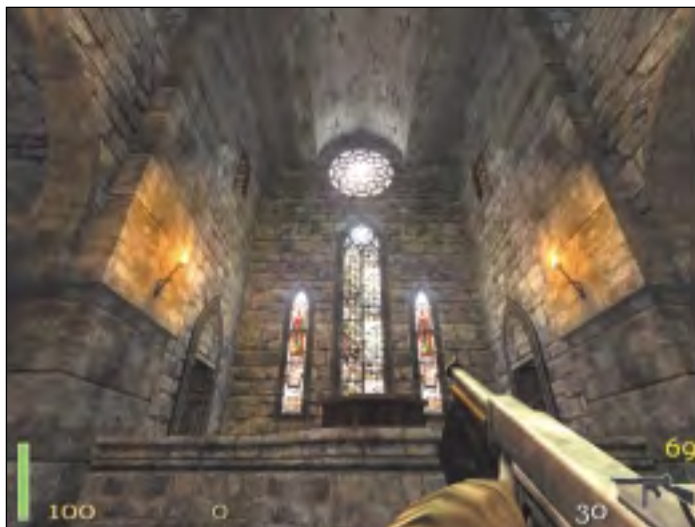
It's been downloadable for a while, but Linux Game Publishing now has the rights for an official version

Absoft ProFortran

A professional development package for number-crunchers everywhere

Rekall

Can the Kompany's user-friendly database rival Access? Find out if we ever get our hands on a copy



Full marks to the curate looking after the superb stained glass windows for not letting a little thing like evil occult ceremonies getting in the way.



Sneaking up behind the enemy and giving them a sharp dig between the shoulder blades can be the most effective way of dispatching them.

3D SHOOTER

Return to Castle Wolfenstein

Pining for the halcyon days of early 3D shootfests, **Paul Cavanagh** seeks solace, and finds it, as id delivers an updated classic.

A quality first person shooter that is available off the shelf to Linux users.

■ **DEVELOPER** id

■ **WEB** <http://zerowing.idsoftware.com/linux/>

■ **PRICE** £19.99

Nearly ten years ago *Wolfenstein 3D* introduced first-person shoot 'em ups to the mass market audience. Shooting Nazis, being attacked by German shepherd dogs and nudging up against walls and paintings in search of secret passages suddenly became a popular, and extremely addictive, leisure activity. Given the popularity of the game, it's surprising that there hasn't been a sequel until now. It could be argued that id's *Doom* and *Quake* games constituted sequels, but while these games developed the techniques employed in *Wolf* to new levels, they didn't involve taking a pop at Nazis.

Installation

The good news is that not only is *Return to Castle Wolfenstein* a worthy sequel, but that it runs admirably under Linux, and you can buy a copy off the shelf just about anywhere. Just

get yourself a regular Windows version of the game, download a few files from www.activision.com/games/wolfenstein/downloads/files.html and you're away.

Well, it's not quite that simple. There's no custom Linux installer, so you'll have to install it on your Windows partition and then copy the whole installation into your Linux

partition, or install using *Wine*, which is a slight pain, but fairly straightforward. Once installed, it's just a question of putting the files into the right place, and you'll never have



Oh dear, I don't think the sniper rifle's going to be a lot of help with an anti tank missile on the way.



There are occasional moments where you can sit back and take in the views – the cable car ride is one such moment.

to boot using Windows to play again. Marvellous.

Achtung!

It's certainly worth this minimal amount of effort because *Return to Castle Wolfenstein* is the best first person shooter to come along in ages. As Captain Blazkowicz, you begin the game imprisoned in the bowels of the

eponymous Castle, armed with a stolen pistol. A fellow agent has been killed during a brutal interrogation, and you'll have to develop some nifty shooting skills in order to escape. The evil of the Third Reich is presented more in the style of *Indiana Jones* than *Schindler's List*, with Nazis speaking in heavily accented English, reverting to their mother tongue for select words



Looks like a certain Nazi officer was enjoying a spot of hanky-panky before the party got broken up.

and phrases such as *schnell!*, *achtung!*, or *Jawohl, Herr Oberführer*. While they may have a certain *Boys' Own* comic book feel to them, they're still pretty deadly, and once they've spotted you they won't hesitate to send some hot lead hurtling towards you. Stealth is often the best policy – you'll stand a much better chance of survival if you can catch them unawares and get the

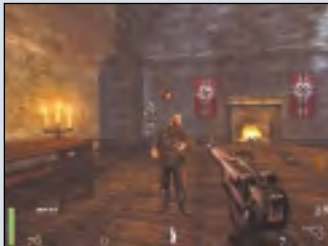
first shot in. The weaker soldiers will go down with a few well-placed shots, but don't expect any of the enemies to be easy pickings, if you miss they'll take cover, firing as they go, and once you've blown your position you'll attract the attention of every soldier in the vicinity.

There's no shortage of things to shoot at in this game, and all of the



Hun Hunting

During the course of the game you'll encounter Gerry in a range of guises. Here's a taster of Boche forces.



FOOTSOLDIERS

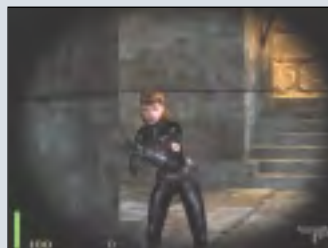
You'll have to do battle with military units who can be armed with sub-machineguns, Mauser rifles and, later on, heavier fire power. Officers like this guy normally have Lugers.

ZOMBIES

Whilst meddling with dark forces, the Nazis have awoken these shambling



hordes. They come in three varieties – skull slinging moaners, axe wielding nasties who deflect bullets with their shields, and fire breathing animated corpses.



ELITE BODYGUARD

Nimble and deadly, these leather-clad women warriors are reputed to belong to a witches' coven (although there are more than thirteen of them). Their silenced machineguns overheat quickly, so they attack then retreat fast, before coming back for more.

FLAMETHROWER OPERATORS

Much harder to kill than your average Nazi, these guys also have the obvious advantage of being able to incinerate you. On the plus side, you



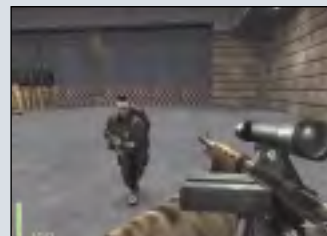
can swipe their kit once you've sent them packing to hell.

PARATROOPERS

These guys normally attack in a group of at least three, carry a fast firing automatic sniper rifle, and they don't miss very often.

LUMBERING BRUTES

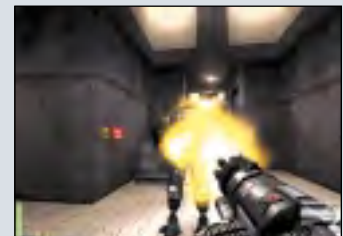
Created by mad evil scientists who



apparently decided that legs are dispensable assets, these ugly fellers walk on their knuckles and attack with massive electric shocks.

SUPERSOLDIERS (UBERSOLDAT)

Another evil creation of the Nazis, easily capable of taking you out in next to no time – they can carry chain guns, anti-tank missiles or a powerful electrical weapon.



LinuxFormatReviewsWolfenstein



Using the binoculars, you can enjoy watching the Gerries scratching their privates or sneaking a crafty fag.

« soldiers and beasts look fantastic, are animated convincingly, and there's a good variety of enemy types to keep things interesting. Those all-important little details are there too – you can gain advantage by attacking while soldiers are reloading, overhear soldiers complaining about the weather, and read sinister notes from the SS High Command which contain dark threats to keep the lower ranks in order.

Take that, schwein!

Given the stiff resistance put up by the Nazi menace, it's a good job that you can salvage weapons from your victims, or purloin their armouries. New weapons come along fairly regularly, which helps to keep things interesting, and all of the weapons are fun to use, and have distinct advantages in certain situations. The sub-machinegun has a rapid rate of fire and packs a punch, but is noisy. There's a silenced version, which is great for stealth, but overheats rapidly. Grenades can be used tactically for great advantage, bung 'em down a hole into an enclosed space and say goodbye to anybody who might have been waiting to shoot you in the back as you climbed down a ladder. Grenades can be kicked back to you, so you can time your throw so that they explode upon landing. I found myself sniping from a cosy hidey-hole a great deal, which can throw the enemy into a panic. This can have its disadvantages too, because while looking through a sniper scope you are unable to see what might be happening in your peripheral vision, so

the enemy can creep up on you unawares. There's also a chain gun, dual colt pistols, and a BFG style energy weapon capable of electrocuting a whole huddle of Hun in one fell swoop. All of the weapons have distinctive sounds, so you'll know when you're being sniped at by a Mauser, because you'll hear a muffled crump that'll echo off walls. The sound effects are among some of the best I've heard, with convincing ricochet effects, satisfying explosions, and the sickening sound of bullets impacting into Blazkowicz's flesh.

Less is more

One of the first things I noticed while playing is that there's a certain back to basics feel about the game, which is odd because it has all of the modern features you'd expect – yes, you can look up and down, there's admirable enemy AI, and there's swimming, jumping and climbing involved. None of this was present in the original *Wolf 3D*, and these features have added much to the genre over the years. What's good about this game is that it has, by modern standards, really small levels. Beautifully designed, detailed, challenging, fun to play small levels. This is a good thing because so many games these days have dirty great huge playing areas that take ages to complete. This might sound like you're getting your money's worth, but all too often you can loose interest in such games. With *Return to Castle Wolfenstein* you can progress through the game fairly quickly, finding new challenges, weapons, enemies and plots at regular intervals, which makes



That was close! This fellow is on the run, but he's not turning his back, and he's not going to stop shooting for anything.

for compulsive playing. The levels included in the game vary from the labyrinthine tunnels below Castle Wolfenstein itself, to streets recently bombed by the RAF, or secret weapons facilities and airbases. There are outdoor sections in lush forests and barren arctic landscapes. This variety is a welcome change from the sewers and train stations that appear all too often in first person shooters.

Secret service

Another feature that harks back to the glory days of Doom is being told at the end of each level how many secret areas you've found, and it's almost impossible not to go back to search if you haven't found them all. Some of the secret areas are fairly obvious – hidden behind pictures, or using a loose book to open up a bookcase. Sometimes you'll have to pick up chairs and use them to access hard to find areas, sometimes scanning with the binoculars will help you locate areas, and some are so cunningly hidden that it'll take ages for you to find them. These little challenges add a few hours onto the overall time that you'll spend playing the game. Some missions have specific objectives that you'll have to complete before you can finish the level – destroying a target, assassinating high-ranking officers, or finding artefacts. These levels give you even more scope for thorough exploration, while others demand a certain level of imagination. One such scenario involves gaining entry to a compound without any guards setting off alarms, you can choose whether to eliminate the guards

before they hit the alarm, or try to avoid them altogether.

And there's more

RtCW offers plenty of action that'll keep you interested from start to finish, and leave plenty of room for replay to boot. And that's just the single player missions. There's an online multiplayer game included, where you can choose to play as a soldier, a medic, an engineer or a lieutenant. In this version, only medics can dispense medikits, lieutenants dole out ammo, and engineers deal with blowing things up. There are three game types to play: Objective, where teams must try to achieve, erm, objectives; Stopwatch, the same but up against the clock; and Checkpoint where teams must try to control a number of flagged areas. As with all online fps games you'll need to practice for a while before you can survive long enough to get your bearings. If you've been playing this on Windows, stick it on your Linux partition now. If you haven't got *RtCW* yet, or for that matter any other FPS, you really should give this a go. **LXF**

LINUX Format VERDICT

Presentation	10/10
Gameplay	10/10
Value for money	9/10

A superbly presented Nazi hunt, which stands head and shoulders above the competition in terms of level design. You'll keep coming back for more.

LINUX Format RATING
 **10/10**

WORKSTATION MANAGEMENT SOFTWARE

Caldera Volution Manager 1.1

David Cartwright simplifies his life with Caldera's management software for diverse Linux networks.

Web-based remote management software for *nix boxes. Compare Trustix XPlay.

■ **DEVELOPER** Caldera

■ **WEB** <http://www.caldera.com/products/volutionmanager/>

■ **PRICE** US list price \$2,995 (server + 10 clients)

Volution Manager (VM) is a Linux-based system management and admin package for Linux networks. It addresses the key problems facing anyone who manages a lot of Linux machines – watching key activity from afar, discovering problems before they stop the user working, and remotely installing patches and new apps.

Although it is fairly straightforward for those of us with one or two PCs at home to keep up with service packs, new software revisions, what machine has what IP address, and so on, it's not so easy for people who look after large collections of computers. Hence the existence of system management software – packages that allow an IT manager to keep tabs on a large number of machines at once, and to do software installations remotely on many machines at once.

The VM server runs, unsurprisingly for a Caldera product, on Caldera OpenLinux (and a CD containing OpenLinux 3.1 was included in the package alongside the VM CD). It can, however, communicate with, and manage, a variety of Linuxes on the remote machines it looks after, including Caldera (2.3-3.1.1), Mandrake (6.1-8.1), Red Hat (5.2-7.2), SuSE (6.1-7.3) and Turbo Linux (4.0.5-7.0) – not to mention UnixWare 7.1.1 (we think there's still someone out there using that) and SCO OpenServer 5.0.x.

Installation

The first step in this review of VM was to install OpenLinux – a dead simple task as the installer's really simple, and it spotted that our machine was already set up with root and swap partitions that we were happy to overwrite with the new OS. All the hardware was detected without hiccup, so it was just a case of waiting twenty minutes for the files to copy (and Caldera's installer is quite cool, because while it's asking you questions about your mouse, monitor and network, it's already copying files in the background).

Once OpenLinux was up and running, installing VM was just as easy

```

Terminal
File Edit Settings Help
Client Config. Information ==> (/etc/opt/volution/volutiond.conf)
LDAP Server : LDAP://maart.korana.com:636
Authentication Name : cn=dsc-korana-com,ou=computers,dc=korana,dc=com
LDAP Software Repository : ou=Software Repository,dc=korana,dc=com

Is Volution Client Running ==> YES - SUCCESS
Client's DCD Service ==> maart:601 - SUCCESS
Client's DENS Service ==> maart:602 - SUCCESS
Client's SLP Service ==> maart.korana.com - SUCCESS
Client's LDAP Service ==> LDAP://maart.korana.com:636 - SUCCESS
LDAP Distinguished Name (DN) ==> cn=dsc-korana-com,ou=computers,dc=korana,dc=com - SUCCESS
Connection to LDAP Server ==> cn=dsc-korana-com,ou=computers,dc=korana,dc=com - SUCCESS
Client's General Health ==> GOOD - SUCCESS
Is SNMP Daemon Running ==> YES - SUCCESS

=====
VOLUTION CLIENT DIAGNOSTIC ANALYSIS COMPLETE
=====

[root@dsc /]#
  
```

volding tells you whether your client workstation is misconfigured.

as setting up the OS. The installation guide goes through a few prerequisites before telling you how to install the VM package itself, but it turned out that all of what's needed is there by default with the 'Complete' installation anyway. The installation manual does a fair job of frightening the user with talk of which LDAP option to choose, and what LDAP is, but in fact LDAP is dead easy to comprehend and, unless you have a particularly large installation, it's fine to stick with the OpenLDAP option that's installed by default with OpenLinux. If you have a sizeable

installation you'll consider Novell's *NDS* (which ships on a separate CD in the VM box) or *iPlanet* as alternatives to *OpenLDAP*, but we didn't go there for this review.

Client packages

The installer program asks for a few key items, most of which are passwords but some of which let you customise the setup to your organisation. It's obvious what you have to change, though – typing your company name where it says 'Organisation' and so on. When you've

NetScape: VOLUTION <2>

CPU hog

Log notice when any process' utilization exceeds

Local event to signal when notice threshold crossed

Log warning when any process' utilization exceeds

Local event to signal when warning threshold crossed

Log alert when any process' utilization exceeds

Local event to signal when alert threshold crossed

Watching for processes 'hogging' the CPU of a workstation.

NetScape: VOLUTION <2>

Searching for Objects to Link to this object

Search expression

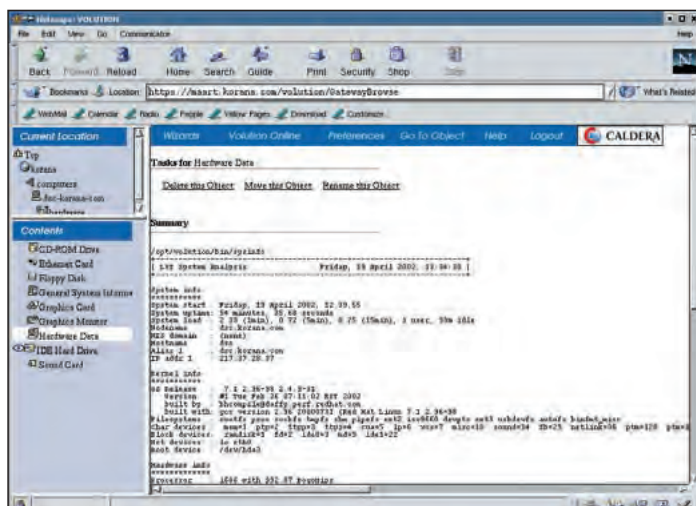
Start searching from here:

☐ Automatic wildcard search

☐ Select all

Object Name	Location	Object Type
dsc-korana-com	ou=computers,dc=korana,dc=com	calderacomputer
maart-korana-com	ou=computers,dc=korana,dc=com	calderacomputer

Linking computers to policies and profiles.



Complete system information for the remote workstation.

answered all the questions, a five-minute file copy is all you need to get going.

Once the server is installed, you need to install the client package on the remote workstations you're going to manage with VM. I made the mistake of running the 'install.sh' script by double-clicking it from the Red Hat GUI – the wrong thing to do because, although in my case it worked okay, it doesn't give you any clue as to whether it's doing anything! When run from a command line window, you get progress information, which is far more reassuring. Once installed, it starts up the VM client and some communicating-with-the-server daemons and you're off and running. Beware, though, that your networking setup has to be correct, or things will only half-work (I had a minor DNS misconfiguration which prevented a Red Hat client from communicating properly with the VM server). Fortunately, there's a *voldiag* command-line tool that runs through the key communication settings and tells you when something is broken.

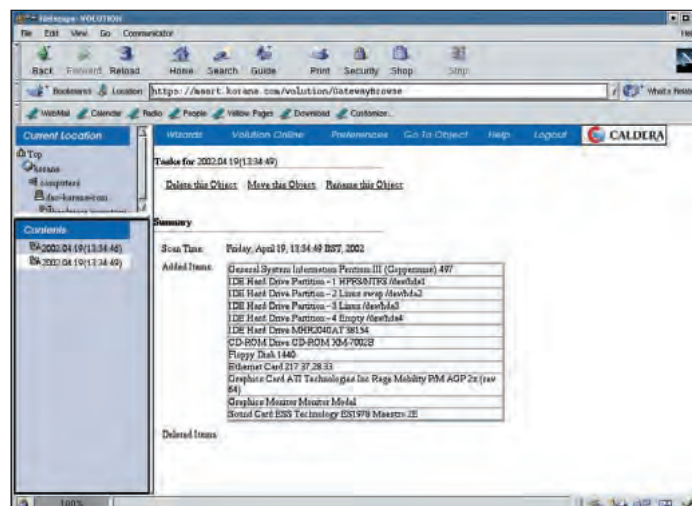
Policy management

Once you have your server and client(s), you can start doing stuff. The server user interface is a secure HTTP (<https://...>) web screen, which I found runs like treacle when you've first installed the package – presumably because the system is still building its back-end world. After a few minutes, though, it's like lightning. The front end is a typical three-pane screen – a little pane at the top left shows the tree of objects you can pick from, another little panel below it shows the things you can pick from in the item you're

currently viewing, and the large right-hand pane gives detailed information on the item you've chosen from the menus on the left. Within the top-level item representing your organisation, you have a number of things to choose from, most notably Computers, Actions, Policies and Profiles.

Computers, as they sound, are the remote machines running the VM client. Policies can be thought of as 'monitoring rules' – it's within the Policies section that you tell VM to (for instance) alert you if any machine's root disk gets more than 80% full. Profiles are software and file selections – so your 'Email user' profile might include the *Pine* mail client RPM and perhaps some config files customised to your organisation. Actions are fairly self-explanatory, though, – they define what policy checks and what profile installations to do on what computers, and when. Computers can be pooled together in Computer Groups and actions, policies and profiles attached to the group – the members of the group will inherit whatever behaviour you've defined for the group.

Policies come in a number of flavours. The two that will probably be used the most are hardware and software. The hardware policy lists what equipment each workstation has, and how it's configured – so it'll tell you the hardware address, IP address and subnet mask of the network card, for instance, or the current resolution of the video adaptor. The software policy, unsurprisingly, tells you what packages a workstation has installed and what version. Because the system takes regular peeks at the remote workstations (in fact it can be configured to do a new scan each time



The hardware we've added to the workstation since the last scan.

something changes on a computer) you can walk through a set of snapshots, where VM Manager tells you only what has changed since the last scan. There's an element of interaction with the VM home site too – a software inventory can show not just what versions of what packages are installed, but whether they're the most current versions available. You also configure health policies (CPU loading, disk usage, etc), gateway policies (network services, such as email or SNMP) and printer policies (printer settings). Each item in a policy can be configured at a number of levels – so you might flag a 'notice' if the CPU goes above 50%, a 'warning' over 70% and an 'alert' over 90%.

Interface issues

Profiles are more complicated than policies to config, but that's because you can do more than just copying the odd RPM over to the workstation: you can specify both RPMs and files to copy, define installation schedules, and even provide pre-install, post-install, pre-uninstall and post-uninstall scripts as well.

VM takes a little while to get to grips with, because the user interface isn't all that obvious in many cases. I guess this is due to it being Web-based rather than a native X-Windows application – the obvious way to do lots of things would be to let the user right-click and select from a popup menu, but this isn't doable in a web browser of course. Likewise of drag-and-drop: there isn't any, for the same reason. I'd also have liked to see less instances where you have to specify search strings (e.g., when allocating an action to a computer, you have to

specify a search pattern and then select from the resulting list) but again, there is some sense to the way it works – namely that if you have 5,000 computers the last thing you want is a monster of a pull-down list to scan for the item you want. Overall the user interface is usable, though one can't help wondering whether it might benefit the user to sacrifice platform-independence in favour of a non-web-based alternative.

Conclusion

Whether VM has a market remains to be seen. Linux isn't exactly the most common desktop application in the world, and this type of package lends itself most readily to fleets of desktops (let's face it, if you want to monitor a small bunch of servers, you use *Big Brother*, and if you want to keep a handful of Linux systems up to date you do it with Debian's *apt-get*, Red Hat's *up2date*, and so on). It can't be denied, though, that VM is a pretty good attempt to address the config and software management task that will face IT managers with loads of Linux desktops to manage ... if such an individual exists! **LXF**

LINUX Format VERDICT

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

Overkill for many networks, and some interface and config issues, but a real time-saver for the hassled admin.

LINUX Format RATING

7/10

TERMINAL SERVER

Cyclades TS2000



Richard Drummond tries to sort his DCEs from DTEs when investigating Cyclades Linux-based terminal server.

Competition includes rival devices from Vecmar and Lightwave, and from Keyboard-Video-Mouse switches.

- **MANUFACTURER** Cyclades
- **WEB** www.cyclades.com
- **TEL** 01724 277 179
- **PRICE** £2025

A terminal server is a box with multiple serial ports and a LAN port. Historically, these devices were used for connecting dumb terminals to a server via Ethernet. In today's bandwidth-obsessed world, where Ethernet has replaced RS-232 as the *de facto* networking medium, nobody uses dumb terminals any more – but the humble serial cable and the terminal server are still useful.

The principal application here is that of so-called, out-of-band network management. While Ethernet forms the main network for data in your local network – since RS-232 is a simpler, point-to-point technology, it can be used as a back-up for management when your LAN goes down. In fact,

many devices – from routers to UPSs – have serial ports for console access rather than supporting video or a keyboard. Making a device 'headless' in this way reduces cost, size and power-consumption.

The terminal server can thus be employed as a console access server – a server which provides console access to multiple devices via the local network, obviating the need to hook up a terminal to each device. Connect that LAN to the Internet and your administrators can have console access from anywhere in the world. Other rôles include acting as a remote access server. Lots of serial ports mean you can hook up multiple modems or ISDN adapters to allow dial-in access to your network.

The Cyclades TS range is a family of stand-alone terminal servers with models offering from 2 to 48 serial ports. These devices may also be clustered, with a single master server controlling up to 512 ports. The TS range run an embedded flavour of Linux (currently with a 2.2.14 kernel) and are powered by the PowerQUICC processor, an embedded system-on-a-

chip processor with dual processing units: a PowerPC core and a special-purpose Communications Controller. The model on test here is the TS2000, which features 32 serial ports. These ports use RJ45 connectors, and so the TS2000 manages to fit all this in a rack-mountable, 1U-height case. There's no standard for serial cabling with RJ45, however, so you'll either have to buy extra cables from Cyclades or construct them yourself (pin-outs are included in the manual).

The TS2000 has many applications and can function as a terminal server, console server, remote access server or a combination of all three. It's all a matter of configuration.

Serial set up

Before you can do anything with it the TS2000 needs some basic setting up. Initially this means connecting a terminal – or more likely a PC running a virtual terminal – to its console management port. When the device has booted up, you can log in and edit the necessary configuration files to set

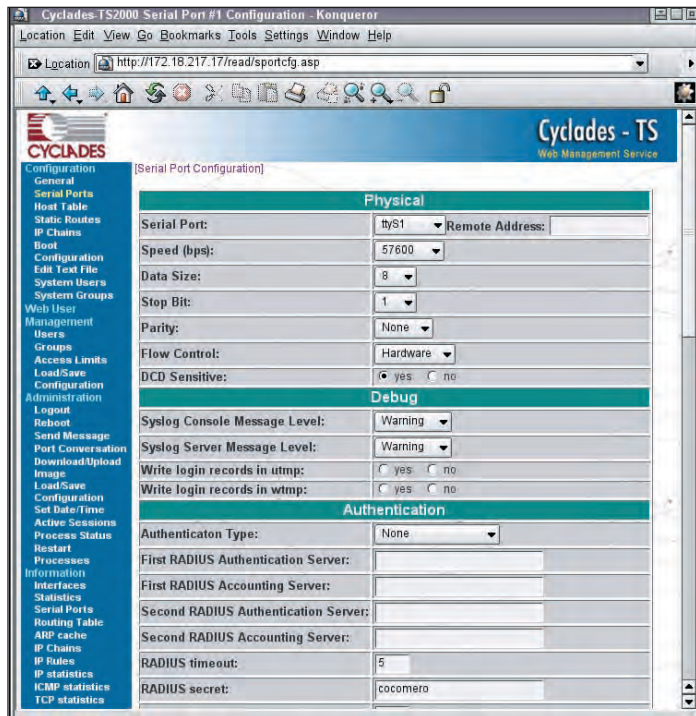
TS2000 Specification

CPU MPC955T
Memory: 64MB
Flash: 4MB
LAN: port 10/100BaseT Ethernet
Serial: 32x RS-232 DTE on RJ-45
1x RS-232 console port on RJ-45
Dimensions: 17 x 7 x 1.75 inches
Power: 40W max

up the LAN connection – either manually or by DHCP.

The TS2000 includes a minimal distribution of Linux with a subset of the usual commands and *Vi* for editing. (Alternatively you can use the file transfer facilities of terminal program such as *kermit*, and edit the files remotely.)

Once the Ethernet port is configured, the rest of the configuration may be performed by the TS2000's web interface, if you prefer. You should also now set the speed of the console port (the default 9600 baud is painfully slow), modify the start-up sequence to start any



The web based interface speeds up configuration, but how do you update the TS2000's firmware?

services you require and perform any other basic configuration.

Performing the initial configuration is not difficult, but it does require familiarity with Linux. The manual states what has to be done, but not in much detail. It would be much easier, if the TS2000 were initially configured to obtain its network settings via DHCP. Then, you could just plug it in and use the web interface straight away. This would be much less off-putting to beginners. Luckily the TS range come with free and unlimited technical support.

The basis for the TS2000's flexibility is a program called *portslave* which performs the task of accepting (and optionally authenticating) connections and routing between the serial ports and the network. Not surprisingly, this configurability comes at a price. It's a complex program to configure, and while using the web interface is often quicker, you still need to know what you're about. Several template configurations are supplied for the various server roles you may

wish the TS2000 to perform, which you can modify for your setup, but *portslave* itself is poorly documented and the manual supplied doesn't help much if you wish to deviate standard configurations.

The *portslave* config file is split into three sections. The first covers general options, such as the IP address to assign to the network connection; the second specifies global options to apply to all serial ports; and the third lets you tailor these options per serial port. So, you can specify, for instance, a default connection speed for all the serial ports, and, for those that that you wish to differ, you can configure individually.

The most important serial-port option is the protocol option which controls the behaviour of the server with regard to connections. To set up a port or ports for terminal serving you would specify *rlogin*, *telnet* or *ssh* here, the IP address of the server in the port's **host** parameter and disable authentication (the server would handle authentication); for a console

In a flash

Solid state Linux

Like many embedded, diskless devices the software driving the TS2000 is stored in flash RAM – thus allowing easy upgradability in the field. Installing a new kernel image, for instance, is child's play: simply copy it to the file `/proc/flash/zImage`.

Also upgradable and stored in the same directory is the RAM disk image which the TS2000 boots from. Optionally, the TS2000 can use BOOTP at start up and retrieve the kernel and RAM disk image from a remote server using TFTP.

access server, you would use **socket_server** or **socket_ssh** as the protocol and specify the IP address and socket number to be assigned to the serial port; for remote access, you would use **ppp** as the protocol and specify the address to assign to the dial-in client.

Thanks to *portslave*, the TS2000 is flexible with regard to authentication, also. You can either disable configuration entirely (in which case the TS2000 performs no authentication itself), or tell it to use the local database and/or a remote RADIUS server for authentication. For PPP dial-ins, *portslave* offers either a chat script or PAP authentication.

Besides *portslave*, the TS2000 runs various other services, such as SSH, NNTP, and SNMP. These all have to be configured manually, however, and are not covered by the web interface. The TS2000 also includes a kernel-based *ipchains* firewall – which, again, needs to be configured manually, although this time it is support by the web interface.

Once the device has been set up appropriately and tested, you need to save those settings. The TS2000's various configuration files are stored in flash memory, and a script called *saveconf* is used to make any changes persistent (a 'Save configuration' option is also provided on the web interface).

Flexible solutions

The Cyclades TS2000 is a very flexible piece of kit. The openness and configurability of Linux adds an extra

dimension to the device, beyond being a plain terminal server. However, the TS2000 is not particularly easy to use. The 100-page manual included is helpful, but just doesn't provide enough detail. The web interface to the device makes configuration quicker in many cases, but provides no on-line help, so is not appreciably easier to use than manual editing of the relevant config files.

The price-tag of £2000 seems steep if you merely consider the components in the box. However, this is a specialist product, and, when you factor in the free support, it seems like not such a bad deal. If you then compare it against the competition, then the TS2000 – being more versatile and smaller – seems positively good value.

Overall, the TS2000 is good solution for all manner of terminal serving tasks and is particularly suited for providing out-of-band network access to racks of headless servers. However, Cyclades have a lot of work to do to make the device user-friendly. **LXF**

LINUX Format VERDICT

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

A low-cost, versatile terminal server – which suffers poor useability and a lack of documentation.

LINUX Format RATING

8/10



Don't be fooled – that bank of 32 RJ45 sockets is for serial cables, *not* ethernet.

SIMULATION AUTHORIZING

RapidBuilder 1.0

Richard Drummond finds out what a simulation authoring tool is and whether this one's any good.

Tool for capturing, annotating and editing X sessions. No direct competition on Linux.

■ **PUBLISHER** XStream Software

■ **WEB** <http://www.xstreamsoftware.com>

■ **UK DISTRIBUTOR** ForLinux
(Web <http://www.forlinux.co.uk/>)

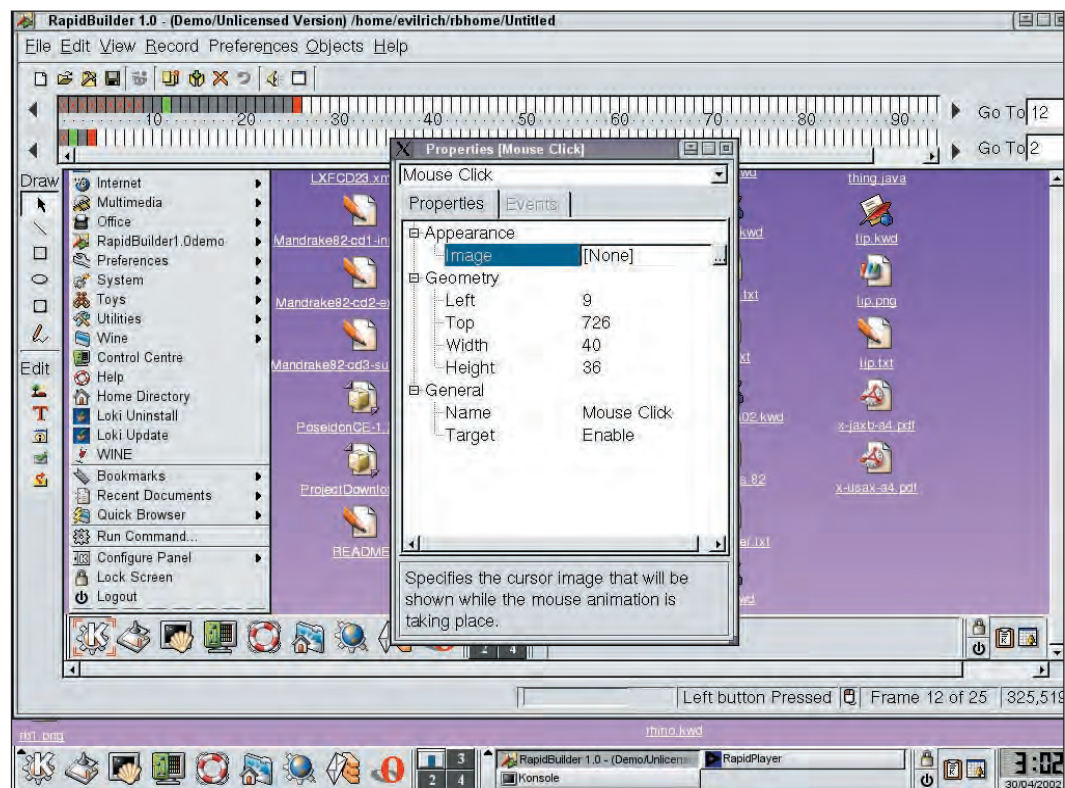
■ **PRICE** £845.99

You would be forgiven for thinking that *RapidBuilder* is yet another rapid development suite in the mould of *Kylix* and *OmnisStudio*, but you'd be wrong. It is in fact a "simulation authoring tool". "A what?" I hear you ask. Let me explain.

RapidBuilder lets you capture any input and output to and from your X display – including all screen changes, mouse clicks, and key strokes – and assemble it, frame-by-frame, into a simulation. This simulation can then be played back to recreate the captured X session. Sessions can be captured in two basic modes: with and without user validation. When capturing with validation, all mouse clicks and key strokes are recorded, and the user will have to repeat these actions when the session is played back. For example, if, when capturing the session, the author clicks the *KPanel*'s start menu and then the *Konqueror* entry to launch *Konqueror*, then the user playing the simulation will have to do likewise. Capturing without validation just captures the graphical state of the display and essentially creates a non-interactive movie. In addition to capturing frames, *RapidBuilder* also provides tools to manipulate frames and drawing tools to enhance and annotate them – and thus transform your captured session into a useful and slick presentation.

Roll the camera

So, what's it for? Like a traditional development environment, *RapidBuilder*'s usefulness is limited only by your imagination. Possible applications include creating interactive tutorials and walk-throughs, or



RapidBuilder provides a video-editing-like interface for tweaking and enhancing your captured X session.

demonstrations of a software product.

The *RapidBuilder* suite includes the *RapidBuilder* authoring tool itself and *RapidPlayer*, a standalone simulation player implemented in Java. Installation of both components is painless with the graphical installer included. *RapidBuilder* is based on the Qt2 toolkit, and a version statically-linked with Qt is included – so there's no need to worry about dependencies.

The *RapidBuilder* interface is basically similar to a video-editing app. The main area of the display shows the current frame, and you can step through the frames in your project with the navigation bar at the top. This features two controls: the top one represents all the frames in your project, while the lower one represents frames with a captured user action. The navigation bar lets you perform basic operations on frames: here you can select, cut, copy, paste and delete frames as well as enabling or disabling actions on selected frames. The main toolbar also has controls to let you

append and insert empty frames.

Before starting to record an X session, you need to set your capture preferences. Here you can choose whether to capture frames with validation or not. You can also select the sample rate in frames per second – or, if capturing with validation, choose to capture frames only when a user action occurs. The higher the frame rate (the maximum is 10 fps), the more memory you need. The stream of frames is compressed as it is captured, but, for example, the manual states that 64 MB is required to capture at 2 fps, 128 MB for 4 fps.

When you hit the record button to begin capturing, the *RapidBuilder* window is hidden and capturing commences. Everything you do now (depending on the preferences you've set) will be recorded. You can stop and pause/resume capturing with the two buttons that *RapidBuilder* places in the bottom right of the screen.

All-in-all this system works well. You need to sit down and plan your

session before you capture, though, because – while you can perform a certain amount of editing after the fact – it is much easier to get it right in the first place. My one complaint is that *RapidBuilder* doesn't take advantage of X's network architecture: it captures from the display that it itself is running on. Now, if your desktop looks anything like mine, it's going to be very messy – not something you wanted included in your captured session. *RapidBuilder* should really be able to capture from a remote X display – then you could dedicate one machine for the development environment and one for the capture environment, with each tailored appropriately.

The editing room

Once you have a captured an X session, you can use *RapidBuilder*'s tools to improve and enhance it. A shortcoming here, however, is that the *RapidBuilder* environment itself provides no function for playing back a simulation; it can only be done with

the external *RapidPlayer* program. This really slows down the edit/cycle. You must perform your edits, make the simulation as an RBX file, and save it out (*RapidBuilder* then, annoyingly, closes the project), play it in *RapidPlayer* to see the effects of your edits and then reload the project in *RapidBuilder* to carry on editing.

The first step to enhancing your capture will probably be to tweak the validations. A captured mouse click is represented in the frame display as a rectangle marking the bounding-box of the click. You'll want to set the size and position of the bounding box to reflect the object on screen being activated, and you can do this directly by dragging with the mouse.

Next, your captured X session is probably not particularly self-explanatory, a problem which can be rectified with *RapidBuilder's* annotation tools. This includes a simple set of drawing tools – line, rectangle, and circle with various line and fill styles – and some simple widgets – an image object (which can import JPEG, GIF and BMP files), various text fields, and a hyperlink object. The hyperlink is perhaps the most useful, since it can perform an author-defined action when clicked by the user during playback. Thus you can set up a link to jump to a specified frame, stop the simulation, open a URL in a browser, play a video file (AVI), launch an external program or even start playing another RBX file. The properties of all these various objects in a frame (including the frame itself) can be edited by popping open the properties editor with a right click on the object. This dialog lets you manipulate an object's attributes – to set colour, position and so on, where appropriate.

RapidBuilder also makes good use of audio clips. You may use an MP3 or

WAV file as a soundtrack or assign sound clips to actions and event. This is a great way to provide the user with feedback for validations. Other less practical features include a selection of wipe and fade effects which may be applied to frame transitions.

Lagging behind?

RapidBuilder is an interesting tool. It lets anybody without any programming experience create interactive presentations. It is easy just to jump in and start using it – and the online manual can solve any problems when they occur. *RapidBuilder* is a unique and specialist product – without any real competition on Linux – and this perhaps helps to explain the high price tag. However, while XStream have got the basic functionality of *RapidBuilder* working well, it is crying out for extra features. Some of these I've already mentioned, but other omissions include a function to export a non-interactive simulation in a useful format (say, AVI or MPEG), the ability to capture more than one X session per simulation and a function to insert frames from an external RBX file. Perhaps the worst sin is that the Linux version of *RapidBuilder* is lagging way behind the Windows version – the Windows release is at version 3.0. [LXF](#)

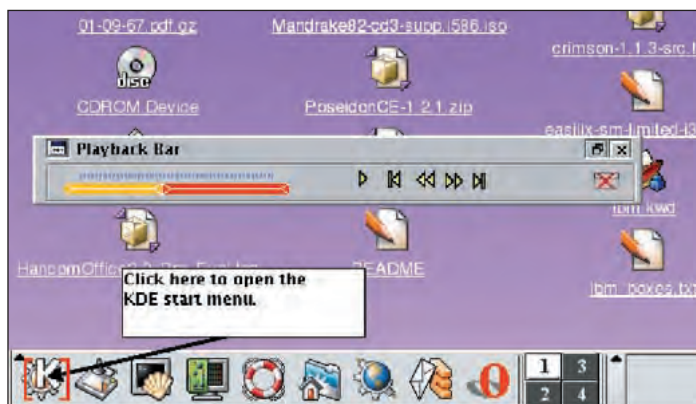
LINUX Format VERDICT

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

A unique and easy-to-use application – but it needs a lot more development to realise its potential.

LINUX Format RATING

7/10



The ability to annotate your simulations is useful for teaching purposes.

ACCELERATE YOUR SYSTEM



NEW! WITH KDE 3.0!

8 REASONS FOR USING SuSE LINUX:

- 8.0 KDE 3.0 - the easiest desktop interface
- 8.0 Extensive and automatic hardware detection
- 8.0 Multi-media applications - digital sound and video (ALSA 0.9)
- 8.0 StarOffice - the MS-compatible office system
- 8.0 Personal firewall security for your system
- 8.0 YaST2 - simple system and network configuration
- 8.0 SuSE YOU - instant online update facility
- 8.0 Stability, Reliability - Open Standards!

Please invoice us at the following address:

Company

Surname/Name

Street

City

Post-Code

E-mail

Phone

☐ Personal 8.0
£ 39.00*

☐ Professional 8.0
£ 59.00*

☐ Update 8.0
£ 39.00*
(only for existing users)

* shipping charges £5.00

Order Online Today! www.suse.co.uk

@ info@suse.co.uk

+44 (0) 20 8387 4088

+44 (0) 20 8387 4010

SuSE Linux Ltd.

The Kinetic Centre • Theobald St.
Borehamwood • Herts. WD6 4PJ



PC VIRTUALISATION

VMware Workstation 3.1

Richard Drummond asks what's new in the latest release of this popular virtualisation system.

Rivals for Windows compatibility include *Win4Lin* and *Wine*, and *Bochs* and *Plex86* for general PC emulation.

- **DEVELOPER** VMware
- **WEB** www.vmware.com
- **PRICE** \$299 (download edition) \$329 (boxed)

VMware's Workstation, the PC virtualisation aimed at the desktop user rather than the server, saw many improvements with release 3.0 – including support for USB devices in the guest OS and faster, more flexible disk access. The 3.1 release, following not that long afterwards, is, as you might guess, largely a maintenance release. Most of the tweaks seem to have been aimed at getting the system running faster on Windows machines – although there are a few bonuses tucked away for the Linux user.

More guest OSs are also now supported under 3.1, including various flavours of .NET Server and Netware 6.0. *VMware* also now offer several Guest OS kits, virtual machines preloaded with a guest operating system. It is now possible to buy Window XP Home and Professional Edition and Windows 2000 as virtual machine images in this way.

The upgrade to 3.1 is free to existing owners of *VMware* 3.0, and there is a special offer for users of *VMware* 2.0.

Getting virtual

Installation of 3.1 is simple. Unpack the tarball, launch a script, and you're away. If you're upgrading from an earlier release, then it even backs that up for you. Virtual machines created under 3.0 should work transparently under 3.1 and the installer script transports virtual machines from earlier releases. This may cause problems, depending on which guest OS you are running, because of the significant changes between versions 2.0 and 3.0. Portability of virtual machines is one area that has been addressed in 3.1, and, for instance, the



Distros such as Mandrake 8.2, that include XFree86 4.2.0, should install smoothly on *VMware Workstation 3.1*

configuration files can now use relative rather than absolute path names.

As far as Linux guests are concerned, one of the problems we encountered with *VMware* 3.0 was with the changes in the video device emulation. The modifications wrought in *VMware* 3.0's virtual graphics card – including a new PCI ID – meant that the old *VMware* X driver no longer worked. You had to install *VMware*'s X driver and configure X manually. Thankfully, the new driver has now been integrated with XFree86 4.2.0, so any distro using that should install. This was the case for Mandrake 8.2, for example.

Workstation 3.1 ships with an updated range of the necessary kernel modules pre-built – including modules for Mandrake 8.2. If you use an unsupported kernel, then you'll have to build these modules yourself. The *VMware* modules compile cleanly against the latest stable kernel (2.4.18), and the config scripts have had tweaks to cope with the device filesystem.

Killer serial

So what's new in *Workstation 3.1*, then? Well, very little, architecturally. The one big change is a brand new implementation of the serial port emulation. A virtual machine may have up to four serial ports – each of which can be mapped either to a physical serial port on the host, a file, or a pipe. The latter option is incredibly useful and is perhaps a sign of the market that *VMware* have found for this product. Developers typically use the serial port for kernel-level debugging – no matter what the OS. *VMware*'s new serial port emulation lets you, for example, run the kernel you are testing in a virtual machine and capture its output on a debugger running on the host. *VMware* even lets you hook up one virtual machine to another – so you could run your debugger in a virtual machine – handy if your doing Windows development on a Linux box and have a debugger hosted on Windows, say.

VMware Workstation 3.1 reinforces

this product's place as the premiere PC virtualisation. It is fast, flexible and – since version 3.0, certainly – easy to use. It has a hefty price tag, however, and this reflects the target market. Running *VMware* on your desktop at home just to gain MS compatibility would be overkill – *Wine* or *Win4Lin* would be a much more cost-effective solution. *Workstation* is all about making working with multiple operating system manageable, and this is a job it excels at. [LXF](#)

LINUX Format VERDICT

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

No revolutionary changes in 3.1, but this solid PC virtualisation keeps getting better.

LINUX Format **RATING**
8/10

WINDOWS COMPATIBILITY LAYER

Codeweavers Crossover Office 1.0.0

Hoyt Duff reviews the dawning of a New Age. When the Moon is in the Seventh House and Jupiter aligns with Mars . . .

Allows MS Office and other Windows apps to run directly under Linux. Compare VMware.

- **DEVELOPERS** Codeweavers
- **WEB** www.codeweavers.com
- **PRICE** \$54.95 (download edition)

Let's get the obligatory rhetoric out of the way: *Wine Is Not an Emulator*; it is an implementation of the Microsoft Windows API for GNU/Linux (and FreeBSD and BeOS). The Holy Grail of *Wine*-dom has been the successful running of MS Office without the need for any native parts of the Microsoft 9x operating system.

As far as *Wine* is concerned, the Age of Aquarius has arrived, ushered in not by planetary alignment, but by Jeremy White and the wizards at Codeweavers, a development and consulting company specializing in *Wine*. With their experience wrought from work with Corel and other clients, they have released their own previews of *Wine 1.0*, as well as an

interesting mix of *Wine* and proprietary code that allows MSIE browser plugins to be used with Linux browsers, *Crossover Plugin*. They have even worked on the infamous Lindows OS, a Linux-based OS that promises to offer a seamless integration of MS Windows usability.

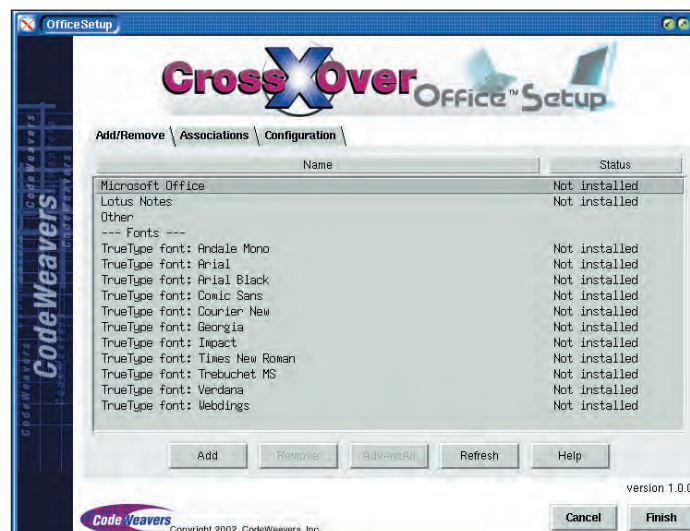
Fortunately for us, their work with Lindows led Codeweavers to develop and release a product that again combines *Wine* with proprietary code and which permits the installation and use of MS Office (*Word*, *Excel*, *PowerPoint*, *Outlook* 2000) as well as the *Lotus Notes* client. And it works surprisingly well. Here's what happens:

The *Crossover Office* app can be purchased through the Codeweavers website. Individual new users pay full price (US\$54.95 download edition, CD available for US\$10 more, quantity discounts and bundles available); registered users of the *Plugin* product receive a discount. The app is presented as a shell script. It can be installed as root (for multiple users) or as a single user, depending on which license you pay for. The

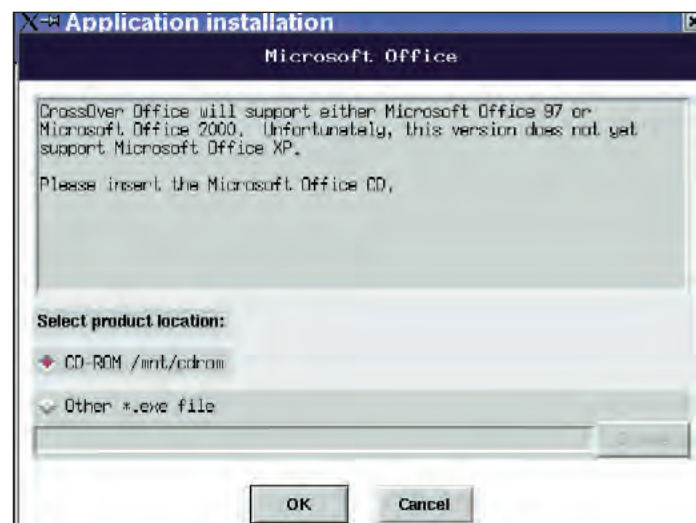
Funky, funky MS Office weirdness

True, there are a number of text-file conversion utilities and several office suites that can handle MS Office formats. Some approach perfection, but none have achieved the 100% compatibility cited by pundits as the ultimate goal. The only way to do that without Windows has been to use MS

Office itself with *Wine*. Doing that has required the use of native Windows DLLs and some major mojo along with more than a little luck. Even then, MS Office performance was, well, 'funky' – it looked a little strange and some components didn't work well (e.g., blank menus, 'File Save' didn't work).



The main Office setup screen prior to any installation. A combined Office/Plugin app is available that installs both programs.



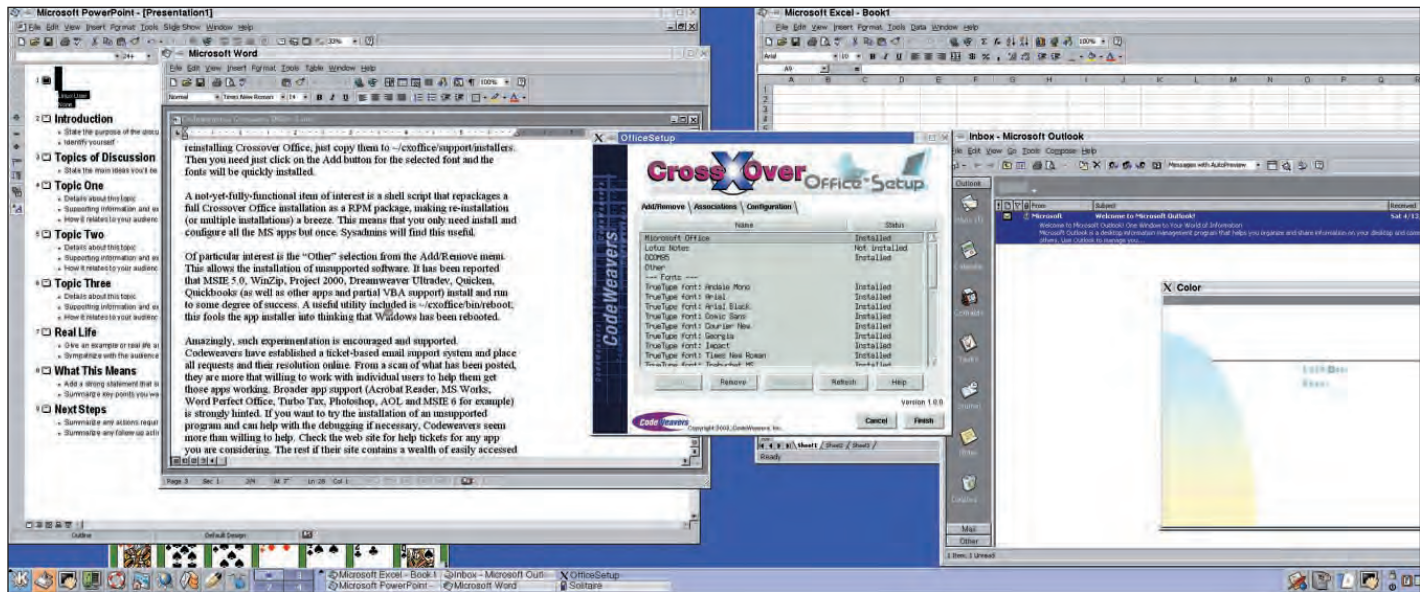
The path to the CDROM is hard-coded in the *Wine* config file. The 'Other' choice allows you to attempt to install unsupported software.

functionality of a root install is not complete in version 1.0.0 but is promised for a coming update.

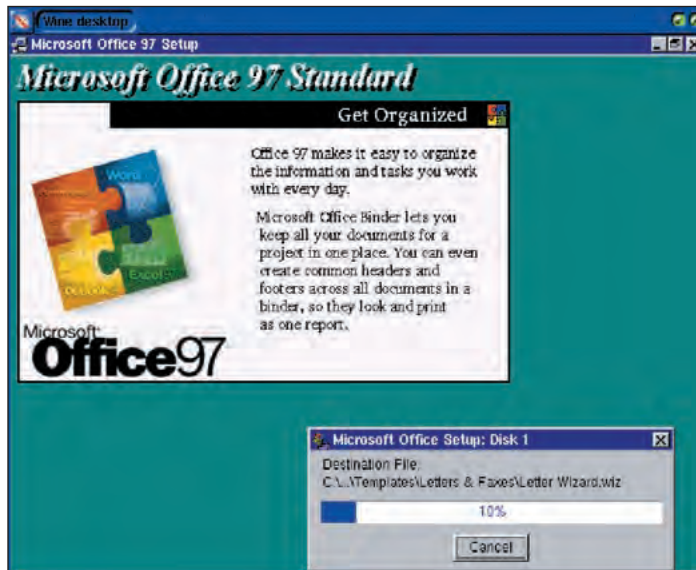
After installing, the app launches a setup GUI. Installation is straightforward and well documented through the online help – some of the best online docs we have seen (extensive HTML docs for this app, as well as *Wine*, are also included). The application also provides for the installation, freely downloaded from the Internet, of the MS True Type fonts that Office expects to see on the system, allowing your documents to look 'correct' in their presentation.

For evaluation, we installed MS

Office97 Upgrade (up to MS Office 2000 is supported, no support for anything XP), which required us to validate with a MS Works floppy before installation could proceed. There was a small adjustment necessary (see the *Tweaks* box) and Office installed flawlessly (this review is being written in MS Word). Upon completion appropriate menus appeared in our Mandrake 8.2 installation (Red Hat, SuSE, Debian and Caldera are supported as well, but it should work with all modern distros) and, well, everything just worked as advertised. We could even cut and paste from the X desktop, unlike some



MS Word in all it's glory. Note a slight problem with fonts in the PowerPoint window. Is that MS Solitaire we see peeking out from under there?



It's Alive! It's Alive! – Now I can read those blasted .docs!

native Linux word processing apps! Printing worked as well.

The implementation of fonts is interesting. The setup menu allows you to easily download and install MS fonts from the Internet. They are installed to `~/cxoffice/support/dotwine/fake_windows/WindowsFonts` (and can be used by the Linux font server if you know how to add fonts manually). A setting on the Configuration tab displays the path to any natively installed MS fonts from a dual boot installation and this feature can be enabled manually if necessary. Although not necessary, making any Linux True Type fonts available to Office is as easy as manually editing the *Wine* config file and adding an entry similar to the following that

points to your True Type fonts:

```
[FontDirs]
"dir1"="/usr/X11R6/lib/X11/fonts/TTF"
```

If you have saved the downloaded MS fonts and wish to reinstall them when reinstalling *Crossover Office*, just copy them to `~/cxoffice/support/installers`. Then you need just click on the Add button for the selected font and the fonts will be quickly installed.

A not-yet-fully-functional item of interest is a shell script that repackages a full *Crossover Office* installation as an RPM package, making re-installation (or multiple installations) a breeze. This means that you only need install and configure all the MS apps but once. Sysadmins will find this useful.

Tweaks

Get your config working

In order to install our version of *Office*, we needed to adjust the *Wine* configuration file at `~/cxoffice/support/dotwine/config` to change the floppy mount point from `/floppy` to `/mnt/floppy` and changed the CDROM device from `/dev/cdrom` to

`/dev/scd0` to allow supermount to work correctly.

Details of the config file are covered in the included *Wine* documentation, but configuration tweaks such as this are not easily accessible to the uninitiated.

Of particular interest is the 'Other' selection from the Add/Remove menu. This allows the installation of unsupported software. It has been reported that *MSIE 5.0*, *WinZip*, *Project 2000*, *Dreamweaver Ultradev*, *Quicken*, *Quickbooks* (as well as other apps and partial VBA support) install and run to some degree of success. A useful utility included is `~/cxoffice/bin/reboot`; this fools the app installer into thinking that Windows has been rebooted.

Amazingly, such experimentation is encouraged and supported. Codeweavers have established a ticket-based email support system and place all requests and their resolution online. From a scan of what has been posted, they are more that willing to work with individual users to help them get those apps working. Broader app support (*Acrobat Reader*, *MS Works*, *Word Perfect Office*, *Turbo Tax*, *Photoshop*, *AOL* and *MSIE 6* for example) is strongly hinted. If you want to try the installation of an unsupported program, and can help with the debugging if necessary, Codeweavers seem more than willing

to help. Check the website for help tickets for any app you are considering. The rest of their site contains a wealth of easily accessed information (kudos to the webmaster) that is well worth your time to read.

While far from flawless, this product is backed by an unconditional money-back guarantee so you have little to lose by trying the product if it may satisfy a need. Codeweavers returns all improvements to *Wine* back to the *Wine* source code, so further development helps the Linux community as well. [LXF](#)

LINUX Format VERDICT

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

With a money-back guarantee, superb documentation and excellent support, a decision to buy this app requires no deep thinking or hesitation.

LINUX Format **RATING**
9/10

Java RMI

Richard Drummond finds out how to do distributed programming Java-style.

- **PUBLISHER** O'Reilly
- **AUTHOR** William Grosso
- **ISBN** 1565924525
- **PRICE** £28.50

RMI or Remote Method Invocation, the Java-native mechanism for distributed programming, may be less trendy than XML-RPC or SOAP, but when you don't need to communicate with non-Java platforms RMI's efficiency and transparency make it an obvious choice. *Java RMI*, the book, explains and helps you make use of this technology in your Java projects.

This book assumes a basic knowledge of Java, but little else, since the foundations of Java RMI are covered in some detail – including Java streams, sockets and serialization. The book thus covers a wide area of ground, but, in fact, this introductory material features some of the best

explanations I've seen on what can be tricky concepts to grasp.

Java RMI is split into three parts. The first covers the basics before introducing RMI itself. Plenty of example code is included, and a particularly instructive example here is a printer server – first implemented using sockets, then using RMI. This really helps the reader to get an overview of RMI and drives home its advantages. Next, we are introduced to the example project, a bank server, which is used to illustrate topics in the remainder of the book. A lot of space is usefully devoted to the design and implementation of this project.

Part two progresses to deeper aspects of RMI. First, serialization and threading and how to test distributed projects are covered, before the author discusses the RMI Registry, naming services, and factories. Next is a chapter on the run-time aspects of RMI, which covers the mechanics of



RPC and distributed garbage collection, a must-read for those who have to know what's going on under the bonnet. The final section of the book deals with more advanced topics such as implementing custom sockets, security policies, HTTP tunneling and integration with CORBA.

There may be more advanced books on RMI out there, but *Java RMI* is an excellent guide for the beginner and provides a good overview of distributed programming in Java. It is well written and packed with clear explanations and useful tips. The sections covering the foundations of

RMI are particularly well executed. My only real complaint is that book seems to contain more errors than I would usually expect from an O'Reilly title.

Linux Format VERDICT

A well-thought out, clear and readable guide to distributed programming in Java.

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

JavaScript: The Definitive Guide (4th ed.)

Richard Drummond investigates the revised Rhino Book, the heavyweight guide to JavaScript.

- **PUBLISHER** O'Reilly
- **AUTHOR** David Flanagan
- **ISBN** 0596000480
- **PRICE** £31.95

Web-scripting has changed significantly since 1998, when the previous edition of *JavaScript: The Definitive Guide* was released. The top browsers of today are actually compliant with the latest JavaScript standard, ECMAScript v3. The fourth edition of this book has been revised and updated to reflect this, and documents the standard – rather than its implementation by any particular browser. The other major change is a repartitioning of material.

JavaScript is now finding favour not just for web-scripting but as a general-purpose, embeddable scripting language. Consequently, the treatment of the JavaScript language here has been split into two: the core language itself and client-side JavaScript.

The first two sections of this book are a guide to the core JavaScript language and to the environment provided by the web browser, respectively. The first is applicable to JavaScript programming in general, whether client-side, server-side or elsewhere. The section on client-side scripting covers the objects and methods available to JavaScript within the browser, including manipulating objects and frames, handling cookies, style sheets and, naturally, the



Document Object Model – the interface that the browser exposes to scripting languages.

The remaining four sections are reference material – documenting the objects and methods in the core language, the browser environment (up to DOM Level 0) and then DOM Levels 1 and 2; bringing up the rear is a life-saving cross-reference.

JavaScript: The Definitive Guide is not a cookbook, although plenty of example code is included; nor is it an introduction for beginners, although

every aspect of JavaScript is covered from the ground up. It is – what it sets out to be – the definitive reference guide for the JavaScript programmer.

Linux Format VERDICT

The fourth edition brings the Rhino book up-to-date and re-affirms its position as the JavaScript bible.

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

Free as in Freedom

Can this biography unearth the man behind the Stallman myth? **Richard Drummond** finds out.

■ **PUBLISHER** O'Reilly
 ■ **AUTHOR** Sam Williams
 ■ **ISBN** 0596002874
 ■ **PRICE** £15.95

Richard Stallman is legendary in the open source community as the creator of *Emacs* and the father of the Free Software movement. Less positively, he is also famous for public displays of petulance and his unwillingness to compromise – thus making it all too easy for others to simply dismiss him. But how much do you really know about Richard Stallman?

Sam Williams's biography takes a sympathetic, but ultimately unpartisan view of Stallman's life, achievements and failures. It is written for a general audience – and, inevitably, can be simplistic – but Williams isn't afraid to tackle some complex topics. The book

looks at Stallman's life from his childhood and his undergraduate days at Harvard, through his idyllic time as a hacker in MIT's AI lab, to his position today as a champion of Free Software. The book draws from interviews with Stallman himself, his mother, and colleagues and rivals from Stallman's professional career. A theme of the book is that the very traits from Stallman's character that have caused friction in the open source world – his stubbornness and unwillingness to compromise – are the traits that have allowed him to achieve what he has: a person less steadfast in their convictions would have failed.

Not surprisingly, recent events in Stallman's life get the most focus: this is the period that coincides with the explosive growth of the Internet and the rise in popularity of free and open source software, and is the best documented. Encouragingly, Williams's account of the 21st century Stallman is



of a man who has overcome his melancholy. Up until the late nineties – every since his departure from MIT – Stallman was a very tragic figure, the self-styled last true hacker. But the new Stallman seems to have found his place educating the world in ethical software: he no longer seems peeved by the Linux usurpers or stuck on the FSF's failure to produce a kernel for his GNU operating system.

Free as in Freedom paints a very human picture of Stallman as a passionate person who loves dancing and food, not just a crusader for ethical software. With all the recent hullabaloo in the industry about

software licensing, it's all too easy to forget just what Stallman has achieved. This book does well to remind us.

Linux Format VERDICT

A compelling and human account of the man who started the Free Software revolution.

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

INTERVIEW Sam Williams

Author of the (above) book *Free as in Freedom*

LXF: What prompted you to write the book?

SW: I'd interviewed Stallman a couple of times – done one or two profiles on him – writing for UpsideToday (www.upside.com) and one of my interviews attracted the attention of a large publishing house.

I see Stallman as the epic hero of the Internet, but if you approach Stallman in that way you will immediately attract his ire because he is not into the myth. He's very particular about how his message gets out. From a writer's perspective, that myth is very attractive.

LXF: What started your interest in Free Software?

SW: I'm not a hacker – but I had covered the open source beat for Upside from '98 to about the end of 2000. My first encounter with Stallman was at a Linux trade show. They were talking about the release of

GNOME 1.0, and I asked him what this release meant for the sales of Linux. It was like walking into a right-cross; He just said "I don't care about sales and I don't call it Linux, I call it GNU/Linux". I had to step back. This was the start of his big media come back – this whole idea of educating people.

LXF: Do you think Stallman is too extreme in his views?

SW: No. I wouldn't say he was extreme. He compromises every once in a while. A lot of his power comes from the fact that he doesn't compromise. The Stallman view is always to look at software from an ethical perspective. His idealistic position is untenable. Not everybody can live that way.

LXF: How crucial was Stallman? Do you think someone else could have created the GPL?

SW: The idea of the GPL was out

there waiting to be conceived. It was to Stallman's immense credit that he put it into play. Nobody has that combination of stubbornness, idealism and will that he had. If he hadn't done it – said that building a free operating system was a valid thing to do from an ethical perspective – it would have taken a decade for somebody else to decide to do it.

LXF: How much help was Stallman: Is this an official biography?

SW: It's unauthorized. He co-operated – gave me plenty of interviews, but there were always conditions. This book was originally conceived to be an electronic book, but Stallman didn't want to participate in an e-book that had a licence which wouldn't let people be able to share the book. I agreed to that. But the publishing house couldn't honour that request, so I put it on the shelf. Six months later my agent went to O'Reilly.



LXF: Has he read it?

SW: He doesn't like it. He says it's inaccurate in places. I went over one chapter with him and made a couple of fixes – I guess the fixes were valid, but I felt it was really chipping away at the reporter's point of view. It was never going to be a biography where he had the final say. There's always the possibility that he will flame the book, but I'm happy to get the dialogue out there – I'd rather do it in public than deliver a watered-down book. **LXF**

Roundup >>

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



Video editing

Hollywood watch out! Nick Veitch takes a look at current non-linear video editing software for Linux, and turns up some gems

Although we often hear the assertion that in some way Linux is not 'ready' for the desktop, the range and quality of desktop software available for the platform tends to contradict this. Even in fairly niche areas of desktop work, there is plenty of choice.

In recent times the advent of low cost camcorders, capture hardware and the introduction of consumer DV cameras has opened up the market

even further for what we used to call Desktop Video. And the good news is that whether you have an analogue camera or a digital one, a capture card, a firewire port or even just a webcam, there is undoubtedly some software that will suit you.

In this roundup we have tried to focus on editing solutions. There is so much software related to changing between file formats, capturing, processing, etc., that we cannot cover it all in this feature – but there is a

useful box on other software included at the end of this roundup.

When we consider video editing software, what features are we interested in? As with any software, the user interface and general ease with which things can be accomplished is a strong factor. With video editing

Our selection at a glance

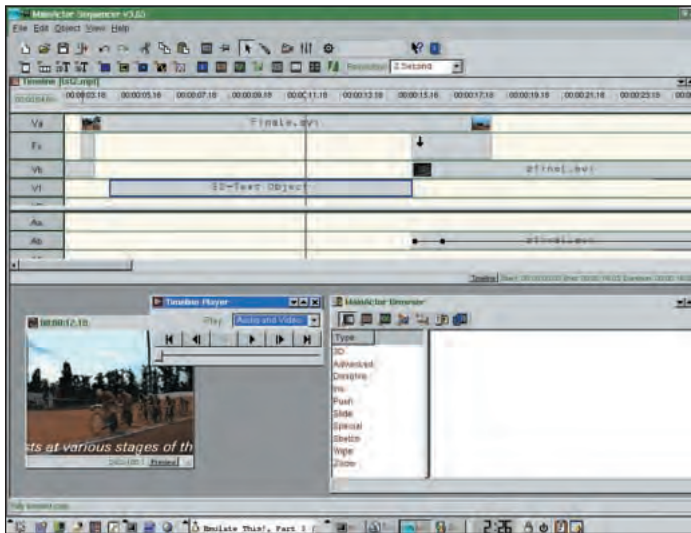
- Mainactor
- Kino
- Broadcast 2000
- Cinelerra
- Linux Video Studio
- Other projects at a glance

software we must consider specifically what it is used for. Most people will presumably be using footage they have generated themselves, so good support for capturing might be useful. Similarly, footage could come in a variety of formats depending on its origins. There really are hundreds of combinations of file formats and codecs, some of which are proprietary, so expecting one application to be able to cope with them all might be a little ambitious. However, good support of the open file standards would be advantageous, especially if the software doesn't cover the whole gamut of editing functions.

The actual editing process might involve specifying in/out points, handling transitions or wipes between one video source and another, titling, and other effects. Good previews of these are essential – it's very annoying to spend ages outputting a file to tape or whatever, and only then find that the titles are jaggy or there is some glitch in a transition.

Finally, you also need to be able to output the resulting work in some suitable way. The application which can manage all this should be pick of the bunch!

'The good news is that whether you have an analogue or digital camera, a capture card, a firewire port, or even a webcam there's some software that will suit you'



Mainactor's timeline approach is powerful, once you get the hang of it.

Mainactor

The oldest desktop video tool?

■ **VERSION 3.6** ■ **WEB** www.Mainconcept.de ■ **PRICE** \$99

The **MainActor** software has been around for years – since 1993 to be precise – which is a long time in desktop video terms. MainConcept, the publisher of the software, first released an Amiga version when the Amiga was the only viable desktop video platform. Since then *MainActor* has been ported to Windows (including NT), OS/2 and Linux. Along the way it has acquired an impressive range of features, but still maintains much of its ease of use from the earlier versions.

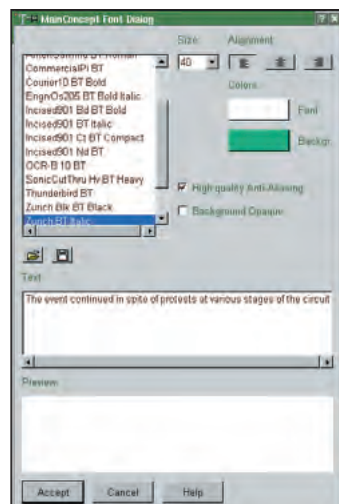
It has to be said that this is probably the most complete video editing solution for Linux, though we'll come to some of its drawbacks shortly.

If you have used any of the previous versions on the different platforms supported, the interface should be fairly familiar to you. The top timeline shows the various objects and channels in your production, and the audio tracks associated with them underneath. A small preview window and controls sit bottom left and the object browsers sits bottom right.

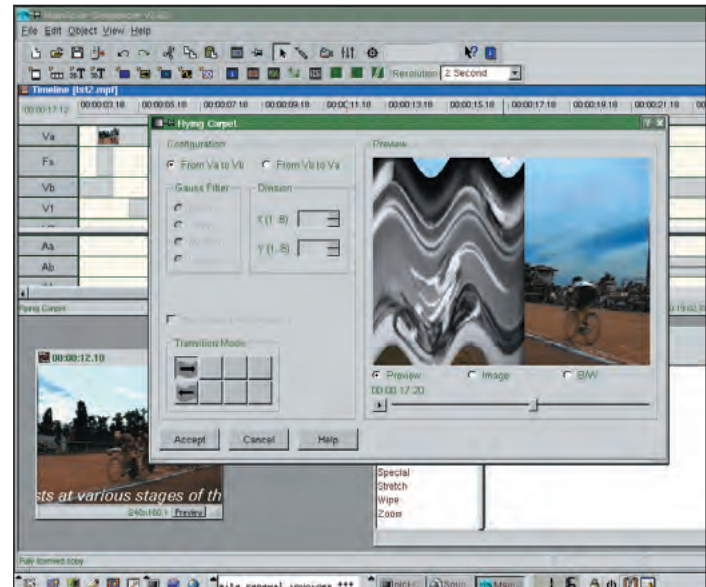
The timeline shows the time, and grey boxes with preview thumbnails denote the video clips. These can be inserted into a number of different channels by right-clicking the channel and loading an appropriate clip. It isn't possible to manipulate the clips themselves in this main window, but



Advanced overlay and compositing effects are available



There's a choice of supplied fonts for titling and captions.



A fairly comprehensive array of effects are catered for, with previews.

you can specify start and endpoints of the clips to use.

Effects and transitions are inserted as objects into the FX channel, which makes it easy to see, visually, where they sit in your production. There are a large number of transitions available – over 50 in fact – which cover all the standard wipes and fades that you are likely to want, and then some. The specifics of the transition can usually be edited, and timing it is a simple matter of dragging out the effect's block on the timeline.

As well as transitions, there are six 'video paths', which are basically 2D or 3D spinning planes of video – a neat effect if you don't overuse it. Some aliasing can become visible here, but at high framerates and good resolution it probably won't be that noticeable.

There are also standard effects, 30 of them, to zoom in on parts of the video field, artificially age your footage, and some more useful ones such as the colour and fade effects.

Unfortunately, although the Linux version supports various DV formats, it doesn't support DV capture or output to a DV device. Mainconcept are working on a version of the DV capture software for Linux, but for the moment you will have to rely on other tools to grab raw footage and export it to tape, at least if you are using DV.

The *macap* software included will capture from any *Video4Linux* supported device though, so if you are using a supported capture device, you'll be fine.

The only real drawback to *MainActor* is the rather limited file support. With AVIs this can be excused because of the sheer number of codecs employed, but even MPEG-I and some DV formats are not recognised. Output wise, as well as a slew of animation formats and stills like FLI, PNG, TIFF etc, video can allegedly be coded to Quicktime, AVI (some formats), AVI-DV(type1) and Realplayer. Unfortunately, *MainActor* seems to be the only software that can subsequently use these formats, with the exception of Realplayer, as non of the other software or standalone players could get to grips with them. Still, that is not so much a reflection on *MainActor* as on the sorry state of video formats – there are far too many of them, often covered by proprietary codecs not supported anywhere on Linux (or other non-Windows systems).

An update to this software will shortly be available, which we will review in full in a future issue.

LINUX Format VERDICT

Installation	7/10
Documentation	6/10
Features	9/10
Ease of use	8/10

A solid performer, but it badly needs DV capture/export support and more output options.

LINUX Format RATING

8/10

Kino

DV editing made easy.

■ **VERSION 0.5** ■ **WEB** www.schirmacher.de/arne/kino

According to the author, Kino should become the *vi* of the video editing world. While that may send uncontrollable shivers down the spines of some, others have probably pricked up their ears (or, er, eyes perhaps) at this point.

The main impetus behind this bold statement is that most of the operations available in *Kino* have keys associated with them, so there is little need to strain your hand with the mouse. One definite advantage this has over other software is that, once you have learned which keys to use, repetitive editing jobs can be done in seconds, and even complex cutting and stitching should be fairly easy.

Keyboard controls aside though, what features does *Kino* offer? The killer-feature here is that this is the only application under test which reliably supports IEEE1394 DV cameras. If you have such a device, you need read no further (although you may wish to consider one of the other applications for other editing tasks). Of course there is always a downside. Currently, DV devices are the only sorts supported for capture, so if you have other capture hardware, you'll need other capture software too, and you'll need to be able to generate the AVI/DV files which *Kino* supports.

Using the IEEE1394 module, and the *raw1394* and *video134* code, this

software can grab in good quality direct from the camera. A limited amount of AV/C is also supported, allowing you to control the camera device remotely, as with similar consumer software on other platforms.

Through various libraries *Kino* supports capturing to DV1 and DV2 streams, including associated audio.

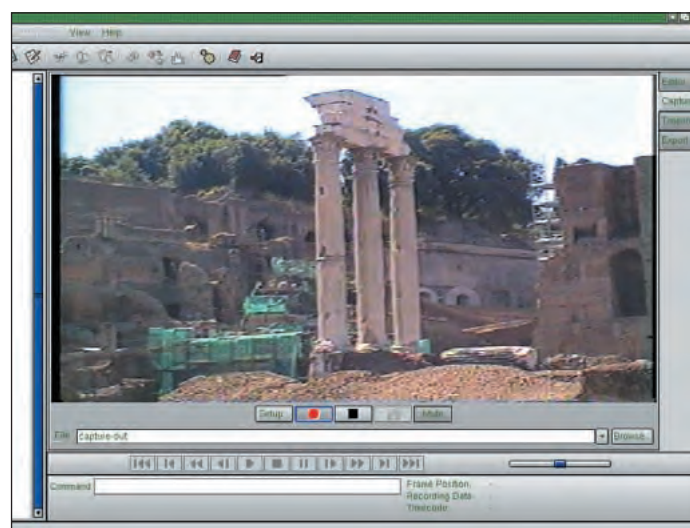
The *GTK* interface is simple and uncomplicated. Clicking on one of the four tab views takes you to the Editor, the Capture screen, the Timeline screen and the Export screen.

The timeline feature might not be exactly what you expect. Essentially this is a thumbnail view of the sequences you currently have loaded. The thumbnails are generated automatically at regular intervals, and appear in the panel with the associated frame number beneath. Clicking on the images takes you to the editor window at that specific frame. Useful, but not perhaps what you might be expecting.

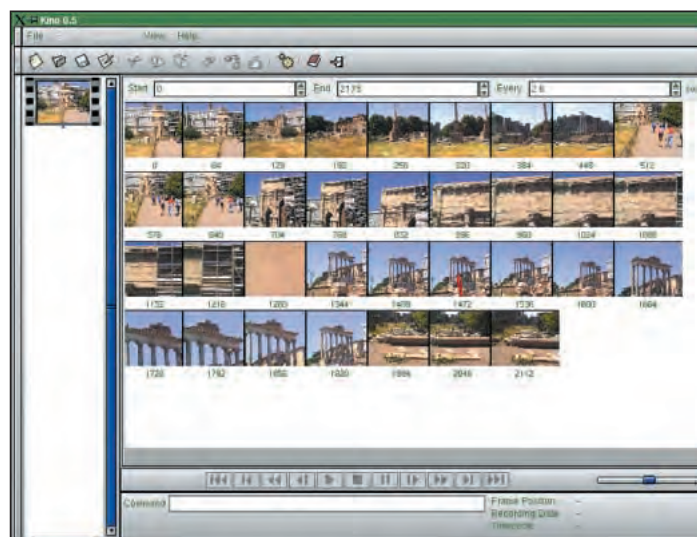
Export handles output to file or camera, and the capture screen allows you to grab stills or sequences from the video source. The Editor screen is probably where you'll spend most of your time though. The transport controls at the bottom should be familiar enough, and all of these are also represented by keystrokes – as are many of the other functions. A



A simple GNOME interface hides some serious functionality.



Capturing from DVD - a choice of preview modes is available.



Thumbnails can be generated at user-defined intervals.

groovy slider on the right operates as a mouse driven jog shuttle control, but if you have a real USB one, you can configure *Kino* to use this.

The only real drawbacks to *Kino* at the moment are those related to it being not quite finished. It captures sequences from camera. It can open DV1 and DV2 files. You can edit these files to the extent of defining in and out points, splitting sequences and rearranging them. You can also export the edited movie in a variety of ways, including saving to a DV1 or DV2 AVI file, outputting a sequence of JPEG stills, or record back to a DV device.

But that's about it. It works, but as yet there are no transitions, no titling, no effects of any sort, and, perhaps most restricting, no way to process audio separately from its parent sequence. That means no way of re-dubbing a sequence or layering additional audio tracks, which more or less excludes any use of the software

beyond editing together home movies at the moment.

If all you want to do is simple edits of recorded sequences, then this may well suit you. If you intend putting together more of a production, you'll need to look elsewhere at the moment. What currently exists in *Kino* (and to be fair, it is only release 0.5) is good, but it's got a way to go before it's a one-stop solution.

LINUX Format VERDICT

Installation	5/10
Documentation	4/10
Features	6/10
Ease of use	8/10

Still some way to go, but a very promising start by this easy to use app.

LINUX Format RATING

7/10

Broadcast 2000

The old faithful.

■ **VERSION** 2000c ■ **WEB** www.heroinewarrior.com

BroadCast 2000 was probably one of the best known Linux video tools, though since it's hibernation it has lost some of this notability. The software was dropped by Heroine Virtual due to their concerns over legal liabilities, but it still widely available and pops up in many distributions.

Broadcast 2000 uses the familiar idea of a timeline, which initially just consists of two audio channels. Video channels can be added, or you can create a new project incorporating new channels. The software supports

loading of standard MPEG video tracks (e.g. video CD format), but is easily fooled by non-standard tracks, which often cause it to add multiple blank audio tracks instead of a video one. Similarly, it will often load files it thinks are understood, but when the correct codec isn't found, the video stream will turn out to be blank.

The major drawback for *Broadcast 2000* is its limited file format support. This is mostly due to it ageing badly, but the capture options aren't inspiring. A sequence of stills or a Quicktime movie couldn't be honestly described as a plethora of options, but it has worked well for some.

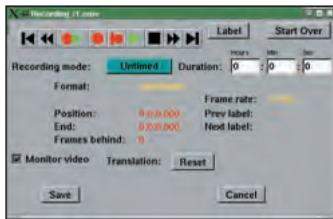
Audio support is very good in this software – it is one of the only applications that actually bothers about the audio tracks, and goes so far as to include fourteen different effects, as well as the ability to 'place' the apparent audio source. It's easy to



Some powerful effects and transitions are available.

underestimate the importance of audio to a finished sequence, but these developers got it right.

Although still with a large user base, Heroine Virtual's decision to drop it has effectively ended the useful life of this software. It has been included here mainly for reference, but also because it is still available on many distributions and has spawned the *Cinelerra* project (see below)..



Capture and export formats are rather limiting.

LINUX Format VERDICT

Installation	7/10
Documentation	4/10
Features	7/10
Ease of use	7/10

Fairly reliable, good effects and audio tools, but no longer actively developed (though still available).

LINUX Format RATING

6/10

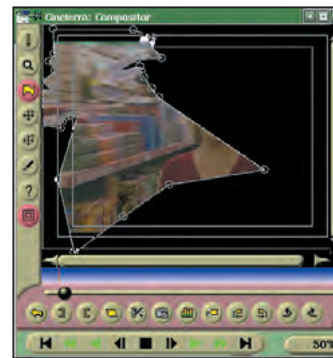
Cinelerra

The broadcast successor?

■ **VERSION** 041902 ■ **WEB** www.heroinewarrior.com

From the ashes of Broadcast 2000, *Cinelerra* is still a work in progress. We checked out the 041902 version source code for the purposes of this roundup. There are no official builds,

though there are quite frequent updates – it's worth checking the sourceforge site (and also check the discussion forums as there may be helpful advice on building it).



An editable mask layer – handy, once it's finished.

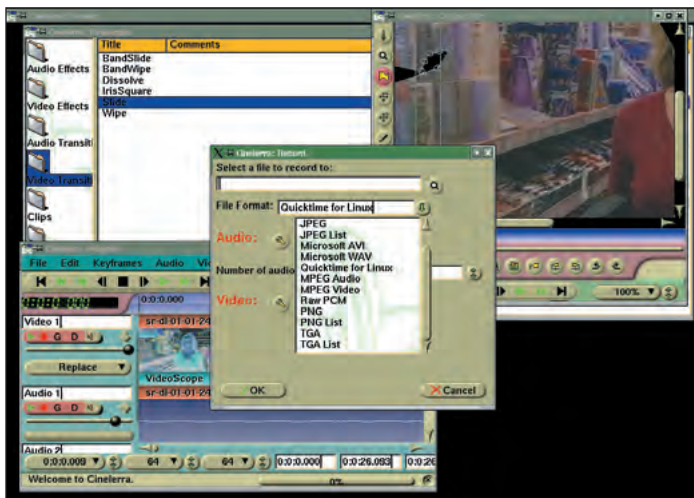
As this is a work in progress, and a snapshot build at that, it's no surprise that it took a bit of convincing to build on our test system. No doubt when the project nears completion there will be a slightly easier build process (hopefully involving *automake*, etc!).

Visually, apart from some graphical updates, *Cinelerra* looks fairly similar to its predecessor. The main differences are an update to the capabilities of the system, in terms of effects, transitions etc., and a change to the way filetypes, are supported, making it easier to create something you might be able to playback on a Linux system.

Direct recording to DV failed to work on our test system, possibly because of difficulty finding the DV device, but when it's finished, the DV support should be a lot less flakey than that in *Broadcast 2000*, and the extended range of supported file formats means you may actually be able to do something useful with it.

Audiophiles are not forgotten – an impressive range of easily configured effects and audio transitions is included to make the most of what is almost as important as the video stream – the stereo soundtrack.

Cinelerra is bound to be a big hit with fans of *Broadcast 2000*.



Visually, Cinelerra looks like a brighter bcast, but it is quite different.

LINUX Format VERDICT

Installation	3/10
Documentation	2/10
Features	9/10
Ease of use	7/10

It's a little unfair to judge it prematurely – this project has great potential.

LINUX Format RATING

7/10

Linux Video Studio

MJPEG solution of choice.

■ **VERSION** 0.1.5 ■ **WEB** <http://ronald.bitfreak.net>

This project is very similar to Kino mentioned before, but with the focus on hardware capture using *V4L* devices. The capture part of the software is essentially a front end for the *lavrec* tools. *lavrec* only records MJPEG video, but it can use either

hardware devices (e.g. Iomega Buz, Miro DC10) or it can handle the encoding itself and use other generic *V4L* devices such as TV cards, cameras etc. There is no support for IEEE1394 capture.

Linux Video Studio relies heavily on

the MJPEG tools for its capabilities, which explains at least some of the restrictions on file support. There is a limited selection of supported file formats, but these tend to be supported well. Motion JPEG AVIs, Quicktime (not Sorensen) and *movtar* are supported for loading, so you may have to transcode files obtained by means other than hardware capture with an MJPEG device.

Encoding of finished productions is only supported in MPEG-1 and MPEG-2 file types, which, while obviously popular, is a bit limiting. Plans are to include divx and MPEG-4 in the future.

The editor itself is fairly well equipped, with basic scene manipulation tools, so you can tweak the in and outpoints, shuffle them around and split them up into different parts. This information can be saved as an editlist if needed, or used to encode a presentation into the formats mentioned previously.

Text overlays, image overlays and transitions are also possible, though we failed to get any to work on our system – but this is likely because of some problem with the *GTK* libraries rather than *Studio* itself.

Apart from the restricted choice of file formats for output, the export functions combine a wealth of options, by creating an easy to use interface for the *lavtools* software. Plenty of predefined defaults should enable you to create trouble-free VCD or SVCD files, as well as customised MPEG-1 or MPEG-2 files.

Linux Video Studio is easy to use and does offer some useful features. If you are preparing video for Video CDs or DVDs, this is a good choice.



With support for *V4L* capture devices, this should be a popular choice.

LINUX Format VERDICT

Installation	5/10
Documentation	3/10
Features	7/10
Ease of use	9/10

Easy to use, though currently restricted to MPEG file formats supported by *lavtools* and *MJPEGtools*

LINUX Format **RATING**
7/10

Other Projects worth watching

CROW

■ **WEB** www.crow.atu.com.au

Crow is a simple nonlinear editing tool which looked promising but seems to have been abandoned. However, the source is still available online at sourceforge, so if you wanted to resurrect the project, I guess a lot of people would be grateful!

Crow uses a simple timeline approach, not too dissimilar to that of *Mainactor*. It supports a limited number of transitions, but capture and export support is fairly minimal.

EffectTV

■ **WEB**
<http://effecttv.sourceforge.net>

Not an editor, but an effects tool which can manipulate video streams to create a range of weird and wonderful effects. *EffectTV* effects are used in



EffectTV in action. The software is used heavily in *Cinelerra* too.

Cinelerra, and can be used with a variety of *Video 4 Linux* software.

Although its main use is in live stage effects, it can be used to generate some interesting real time effects on video as you capture it, and is well worth a look.

Gimp Film

■ **WEB** <http://film.gimp.org>

The HOLLYWOOD branch of the Gimp project is an attempt to bring some useful film-editing functionality

into everyone's favourite image processing software. Such an app would at least be better able at batch processing stills and recording in a variety of output formats, much like old video favourites from years gone by (e.g. *ImageFX* on the Amiga).

Unfortunately, though we managed to check out the cvs version, we had difficulty getting it to build correctly, and consequently ran out of time to do it justice here.

dvgrab

Mentioned here mainly because it was authored by the man responsible for *Kino*, *dvgrab* will have to stand as an example of one of the multitude of capture tools available for Linux. *dvgrab* is specifically for grabbing DV-2 and DV-2 streams from an IEEE1394 device, which could be invaluable if your chosen editor doesn't support DV grabbing natively.

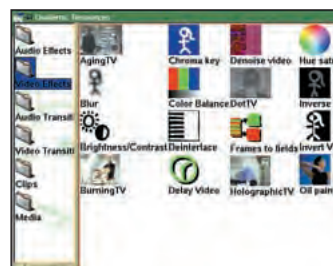
Video editing **The verdict**

As we have seen from the projects covered, the state of Video Editing software on Linux is very much in flux at the moment. There are some long established solutions, like *MainActor* and *Broadcast 2000*, but also a lot of entirely new and promising projects, like *Kino*, which haven't yet reached maturity.

One of the main problems, as we have highlighted here, with all of the software is file support. Many of the applications have difficulty with 'standard' multimedia files such as

Quicktime and AVI, mainly, as we have seen, because of the reliance of these formats on proprietary codecs which haven't been implemented under Linux. Even so, there are plenty of standard open codecs which, if all the applications supported them, would mean at least some sort of interoperability.

When it comes to interface design, there are also stark differences between the long established and the new. *Kino* and *LVS*, for example, seem to follow the easy to use and friendly style of interface design, akin to a



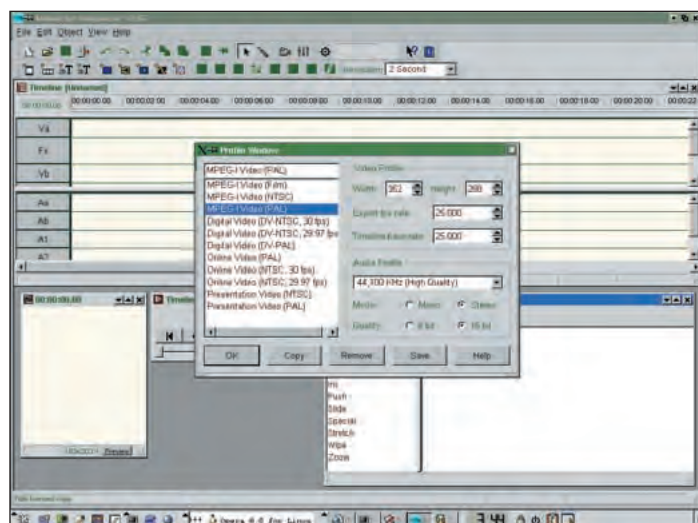
Cinelerra has a promising range of effects available.

product such as *VideoWave* on the Windows platform.

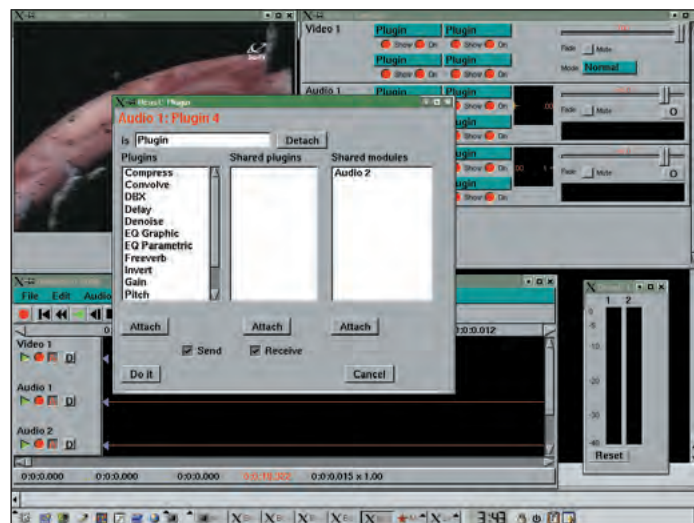
By contrast, software like *Mainactor*

and *Broadcast 2000/Cinelerra* rely on the more 'professional' timeline editing setup which may be a bit more confusing, but is perhaps easier to use when exact results are required.

If you could only choose one, then the most rounded application is *MainActor*. It is commercial, but it does have the best all round file format support, and most of the features you could possibly want. It is a little intimidating for beginners, but doesn't take that long to pick up, after which the interface is time efficient and powerful. DV support is a sad omission



MainActor is, overall, best of the bunch at the moment.



Broadcast 2000 – gone, but not forgotten by audiophiles.

Table of features

Name	Kino	MainActor	Linux Video Studio	Broadcast2000	Cinelerra
Version	0.5	3.6	0.1.5	2000c	041902
GUI	GTK	X11	GTK	X11	X11
Video4Linux capture	No	Yes	Yes	Yes	Yes
DV capture	Yes	No	No	unstable	unstable
DV to Camera	Yes	No	No	unstable	unstable
DV Camera Control	Yes	No	No	No	No
Still capture	Yes	Yes	No	Yes	Yes
Titling	No	Yes	Yes	Overlay	unstable
No. of video Digital effects	0	36	0	24	29
No. of Transitions	0	55	0	6	6
Audio Effects	No	9	No	14	11
Audio Transitions	No	No	No	No	1
Thumbnail views	Yes	No	Yes	no	no
AVI formats	DV-1, DV-2	Some	No	No	Yes
Quicktime	No	Yes	input	Yes	Yes
MJPEG	No	Yes	Yes	No	Yes
Overall	7	8	7	6	7

from this software's capabilities, but one that should be rectified shortly.

Kino is probably the best Linux software for DV support. Although *Cinelerra* and *Broadcast 2000* support it to varying degrees, both seem rather unreliable compared to *Kino*. Again *Kino* suffers from a narrow choice of supported file formats, but it seems whatever you are doing with multimedia on Linux, you are going to have to equip yourself with a slew of transcoding utilities.

Cinelerra is probably the one to watch in the future, though. The instability which is only to be expected from a work in progress is its major drawback at the moment, but this could develop into a very interesting product in the future.

And remember, we'll be taking a look at the new *MainActor* as soon as it's released. [LXF](#)

HotPicks

The best new open source software on the planet!



Maurice Kelly

Burning the midnight oil to bring you the hottest picks.

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

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

If you have any suggestions for things that we should cover, email us at linuxformat@futurenet.co.uk

HotPicks at a glance

AlsaPlayer	38
TeXmacs	39
FreeAmp	40
Samba	41
NcFTP Client	41
Liquid War	42

HotPicks award

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

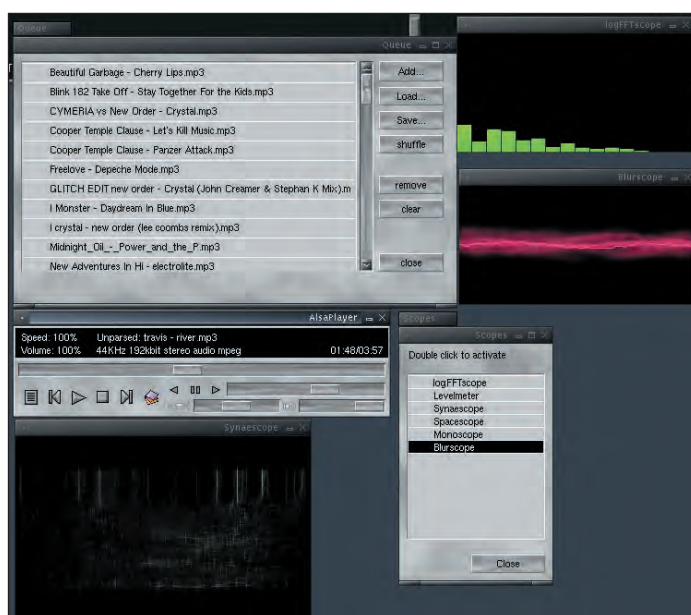


AUDIO PLAYER

AlsaPlayer

■ **VERSION** 0.99.57

■ **WEB** <http://www.alsaplayer.org/> <http://alsaplayer.org/alsaplayer-0.99.57.tar.gz>



There was reluctance to use a screenshot of the gimmicky scopes, but the *AlsaPlayer* interface isn't that exciting by itself, is it?

Everyone has their own particular favourite types of software. A lot of software floating about these days is of such a similar quality to its closest competitors that to choose one over the other comes down to the tiniest of features (or annoyances). The world of audio players is one such example – having evolved greatly as Linux multimedia has taken off.

For many people the first experience of such software will be something a little bit lame like the default CD players supplied with GNOME or KDE, but it's usually not long until they experience the fully-featured *XMMS*, which seems to be a prime choice for most distributions. *XMMS* might not suit everyone though, with its fiddly display, so for those who want something slightly different I give you *AlsaPlayer*.

AlsaPlayer started out as an audio player aimed at exploiting the features of the *ALSA* library, but has since expanded the range of supported audio systems to include *ESound*, *JACK* and *NAS*, as well as legacy OSS for older Linux systems.

The *AlsaPlayer* interface is pretty basic – a simple *GTK*-based display of the song information, some buttons to control playback and a number of sliders to play about with. It's not skinnable, doesn't have a whole lot of extra windows hanging off it, and looks pretty average, but don't dismiss it just yet. The plug-in architecture utilised means that later versions can have user-defined interfaces (although personally I find nothing wrong with the defaults) and allows the addition of support for alternative (and future) media types.

Playing tracks is simple – press the Queue button to bring up the queue window, from which you can press the Add button to start browsing your hard disk for tracks. You can load and save playlists to m3u files that can be shared with other applications, and there's a shuffle button as well for those who like life to hold some surprises. The only complaint about this system is that you have to choose the songs in the order you want to listen – some re-ordering of the playlist would be nice.

Oh, and if you really can't live without a few gimmicks you should choose the Scopes... option from the menu. It offers a number of display "scopes" which respond to the music to create weird and wonderful patterns – you know the score. These are based on a plug-in system too, so you can download more if you really must!

The nicest gimmick is the pitch control slider. Everyone loves to mess about with the speed of a track and *AlsaPlayer* allows you to not only speed up and slow down your tracks – Useful for people wanting to alter speed in order to learn how to play a song, or to sample it at an alternative speed – but also to play them in reverse, to listen to Satanic messages on your heavy metal records.

Playback of music is top quality. *AlsaPlayer* is a multi-threaded application and so avoids the potential for lockup noted in some other audio players. I was able to happily leave the program running through a long list of MP3s (I grabbed in all files to the playlist and gave them a shuffle) and the program sat happily in the background for a couple of hours.

The basic playlist and the fact that the buttons for the playlist and the menu aren't a bit more obvious are just minor quibbles. For the average music listener the ability to play a few MP3s and CDs is all that is really needed from this sort of software. If you want a straightforward audio player for Linux, with stable and responsive playback, then you needn't look much further than *AlsaPlayer*.

WYSIWYG SCIENTIFIC TEXT EDITOR

TeXmacs

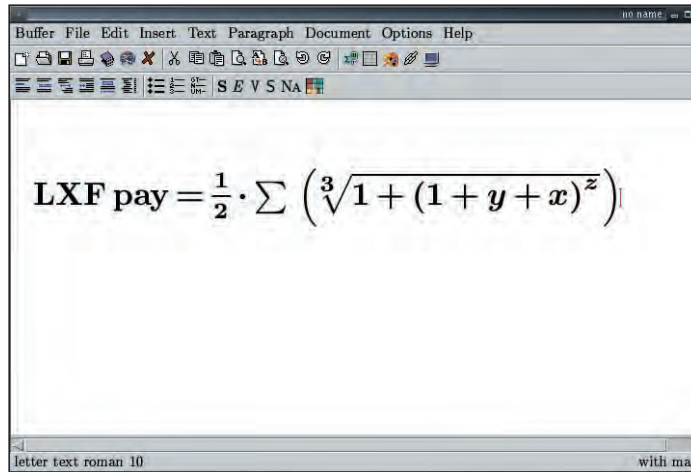
■ **VERSION** 1.0.0.1 ■ **WEB** <http://www.texmacs.org/> <ftp://ftp.texmacs.org/pub/TeXmacs/targz/TeXmacs-1.0.0.1-src.tar.gz>

Texmacs is an Emacs-like scientific text editor. However, unlike Emacs (GNU or X versions), it is actually a WYSIWYG application. It is not in fact based on TeX or LaTeX, although the website does recommend that you have LaTeX (and Guile) installed in order to run TeXmacs. Although it spits out its own file format, it is more than helpful to have a reasonably good grounding in the use of LaTeX or TeX in order to understand how best to lay out your documents.

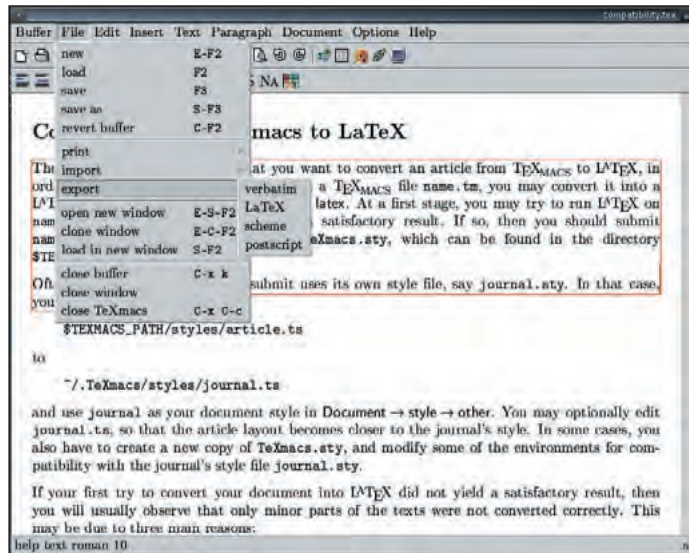
Neither TeX nor LaTeX are making any serious headway into the business environment because of the steep learning curves required to become an efficient user of either. However, both are popular within the academic and scientific communities, where the ability to finely control the presentation of a document is more important than the speed at which it can be churned out. Although not specifically aimed as such, TeXmacs forms a sort of halfway-house between ease of use and control. To call this application a text editor is a bit like describing a space shuttle as an aeroplane. While TeXmacs doesn't have all the features of a word-processor like word counts, and spell-checking, it offers so much more than a basic text editor could.

The first time it is opened you will notice that it takes a while to become ready for use. This is generally a one-time occurrence as the program has to automatically generate the fonts that it is going to use. When it has finished starting up you get presented with a menu, a couple of toolbars, the work area and a status bar. The status bar also doubles for command entry so keep an eye on it when you're experimenting with the options.

You can start typing straight away into the work area, although it is probably a reasonably good idea to have some sort of plan for your document structure. While it is feasible to use this program like you would any other word processor, for serious document production it is a good idea to use the TeX-like features and give the document a sensible structure.



Okay, I admit it – my mathematical ability is currently nil, but I like pretending I'm Einstein!



It may sound like a program name, but TeXmacs is not LaTeX – it does do a reasonable job of exporting to LaTeX though.

One of the strongest features of this software is the ability to produce complex mathematical notations. From the Insert menu you can find the option to insert mathematics. This causes a new toolbar to replace the standard text toolbar. Having been out of academia for a while now I wasn't able to fully appreciate the range of options available, but it was certainly more than impressive and the representation on screen was top quality.

As well as being suitable for creating your own formulae, TeXmacs is also capable of rendering

automatically generated equations, which make it highly suitable for use with computer algebra systems. TeXmacs can be used as an interface to such systems (examples are listed at the website) or can be simply used to capture system output in your documents.

The default output format is TeXmacs' own, but it does come with some basic conversion filters for dumping output in a LaTeX format which can be shared with other people who don't use TeXmacs. It also has filters for importing LaTeX documents. These are, by the author's

own admission, not perfect in any way, but they will be improved. As well as import/export using LaTeX there are also options to utilise HTML and scheme, as well as plain text. Printing is handled by converting the document to a PostScript format that is sent to your printer or to a file if you so desire.

The program is menu-driven but there are many keyboard shortcuts available. These are documented in the menus and many are similar to corresponding Emacs commands (such as C-x C-c to quit the application.) Again, like Emacs, confirmation of actions is carried out by typing into the status buffer at the bottom of the screen, and this buffer also displays the current keypresses which allows you to monitor what commands you are using.

Features include the use of tables and figures which are all appropriately labelled, and you can use external programs as plugins in order to create diagrams. Creating a proper document allows you to utilise options like the automatic insertion of a table of contents, index and glossary.

While you can't perform word-processor features like spell checking and word counts, there is the option to extend the program using Guile, an embedded version of the scheme scripting language. This allows you to extend the program interface itself as well as the features of the program. It is also possible to enable a shell within the document to capture command line output.

The menus and dialogs are all very nicely rendering using TeX fonts, which while not crucial to the operation of the program, certainly makes for a slick and professional application. There is comprehensive assistance available from the help menu, which is backed up by additional documentation on the website. This really is a top application which is of use to more than just scientific and academic types. It might be a little too bulky for use as a basic text editor, but it's just as handy for writing a letter to the bank manager as it is for typing up your thesis.

FEATURE-PACKED AUDIO PLAYER

FreeAmp

■ **VERSION** 2.1.1 ■ **WEB** <http://www.freeamp.org/>
<http://www.freeamp.org/download/src/freeamp-2.1.1.tar.gz>



Surely it must be a crime against humanity to create a theme like this?

So I've expounded at length on how nice an application *AlsaPlayer* is, when I turn instead to *FreeAmp* and start extolling its virtues as well. All of a sudden the *HotPicks* readers start claiming that yours truly has lost it, and is writing contradictory reviews. Not so, I retort. For while *FreeAmp* and *AlsaPlayer* are both audio players, they are pitched at slightly different users. Let me explain...

Many people like to listen to music whilst using their computers. Some just bung in a CD and listen to it from start to finish, or carry out a similar activity using some MP3s – this group of people is best served by a program like *AlsaPlayer*, which performs the tasks they require with a minimum of fuss. Then there are people who've converted a lot of their CDs to MP3 format – for those a meatier app is required. The ability to browse through a large collection of MP3s is essential, and so a different class of application is required – an audio jukebox.

As a user who regularly has around 2700 songs on the hard disk at any one time, I like to be able to browse

my entire collection of songs within an easy-to-use interface, and construct playlists from the collection. When using Windows you can find me listening to tracks through *MusicMatch*, but to my disappointment, the Linux version is an older version running under *Wine*. Searching in Freshmeat revealed the existence of *FreeAmp*, which now satisfies my Linux needs.

FreeAmp is (as the name suggests) a free audio jukebox, available for both Windows and Linux. It is available in both compiled and source forms, both of which are fairly hefty downloads. If you are compiling from source you will need a few other things installed, including the *NASM* assembler, *GTK+* and *freetype*. More important is to download and install the *musicbrainz* library from www.musicbrainz.org – this is needed for CD lookups over the 'Net, and whilst you may not want it, you won't even get to compile *FreeAmp* without it. When you've compiled the *musicbrainz* source, don't forget to run **ldconfig** as root in order to add the library to the library cache so that *FreeAmp* can pick it up.



Once you've got *FreeAmp* installed the first thing you'll want to do is open up the MyMusic section (underneath the volume control slider) and start adding tracks to it. *FreeAmp* is not an MP3 ripping/encoding suite – you need to have your MP3s created before you start playing with *FreeAmp*. If you need some decent MP3 creation software then have a look at *Grip* – a *HotPick* back in LXF 26.

The MyMusic dialog is split into two sections, "My Music" and "Currently listening to" – you can add songs from your hard disk to the "My Music" section. *FreeAmp* reads the tracks in and organises them according to the internal track tags – tracks without tags will be deposited in the "Uncategorized Tracks" section for you to deal with later, if you so desire. Tracks with reasonable tags will be sorted into a hierarchy of artists and albums. Every track you add is also present in the special "All Tracks" folder. Other folders include ones for playlists, Internet audio streams and one to represent your audio CD player.

When you've added all your tracks to your library you can then go about creating playlists. Simply select the tracks you want in your playlist from your "My Music" pane, and right-click to choose the "Add To Playlist" option. It is possible to add an entire album, or even all tracks by a single artist if you right-click on the album or artist branches in the folder hierarchy.

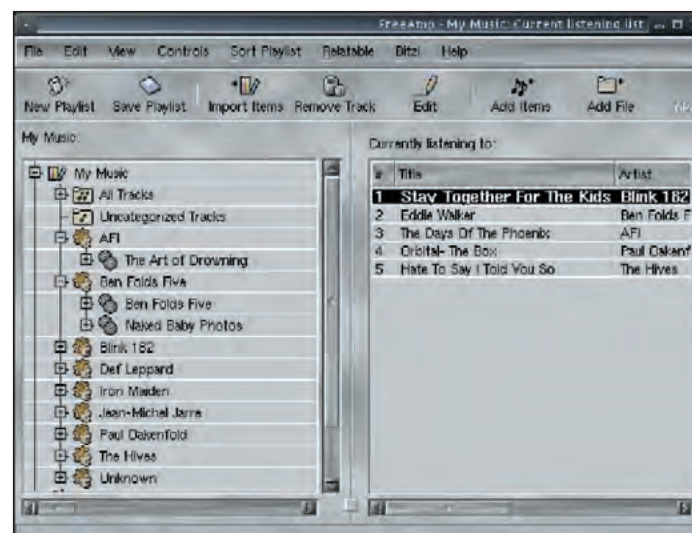
Once the tracks have been added to the playlist you can either start listening or re-jig the play order – the equivalent of programmed playback on an ordinary CD player. You can achieve

this by dragging and dropping the tracks using your mouse, or highlighting a track and selecting the Move Up and Move Down buttons on the toolbar. Interestingly it is possible to add both audio files and CD audio tracks to the playlist at the same time. This is not entirely amazing, but it is a feature I have not seen in any other jukebox, where the norm is to play either CDs or files, but not a mixture.

Playback of audio is very good, and the player is very easy to control. The default theme is more than serviceable, but there are a number of alternatives supplied with the source distribution, and there are even more available to download from the *FreeAmp* Web-site. Display modes include Normal, Medium and Mini – for those who need to be stringent with screen real-estate.

A comprehensive range of options is available to tweak, including the ability to choose the output source (sound-card, wave file, or ESOUND daemon) and some playlist display options. There are a number of Internet-related features such as "Relatables" – a system by which you can have music suggestions sent to you based on the music you currently have although I couldn't get any decent recommendations.

FreeAmp has a lot of extremely useful features – it's a solid option for serious audio enthusiasts who require more than basic CD/MP3 playback. I'm pretty fussy when it comes to jukebox software, but I find *FreeAmp* to be more than up to my requirements. New features are added regularly: definitely one to be keeping an eye on.



The extensive playlist controls and file organisational functions make this an essential program for those with a lot of MP3s to manage.

FILE AND PRINT SHARING SOFTWARE

Samba

■ VERSION 2.2.3 ■ WEB <http://www.samba.org/>

This might seem like a bit of strange program to choose for a *HotPick* – it hasn't had a major update for a while, and isn't that high profile an app on the desktop. Yet *Samba* is one of the most important Open Source projects and I thought it was due a mention for people who might not be entirely sure what it is.

Samba provides a means to share files and printing services from Unix boxes to an MS Windows environment, by implementing the SMB protocol. It is widely used amongst businesses (large and small) who utilise MS OSs on the desktop, whilst relying on Unix-type operating systems to provide back-end support. But with the growing popularity of home networks, it is not unusual to find a mix of OSs in some homes, too, and *Samba* provides an excellent means for these machines to share data.

You can get the latest version of *Samba* from their website, but it is probably more useful initially to install the packages from your distro CDs, in order to get it running as a service on boot. Whatever way you install it though, you'll find that one of your first ports of call will be to the `smb.conf` man page – it gives you plenty of details on the config options you can use within the `smb.conf` configuration file (which should be deposited somewhere around the `/etc` directory). This file usually comes full of healthy examples of how to use the options.

If you're feeling a bit frightened of the `conf` file, then have a read through the docs in order to find out more about the web-based configuration tool that comes with *Samba* – *SWAT* – usually enabled in `/etc/services` with:

```
swat          901/tcp
```

You will also need to add the

```
root@chatswood:chatswood.org.uk:/etc/samba # cat /etc/samba/smb.conf
[global]
    workgroup = DEADHEART
    netbios name = CHATSWOOD
    server string = Chatswood Samba Server
    log file = /var/log/samba/log.%m
    max log size = 50
    guest account = pguest
    security = user
    encrypt passwords = no
    smb passwd file = /etc/samba/smbpasswd
    socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
    interfaces = 10.0.0.1/255.0.0.0
    dns proxy = no

===== Share Definitions =====
[homes]
    comment = Home Directories
    browseable = no
    read only = No
    create mask = 0644
    writable = yes
"smb.conf" 44L, 1074C                               1,9      Top
```

The `smb.conf` file may not look the most exciting but you can usually find plenty of examples and comments on common configuration options.

following line to `/etc/inetd.conf` (or update your *xinetd* configuration):

```
swat stream tcp nowait.400
root /usr/sbin/swat swat
```

You should then be able to browse to *SWAT* on port 901 of your local machine. The tool comes with plenty of online docs – a useful resource if you want to edit `smb.conf` by hand.

Samba offers a full range of access controls to the file shares that you can create from wide-open to

password restricted. Shares can be created to represent just about any directory on your machine, and if you have a printer connected to your Linux box it is a fairly trivial task to share it as well.

It's not particularly glamorous but *Samba* has a very important rôle to play in computing. If you have any need to share files between Windows and Linux machines then you should definitely have a bash with it.

FTP CLIENT

NcFTP Client

■ VERSION 3.1.3 ■ WEB <http://www.ncftp.com/>
<ftp://ftp.ncftp.com/ncftp/ncftp-3.1.3-src.tar.gz>

The File Transfer Protocol (FTP) predates the Web and is still a very popular method for making available large volumes of files to both public and private users. Many people use nice GUI clients for connecting to FTP servers, or even use the FTP support of the file system browser, not realising that beneath the pretty façade, there's the same bunch of basic commands being sent.

NcFTP is both client and server FTP software, but here I'm looking at the former. Instead of using a pretty GUI, *NcFTP* operates from the command line – when you log into a server you are presented with a prompt (a bit like being at a shell prompt) and are expected to find your own way about.

However *NcFTP* tries to throw in

some of the convenience of GUI clients, while at the same time maintaining a lean tool that can be used in a wide range of situations. After all, what use to you is a GUI FTP client when you are having problems getting your X session up and running, or if you are trying to access sites from a machine with no X installation.

Just about every Linux installation will contain the standard FTP client – an FTP client that defines what "basic software" means! *NcFTP* manages to surpass the old *ftp* program by using such convenient tricks as filename completion and allowing a wide range of config options to be defined. File transfers are accompanied by handy progress meters which indicate both percentage completion and an ETA based on current transfer rates. *NcFTP* also has a comprehensive bookmarks

```
mikelly@moginie:chatswood.org.uk:/home/mikelly # ncftp
NcFTP Bookmark Editor

Number of bookmarks: 2
Open selected site: <enter> Chatswood chatswood, chatswood.org
Edit selected site: /ed UKShells telnet3.ukshells.co.uk
Delete selected site: /del
Duplicate selected site: /dup
Add a new site: /new

Up one: <u>
Down one: <d>
Previous page: <p>
Next page: <n>

Capital letters selects first
site starting with the letter.

Exit the bookmark editor: <x>

Ftp://mikelly:PASSWORD@telnet3.ukshells.co.uk
```

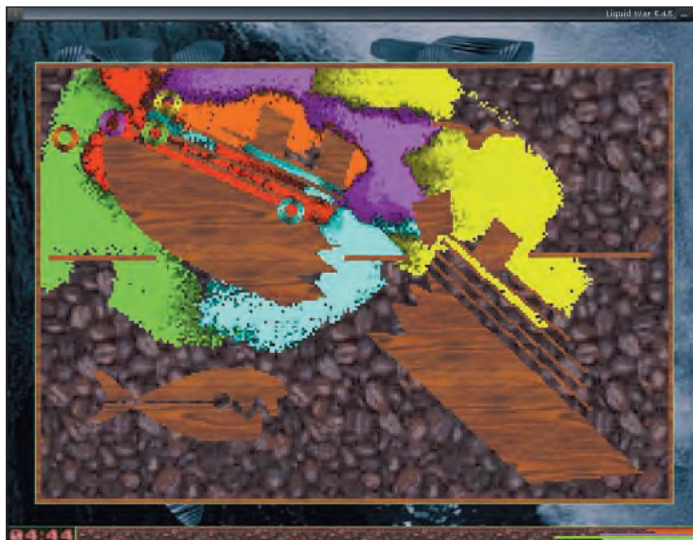
NcFTP allows you to store bookmarks for your FTP sites – very useful if you're always returning to the same place.

system that can be used to store the details of frequently accessed sites.

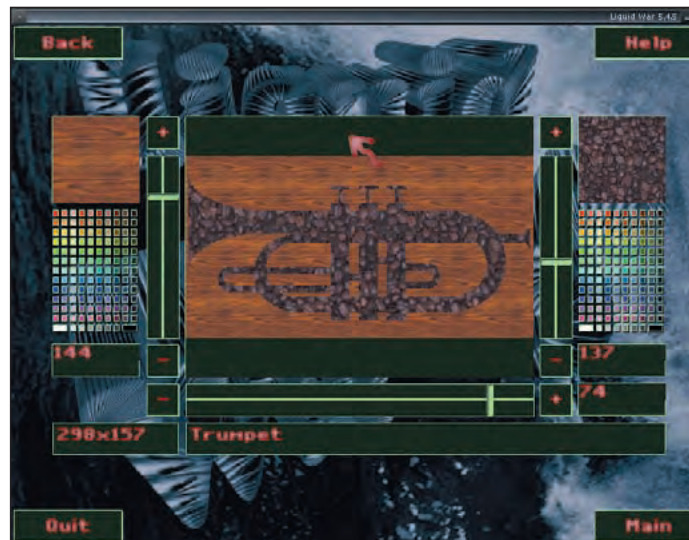
Support for passive FTP transfer is available, which is very useful these days when so many people are utilising firewalls as a result of being connected to the 'Net for longer periods of time. It also supports resuming downloads, caching directory listings and handling the download of directory structures. *NcFTP* is actually a suite of tools, and as well as the standard FTP "browser" comes with

command line tools such as *ncftpget* and *ncftpput* which can be used directly from the command line and within shell scripts.

If you've come from a program which does all the above, you could be forgiven for thinking that *NcFTP* is a step backwards. However, this program will do everything you need, from anywhere you might need to do it. It is fast and efficient, and with a bit of practice you'll probably find it even faster than mouse-clicking.



Obviously taking a bit of a beating here because of stopping to take the screenshot. Dunno why they always gang up on me though.



Some of the maps are just plain ridiculous – thankfully you can create your own very easily.

SNEAKILY ADDICTIVE GAME

Liquid War

■ **VERSION** 5.4.5 ■ **WEB** <http://www.ufot.org/liquidwar/>
<http://freesoftware.fsf.org/download/liquidwar/liquidwar-5.4.5.tar.gz>

Sometimes you come across a game which, going by its name, sounds extremely promising. Then you actually download the software, start using it, and your immediate reaction is “Why did I bother with this?” It seems really bad, the graphics are not that impressive, and you really don’t know why you are actually still playing it. Two minutes later, you are at the “I’m just giving it a chance” stage, and before long it is past your bedtime and you are still giving it one last chance. This is exactly how you would describe Liquid War.

To be honest I have no idea why I downloaded it – the name was catchy I guess. Reading the site as it downloaded I began to wonder how good a game about little dots moving around the screen like some goop from the X-Files could be.

Installation was a bit of a pain, as the game requires the Allegro gaming library (<http://www.talula.demon.co.uk/allegro>) to also be downloaded and installed. Both packages require a fair bit of compile time but were pretty straightforward. It also makes a change to have a game that’s not based on the SDL library!

So how does one engage in Liquid War? In fact, maybe *Liquid War* would make a good topic for a future “What On Earth...” article? To save you waiting for a couple of months, a Liquid War consists of between two and six players who control an army of thousands of fighters striving for complete domination. The game is played on a variety of maps utilising a cursor that is guided round the screen by four directional keys (completely redefinable of course.) When you move your cursor your army will attempt to follow it.

“Army?” I hear you cry in confusion. Yes, those dots on the screen represent your army. Every player is assigned a colour, and a bunch of dots on the screen in your colour is your army. The logic of battle is simple – your army is striving to get from where they currently are to where your cursor is. If an opponent is in the way, conflict occurs, wherein the opposing soldiers zap energy from each other. Once an opposing soldier succumbs to your might, he is converted and becomes one of your boys and his colour changes to that of your side. Relish those feelings of power and conquest – soon the whole world will be yours!

It’s an initially confusing situation – especially if you start off playing one on one against a computer opponent. You’ll certainly feel a small tinge of alarm when you see a mass of pixels heading your way and slowly engulfing your army. Note the movement of the opposing force – it’s a bit like liquid isn’t it? (And the penny drops.)

It really doesn’t sound that exciting, and to be honest, at first it isn’t. Every game plays the same for a while – it starts, the computer army swoops towards you and suddenly before you know it, you are watching your head-count drop off at a ferocious rate. You can observe your progress by keeping an eye on the coloured bars at the bottom of the screen – your bar will shorten as you lose men, and the computer’s bar will increase – it’s inevitable so don’t be too downhearted about it!

So far I’m not selling this game too well but that’s not important. At this stage you’ll be annoyed, you’ll be cursing me for suggesting you download this, and then you’ll be taking just one more game. To save you a bit of time may I suggest that you head straight for the Teams option on the main menu? You can use this screen to throw in a few other opponents which means that the blasted CPU player has to think about more than just destroying you.

The multiplayer option is really where this game is at. You can select up to 5 CPU players which can make the games rather crowded, but really

does give you more of a chance of actually winning. The really good thing about *Liquid War* though is the opportunity to play against other human beings. You could try to squeeze up to 6 people round one keyboard, but for those with network access you can engage in multiplayer combat across a network. Once you’ve started playing it this way, you really won’t want to stop.

So, we’ve established that the gameplay is pretty addictive. It would be nice to say that the graphics and sound are absolutely fantastic as well, but they aren’t. The graphics are more than serviceable, if a little blocky on some of the maps. You can add your own maps though so it’s not much of a complaint. One thing you might want to do is turn off the nausea-inducing “wave” effect that causes the maps to ripple by pressing F4 during the game. Sound was undetectable during the game – I can’t see it adding that much, and the lack of it did not detract to any noticeable degree.

There a number of options to tweak ranging from graphic/sound setup, to the actual rules of the game. The artificial intelligence of the CPU players is supposedly basic, but they do a good job of beating you on a regular basis.

With the ability to add your own maps, and play over a network with your mates, you might find yourself losing many hours in battle – or days and weeks, even. Even if they are the stupidest soldiers you’ve ever seen. **LXF**

Mozilla

MOZILLA!

King of the browsers!

cover feature



It's been four years since Jamie Zawinski threw his open party in San Francisco to celebrate the release of the Netscape Communicator code.

Andy Channelle probes the past, present and future of the beast they call Mozilla.

On March 31st 1998, Netscape Communications Corporation took the 'unprecedented' step of releasing the code for their flagship product, *Communicator*, into the wild where, it was hoped, it would "accelerate development and distribution of future high-quality versions of *Netscape Communicator* to business customers and individuals."

At the time Marc Andreessen, Netscape's executive vice president, said the plan was to tap into the 'talent of the net'. It would be a new way of doing things for a new era. "Everyone wins with the evolution of the Netscape source. Developers and programmers can participate in generating state-of-the-art products, and Netscape maintains its rôle as a standards-based technology leader in innovation and product development by delivering quality products that customers demand," Andreessen said at the time.

Four years on and the Mozilla hackers have finally achieved their goal: a stable, cross-platform web browser based on open standards. The big question is, is it too late?

Lizard basics

Mozilla is a suite of applications – comprising a web browser, mail/news client,

IRC application and HTML editor – that emerged from the decision by Netscape Communications to move away from closed source development back in 1998. Mozilla is also the organisation that sprang up in the wake of the source code release in order to shepherd the project to a fruitful conclusion.

At its launch the project's aspirations seemed breathtaking. Netscape executives talked of moving beyond simply displaying web pages and writing email to providing a complete OS-independent platform capable of running everything from a calculator to a fully featured spreadsheet. Despite this, Mitchell Baker – Mozilla's 'Chief Lizard Wrangler' – says the intention has always been first and foremost to create a world class Internet suite.

"In one sense, Mozilla 1.0 is very close to what was originally proposed," she says. "Mozilla is a suite of applications with a wide range of functionality. In this sense, it is close to its starting point. When the project was launched, Netscape was shipping an integrated suite of browsing, mail and news applications, and the plan was that future versions would be developed through the Mozilla project."

The desire 'to go beyond the web' was still there though, and the developers were soon hitting the limits of the Netscape 5 code. Baker says the plan to make incremental improvements to the inherited Netscape codebase wasn't enough to create the product that had been proposed. "A series of difficult decisions were made to rewrite almost everything in order to have a technological framework adequate to support the types of applications we had in mind," she says. As a result, Mozilla is also a set of technologies that create a robust application framework. "This is an enormous benefit. It's ambitious perhaps, but important for creating technology with ongoing usefulness."

Not everyone was pleased at the prospect of dumping so much work. In his rambling resignation notice posted on April 1st 1999, Mozilla pioneer Jamie Zawinski said this decision was instrumental in his decision to leave. "In my humble but correct opinion, we

should have shipped Netscape Navigator 5.0 no later than six months after the source code was released." The problem was, Zawinski said, that the Mozilla team "couldn't figure out a way to make that happen."

Zawinski's departure – hot on the heels of AOL's acquisition of Netscape – was seen by a number of commentators as the project's darkest moment; obituaries were published bemoaning 'the lack of responsible adult supervision' and suggesting it was time to strip the working bits from the corpse and 'get on with our lives'. Greg Knauss, one of the projects most vocal critics, wrote: 'Even the Project's most ardent supporters have to admit the possibility that their bouncing baby browser now more



"The project's aspirations seemed breathtaking, providing a complete OS-independent platform to run everything."

Keyword magic

Who needs a mouse?



Mozilla Keywords are a great way to speed up browsing (go to Bookmarks | Manage Bookmarks and select properties and choose a keyword to associate with a site), but they can also be used for searching from the address bar.

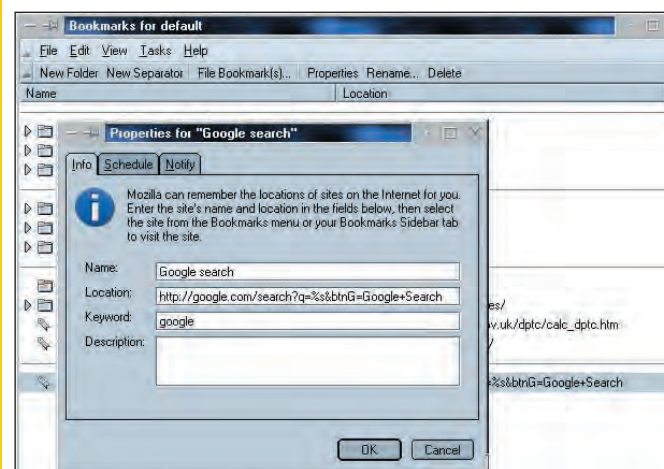
This trick will let you search Google by typing "g [search term]" in the address bar.

1 Bookmark
'http://google.com/search?q=%s&btnG=Google+Search'.

2 In the properties dialog define the keyword as 'g'.

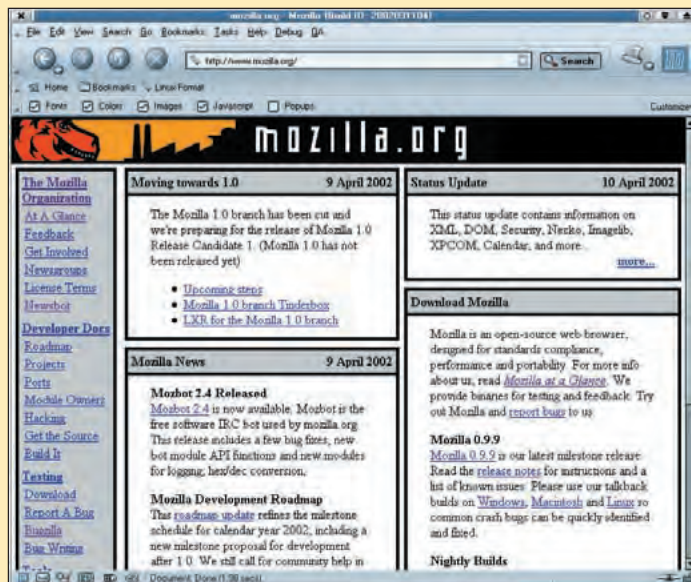
3 Type 'g' followed by your search criteria in the address bar. Hit enter and you'll be redirected to Google with the results presented as usual.

This trick should be adaptable for your favourite search site. Type a query, bookmark the result then edit the URL replacing your search string with "%s". Finally give your bookmark a memorable keyword or letter.

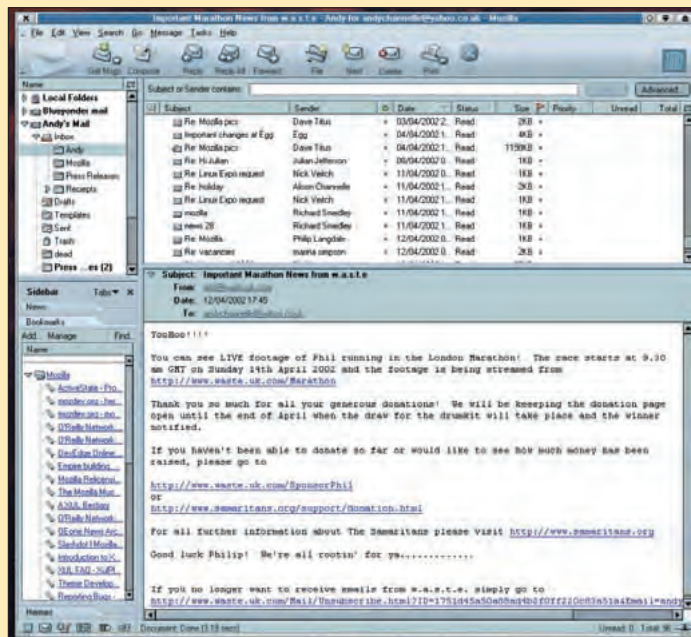


Set up a keyboard shortcut for quick Google searches.

Mozilla



Mozilla's default Chrome is the demure, pretty 'Modern'.



The best word to describe Mozilla's mail and news client is beautiful.

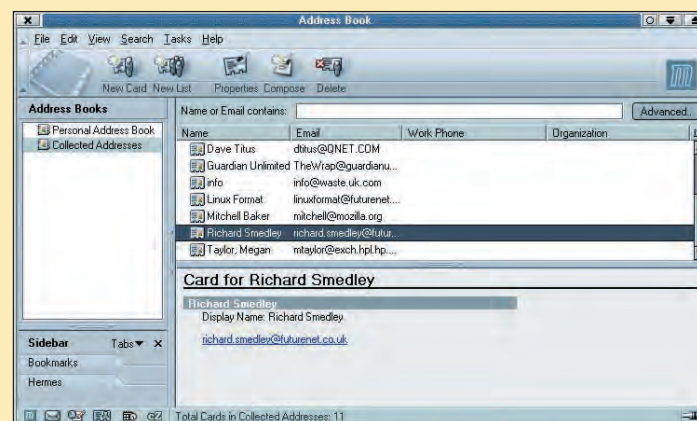


resembles a massive, festering cyst. Baker puts this mid-cycle disillusionment down to the initial hyperbole that accompanied the source code release: "The Mozilla project has sometimes been misjudged due to the unreasonably high expectations with which it was launched."

Despite this gloomy prognosis, every developer contacted by *Linux Format* expressed praise for mozilla.org's stewardship of this enormous project, particularly the organisation's ability to work with 'both major corporate players as well as a large number of volunteers' and



No frills web authoring courtesy of Mozilla Composer.



Alarms but no surprises, the address book gives you access to your contacts and integrates well with the mail client.

“Cross-platform nature makes it easy for IT managers to provide a consistent, corporate suite to all their employees.”

produce an extremely impressive codebase. Peter Bojanic, who recently joined the project as an associate staff member, says his experience has been overwhelmingly positive, and described Mozilla as a “critical part of Linux as a free operating system.”

Baker, meanwhile, counts the features that go beyond browsing among Mozilla's real achievements – highlighting the component model (XPCOM) and the XML-based user interface building tools for particular praise. “Our toolkit technologies can be extraordinarily useful for those building cross-platform applications.

We use them for the browser application suite, but others can use them to build a wide range of different applications.”

The Five Elements

The Mozilla suite, in contrast to the commercial offering from Netscape, is regarded by mozilla.org as a tool for developers; a starting point. Despite this it makes a very sound choice for day-to-day browsing or email management, and the simplicity of redesigning or branding the interface, coupled with its cross-platform nature, provides an easy way for IT managers to provide a, consistent, corporate

Gecko

The heart of the beast

At the heart of Mozilla is **Gecko**, a revolutionary custom layout engine that was designed from the ground up to support open Internet standards such as **HTML 4.0**, **CSS 1/2**, the **W3C Document Object Model**, **RDF**, and **JavaScript**. As **Gecko** essentially draws web content onto the screen, a lot of attention has been paid to optimising speed and efficiency, and this also

benefits the browser (or other Mozilla-based application) because **Gecko** is very good at rendering **XML** – the foundation of the entire project. This is the 'revolutionary' part. **Gecko** doesn't just render web content, it paints the container too; scroll bars, buttons, menus and every other screen element – it's like **M.C Escher** went into **User Interface design**!

suite to all their employees regardless of OS choice.

It consists of five main elements, with the Calendar a, currently unofficial, sixth.

NAVIGATOR The web browser is the core component of the suite and is synonymous with Mozilla itself. The layout is fairly standard and follows the conventions established with the original Netscape browser. Simplicity seems to have been the goal of the UI designers, putting the bare minimum onto the main toolbar though you can, of course, add extra toolbars with links to your favourite sites and search engines. At the bottom of the browser, to the left of the status bar, is the component bar, which is constant on every element of Mozilla and allows single click access to other parts of the suite.

Under the 'Tasks' menu, you'll find Privacy and Security options such as Cookie and Password management and the Tools menu providing access to additional applications you've added to Mozilla. Like the component bar, the Tasks menu is constant across all Mozilla applications.

Mozilla's version of Navigator also features two extra menus that don't appear on **Netscape 6.x**; these are the developer-centric 'Debug' with its useful Leak Detector, and 'QA' which contains links to various areas of the mozilla.org site.

Mail and News Email is now an essential part of life, so a complete browser suite should offer competent mail facilities (perhaps less-so from a Linux viewpoint). Fortunately Mozilla has one of the classiest looking mail clients (with the Modern theme at least) going.

Mail is IMAP and POP3 capable, integrates well with the address book

and calendar and, as it has **Gecko** at its heart, can handle the dreaded HTML email without even breaking into a sweat.

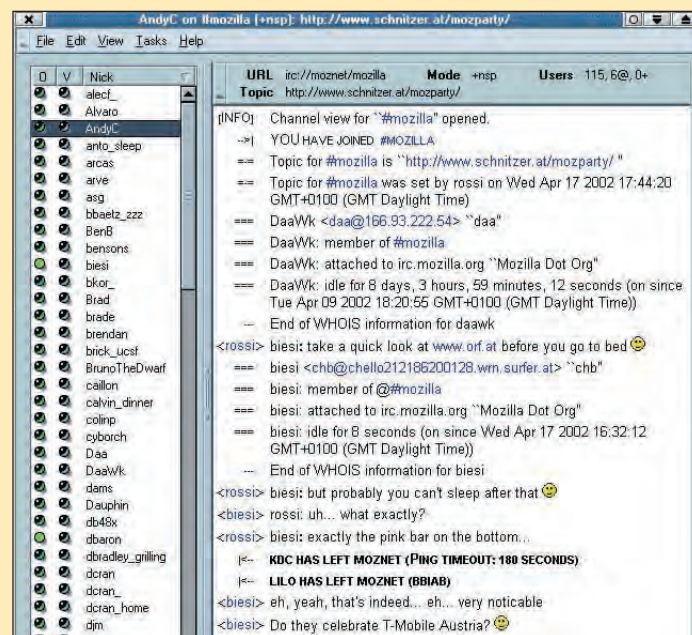
The earlier releases (especially on Windows) seemed to be the least developed of the core applications, but those days are long gone now. Did we mention it was pretty? And fast?

Composer Composer is Mozilla's web authoring application. It's basic but stable and is more than capable of pulling together the elements of your average website. Site management facilities are almost non-existent but it does have an integrated ftp tool that makes uploading your work a breeze. The layout is intuitive and, as it's based on a very standards compliant engine (and has a rather good HTML validation tool), you should be able to create sites for any browser without too much trouble. Dedicated coders, though, will find themselves hitting the limits fairly early on.

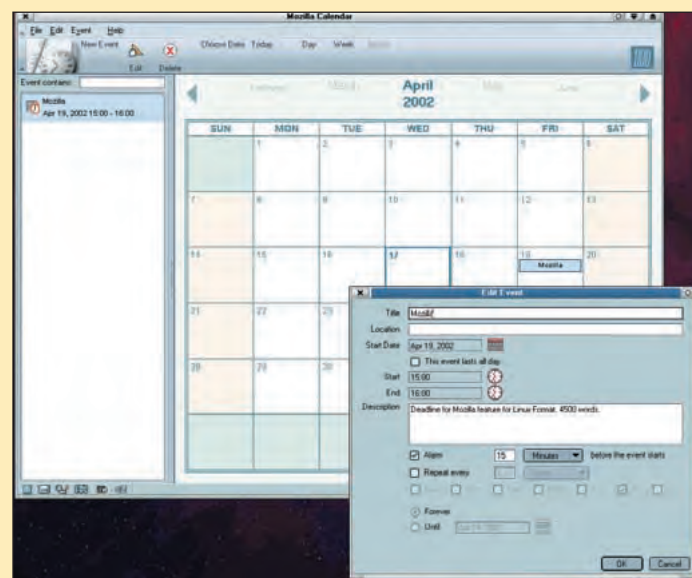
Address Book This does pretty much as you'd expect and can be configured to capture addresses from both incoming and outgoing mail. It also has tight integration with the mail client.

Chatzilla IRC has played a significant rôle in the development of Mozilla so it's unsurprising that the suite's IRC client is well designed and implemented. It's not quite as visual as some other chat apps, but it does the job.

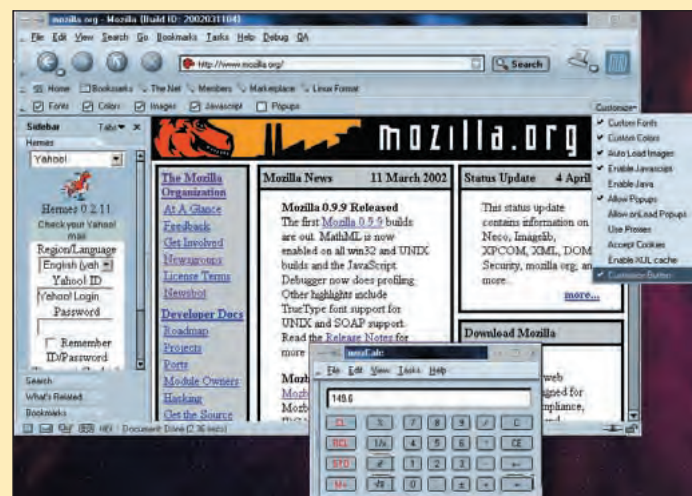
Calendar Though an Alpha version, the Calendar is a surprisingly well-developed bit of software. As with the address book, Mozilla's Calendar is not going to set the world afire, but it is an essential part of the suite, and is one less app you'll need to spend time looking for (unless it is



Chat to people across the world, from the discomfort of your office chair.

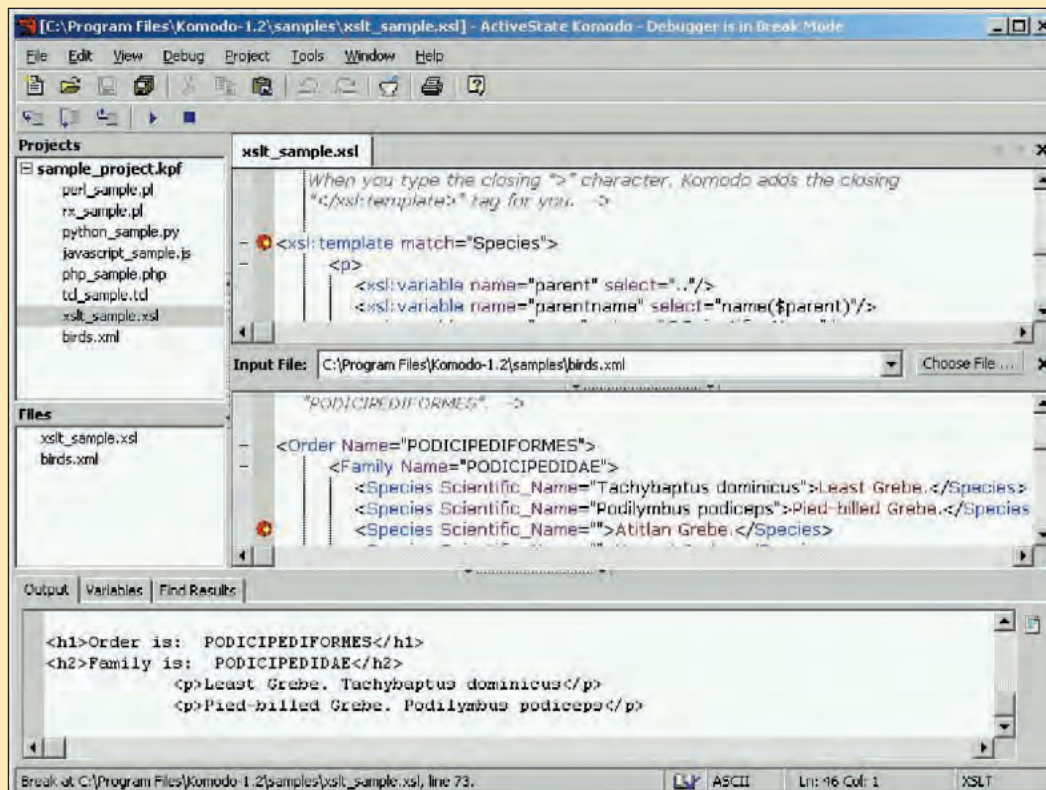


Calendar will officially join Mozilla clan sometime after the 1.0 release.



Hermes (webmail) and Mozcalc – demonstrating Mozilla's extensibility.

Mozilla



Komodo – Integrated Development Environment from Mozilla's XP technologies.



missing a feature that you need, like emailed alarms). You can pick it up at www.mozilla.org/projects/calendar/ – and it installs well via XPI.

As well as the main browser suite, innovate developers are coming up with extensions that speed up and improve browsing or just add an extra tool that is useful to have a click away. One of the best examples of the former is Aaron Anderson's Preferences Toolbar that allows you to manage popups, Javascript and cookies. You can get it from www.xulplanet.com.

A good example of a really useful extra is Martin Kutschker's *MozCalc*, which adds an XUL calculator to Mozilla's Tools menu. It has a full scientific feature set as well as pretty good integration with the UI. It's also available from xulplanet.

XUL and XPCOM

Mozilla's extensibility is based on two key technologies, XUL and XPCOM.

XUL (pronounced 'zool') is an XML-based user interface language designed to allow developers to create complex cross platform applications in a way that would be familiar to anyone with experience of Dynamic HTML (DHTML). However,

instead of presenting web content, XUL is used to position and define windows, buttons and all the other parts that make up the application's user interface, which means the UI can take advantage of the swift Gecko rendering engine to draw screen elements.

At its most basic, XUL allows you to take the fundamentals of the Mozilla suite, such as the sidebar, buttons and 'throbber' and brand or localise it for your customers and/or staff. It is capable of far more than this though and is a key component in making the porting of applications across platforms almost trivial.

In his very readable *Joy of XUL* paper, Peter Bojanic runs through the process of porting the Calendar application from its Linux origins to Windows and Mac, which was something that had to be done if it is to become an official component.

"The UI for the XUL application required zero porting effort because it worked with almost no alteration on both new platforms. Similarly, since the UI code is written in JavaScript, the interaction logic worked with no effort"

The only sticking point was the *libical* library, which, because it's written in C, required a little more work. The Macintosh version was up and running "within a few days" thanks to a native *libical* library. However there was no existing Win32 port available, so developers had to undertake a porting effort to migrate *libical* to Windows. "Within a week this too was complete and Mozilla had a working calendar for all three primary platforms."

XPCOM (Cross Platform Component Object Model) – and XPconnect for simple interoperability between XPCOM and JavaScript – complements XUL by enabling the integration of external libraries into your applications. XPCOM components can be written in C, C++, and JavaScript, and will run on 'just about any platform that hosts a decent C++ compiler.'

Perfecting these components has taken time and has been the source of much ire, but according to *Galeon* developer Philip Langdale it's been time well spent, ensuring that Mozilla is capable of running on dozens of platforms. "Without the XP infrastructure," he says, "it would probably have supported three platforms at most"

An in-depth discussion of XUL and XPCOM is a bit beyond the scope of this feature, but check out the web links on the Mozilla projects page (www.mozilla.org/projects/).

Mozilla inside

The following three projects demonstrate the flexibility of the Mozilla code and licensing structure. They range from a superficially simple browser released under the GPL

"The UI for the XUL app required zero porting effort because it worked with almost no alteration on all platforms."

via a commercial programming environment to a complete home entertainment centre built on Linux and Mozilla.

Komodo

One of the more ambitious Mozilla-based projects we've encountered is *Komodo* from ActiveState.

This complete Linux and Windows Integrated Development Environment (IDE) built using XP technologies really shows the extent to which you can remake Mozilla. The most obvious thing about the application is that it looks nothing like Mozilla, thanks to a fairly radical XUL and Javascript overhaul. "We use XPCOM extensively, and leverage the NSPR (Netscape Portable Runtime) as well," says David Ascher, Technical Lead on the project.

But this is not just an interesting product; it also demonstrates the potential for marrying open source and proprietary licensing. Though this is often considered a legal minefield for mainstream developers, Ascher said working within the Mozilla Public License (MPL) and GPL had not caused any problems. "ActiveState has a long history of working with both open source and proprietary code," he said.

As long running users and advocates of open source technology, ActiveState have been major contributors to Mozilla, especially the component object model (XPCOM) and the plugin interfaces. "An ActiveState developer wrote the Python bindings for XPCOM, which allow the creation and use of any XPCOM component from the Python programming language. We donated the PyXPCOM code to the Mozilla tree, making Mozilla stronger. We've also been involved in several aspects of the plugin API, but that was mostly to scratch our own itch."

Komodo has been optimised for open source programming and includes an advanced language-aware editor, graphical debugger, distributed development support, CGI emulation and a powerful regular expression toolkit for Perl. Add extensive XSLT abilities, support for over 24 languages, including Perl, Python, PHP, Ruby, Tcl, XML, XSLT and the UI options provided by XUL and

you have an environment that should handle anything you can throw at it.

Galeon

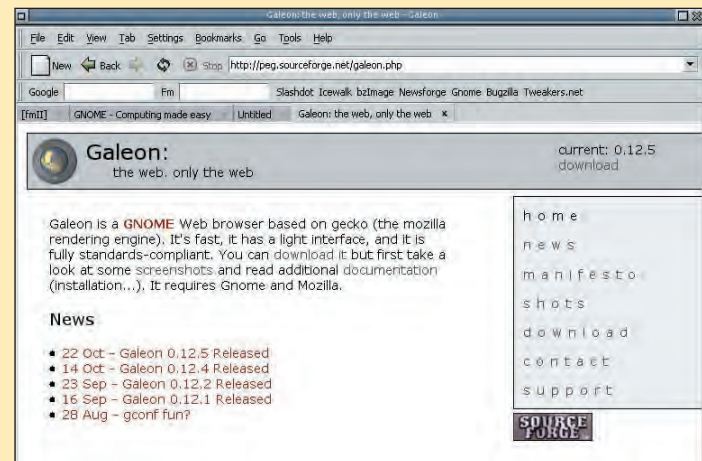
Galeon is a Linux/GNOME specific web browser that has followed Mozilla releases fairly closely, and it has been called the product that Mozilla could – and perhaps should – have been. This is not a view shared by *Galeon*'s developers. "The fact that mozilla.org has produced a top quality, fully functional product is undeniable," says Philip Langdale.

While Mitchell Baker and her fellow hackers were concentrating on the infrastructure technologies, *Galeon* was able to take their code and make platform specific optimisations, such as re-implementing the prompt dialog service to use *GNOME* dialogs and replacing the download handler with one that interfaces with the *GNOME* MIME database to retrieve information on helper apps. And *Galeon* takes more Mozilla code that most of us realise.

"A lot of people think we just embed *Gecko* and nothing more. In reality, we try to take advantage of as much Mozilla functionality as we can to avoid reinventing the wheel," says Langdale. "As a component system, it is well suited to embedders like us; we can easily use a lot of the services like Cookie and Password management and we can also drop in our own code to override certain main functions when we need to." These optimisations could be merged back into the main project – though the team's main contributions so far have been in the form of bug fixes. "We haven't committed anything big yet," Langdale says. "However, there is code in *Galeon* – our external protocol handler which passes unrecognised protocols to the *GNOME* URL handling mechanism is one – that can be run without *Galeon*."

"We've also developed a help documentation component. I added support for the displaying of man/info/gnome-doc documents in *Galeon* but it's all implemented as Mozilla components and this code will just drop into Mozilla and work (we tried)."

Galeon is currently at version 1.2.0 which requires Mozilla 0.9.9 and



Galeon – the GNOME browser embeds Gecko and much more.

Mozilla and Netscape

A complex inter-relationship



Netscape and Mozilla are often seen as interchangeable, but in fact the relationship between the two distinct entities is a complex one – and it's not without its dynamic tensions. Adding the enormous media multinational AOL/Time Warner into the equation doesn't simplify matters, but it helps if you repeat the mantra (and this one must be popular at mozilla.org) "Mozilla is not Netscape." However, it would be easy to see why anyone would think they were one and the same:

- Most of the developers working full-time on Mozilla are – or were – paid by Netscape/AOL/Time Warner.

- *Netscape 6.x* is the biggest project based on the Mozilla core and looks extremely similar to its developmental cousin (albeit overflowing with Netscape branding).

- The whole project rose phoenix-like from the ashes of Netscape's "air-supply" free version 5 web browser.

There are distinctions, though, and these have important – you may even say life-saving – implications.

- Not everyone who works for Mozilla works for Netscape. In fact, Mitchell

Baker told us Mozilla had attracted an 'astonishing range of participants'. "The number of key engineers who came to Mozilla through volunteer efforts is quite high, and many of these have since been hired by companies to work on the project, so they don't remain unpaid volunteers indefinitely." Mozilla has served, she said, as a very effective hiring tool, since companies can now hire people whose code and work style are already familiar to them.

- You can start your own Netscape rival if you fancy – it doesn't take a great deal of effort to rebrand the browser.

Netscape Communications has been transformed from vanguard of the Internet boom to browser also-ran, from corporate giant to AOL/TW department; but regardless of the changes 'at the top', Mozilla's open source status has ensured that the project survives and thrives. And with CompuServe's latest software dropping Internet Explorer in favour of a Mozilla solution (and a similar Beta product from AOL), we could soon see a little more competition in the browser market.

works like a dream, but the development team are still enthusiastic about the 1.0 release "if for no other reason than that we hope the API will stabilise long enough for Mozilla to make two releases without requiring any code changes in *Galeon*!"

Does he think Mozilla's success will lead to more people investigating open source software development? "Mozilla is a pretty enormous project, and its scale is probably not



Mozilla



something that others would attempt to copy," he says. "The final touch would be AOL adopting Mozilla for their new software. I think that sort of success would encourage firms to investigate open source development, but the issue is less the effectiveness of the development process and more the extent to which it fits with the company's product and their goals."

The team is currently working on *Galeon2* for *GNOME2*. "Marco [Pesenti Gritti, project originator] has completely redesigned the way *Galeon* works inside, taking advantage of new features in the *GNOME2* libraries. We're currently waiting for the updated *gtk2* version of Mozilla – particularly the *gtk2* embedding

widget. Until then, we can't actually use *Galeon2* as a browser!"

OEone HomeBase

While *Komodo* and *Galeon* represent two very different aspects of the development spectrum, there is one project that takes Mozilla on Linux to the stage that was envisaged by Netscape executives back in 1998: a system equally at home composing letters or emails, browsing the web, watching video or managing your MP3 collection.

HomeBase is a hardware/software Internet Appliance along the lines of Larry Ellison's Internet Computer, though with a home entertainment bias. It is based on a custom Red Hat/2.4.17 core with Mozilla providing

access to a range of services and applications including an integrated word processor and 100MB of online storage for either back up or remote access purposes. Of course, as it's based on Linux, HomeBase can also run everything from *Apache* to *StarOffice*.

The company are currently selling (to US and Canadian buys only at present) three configurations: an (old fashioned) iMac-style PC called AIO, a monitor-free mini tower called Terminator and a software only package called *HomeBase in a Box*. The Terminator package is available with or without a TV tuner.

Have we mentioned the UI? It appears that OEone are attempting to move away from the WIMP idea to something more like the interface you'd find on a cable or satellite STB. While this simplifies operating (and demonstrates the extraordinary abilities of Mozilla), it does mask a lot of the potential and complexity of Linux – which may just be the point. Timothy Butler in a review of HomeBase for the open source migration portal Open for Business enthused "finally someone dared to try something different, and came up with a new UI concept that actually works."

What's more, OEone and its HomeBase developers are completely committed to open source, Mozilla and Linux, and have contributed significantly to the multimedia and productivity capabilities of Mozilla. Peter Bojanic, the company's Vice President of Software Development, said OEone's principle involvement with Mozilla had been in the development of *Penzill* and the Calendar, but it had also been instrumental in bringing a number of technologies to Mozilla, even though these wouldn't be integrated into the Mozilla core.

"OEone has developed and contributed a number of projects to the Open Source community," he says. "These include *MozXine*, a plugin that integrates the *Xine* multimedia framework into Mozilla, enabling developers to play audio and video directly in a Mozilla application; *MozStreamer*, a plugin for the GStreamer multimedia framework; and *AbiLib*, an XPCOM component

List of other projects in development

Making Mozilla a complete OS

Mozblog

Mozblog lets Mozilla users jump up on the Weblog bandwagon with the minimum fuss.

Jabberzilla

Multi-standard instant messaging based on cross-platform technology. Works with AIM (which is included in the Netscape branded Mozilla suite), MS Messenger and ICQ.

<http://jabberzilla.mozdev.org/>

Hermes

This is a project to allow easy access to webmail services such as Yahoo! and Hotmail from the Sidebar. The current release slows up the main browser a little, but it's not unworkable.

<http://hermes.mozdev.org/>

Amoeba

This is an unusual one: a Mozilla based engine for designing and distributing rôle-playing games. The first games

built using the engine follow the Zelda design principle – top down exploration, *et al.* – and developer Andrew Wooldridge has a rather healthy attachment to fruit. His main aim though is to create classic games.

"These are lofty goals, but with Mozilla being so powerful, and XML being the language of choice – I'm sure I can succeed in this." He also wants your help.

<http://amoeba.mozdev.org/index.html>

Meow

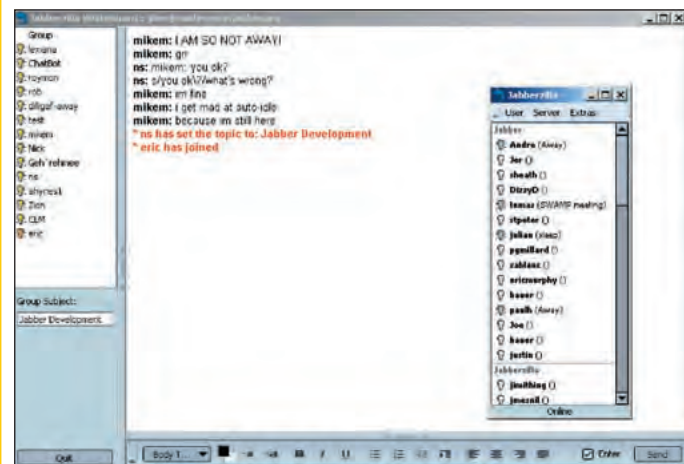
Meow is a cross-platform file manager/explorer, designed with the intention of homogenising users access to both web and local files. It's very early days for this project at the moment, but it should be an interesting experiment.

<http://meow.mozdev.org/index.html>

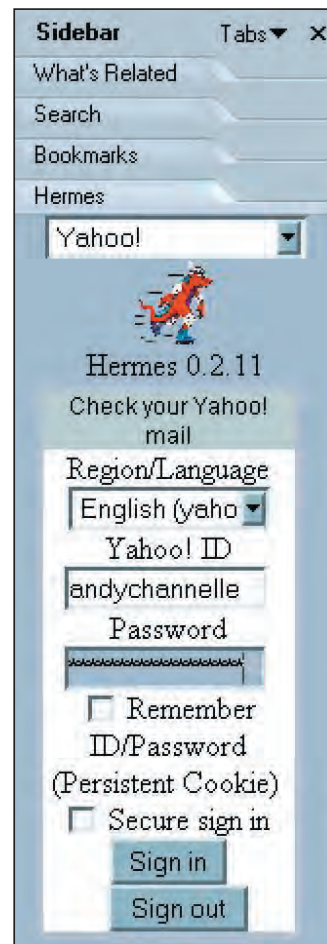
For an exhaustive list of Mozilla

projects including games and media players, visit

<http://www.mozdev.org/projects.html>



Jabberzilla – multi-standard instant messaging.



Hermes – providing easy access to your webmail.

Acronym corner

Drowning in alphabet soup

The Mozilla project is an acronym magnet, so if you have letters swimming before your eyes, here's a primer.

XUL – XML User Interface Language.

The language Mozilla uses to configure UI elements and applications. Eases creation of cross-platform applications.

XPCOM – Cross-Platform Component Object Model. Technology to add external libraries into XUL applications. XPCOM is based on the COM technology that was born at Digital and raised at Microsoft. On Windows XPCOM interfaces are wrapped in an ActiveX control that Visual Basic developers can use.

XML – Extensible Markup Language. XML is a cross-platform, Unicode

compliant markup language designed for presenting structured data.

It is an 'official' web standard governed by the World Wide Web Consortium (W3C). Anyone familiar with HTML would recognise an XML file, though XML is far less forgiving of bad code.

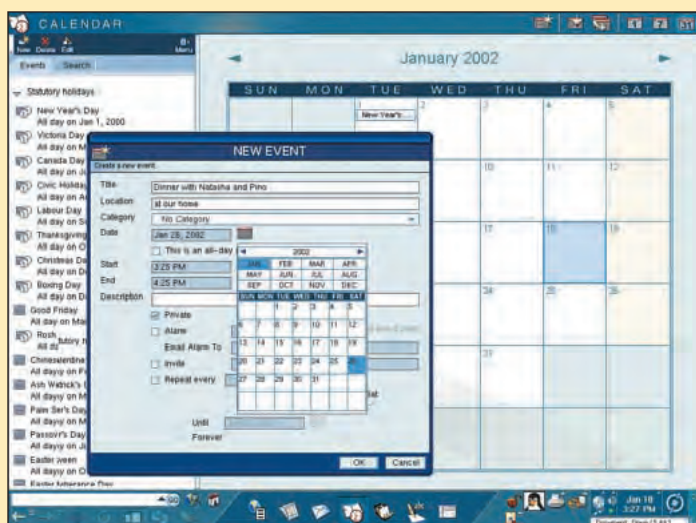
CSS – Cascading Style Sheets.

CSS is a language that allows developers to configure 'styles' (fonts, spacing, and aural cues) HTML documents and XML applications. By removing the presentation aspect from the content of documents, Cascading Style Sheets simplify web authoring, site maintenance and now application building.

DOM – Document Object Model. This is a "platform- and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure and style of documents." It provides a standard set of objects for representing HTML and XML documents, a combinatorial model, and an interface for accessing and manipulating them.

RDF – Resource Description Framework.

RDF is another W3C control language. It has been formulated especially for representing metadata such as author, title and changelog for a web page. It provides a common framework for cross-platform expression of metadata without loss of meaning.



The well developed Calendar belies its alpha tag.

that integrates the *Abiword* word processor into Mozilla."

You're my number 1

So, are there any lessons the average open source developer can learn from the Mozilla experience? Mitchell Baker concedes the big mistake in this project was a lack of separation between the development efforts for the toolkit technologies and the applications that use them. "We have done both simultaneously, and this causes challenges," she says. The real strength of open development has been demonstrated in the community of enthusiastic users and developers that has grown around Mozilla. "A phenomenal testing and QA program can be developed for applications that have a user interface and functionally

people care about." The community QA program has been enormously successful with download numbers for each of the last milestones exceeding 200,000. The 0.9.9 milestone was the most popular yet receiving 150,000 downloads in the first 48 hours it was on the wire. Baker says each of these users can make a useful contribution to the project. "The feedback and crash data from this is critically important to the project. We also have an active community QA group working on our daily milestones, providing bug reports, categorisation, prioritisation, test suites, verifications, etc."

David Ascher of ActiveState says that as in any very large project, some things could have been handled better, but that shouldn't take

anything away from the team's achievement. "It's an impressive project, which deserves congratulations on reaching 1.0"

As Mozilla reaches its first official release, it also realises one of the project's first ambitions: decentralised development. From every corner of the world, bugs have been reported and squashed, features have been proposed and added and real users – the enthusiastic testers of a seemingly endless stream of 0.x releases – now have a real choice. The last word goes to Mitchell Baker. "One of biggest lessons we can learn from the Mozilla project is that people want an alternative and are willing to help make it happen." [LXF](#)

The nature of the beast

From Mosaic-killer to office suite

Mozilla: The Name

Mozilla was the codename for the nascent *Navigator/Communicator* project and is a contraction of 'Mosaic Killer' – *Mosaic* being the pioneering graphical web browser that Netscape superseded.

Mozilla: The Beast

Designer Dave Titus created Netscape's official mascot in 1994. The icon was updated with the launch of *mozilla.org*, turning from green to red and looking far fiercer; however, the original monster still lives on in the opening splash screens on some versions of the application. www.davetitus.com

Mozilla: The Project

The Mozilla browser is based on code open sourced by Netscape in 1998, though after a policy change in October of that year, it is thought that only about 5-10 per cent of the original code remains. Mozilla is actually an amalgamation of the program destined to become *Netscape 5* (re-christened *MozillaClassic*) and another project called *NGLayout* (which is more commonly known as *Gecko*). *NGLayout*'s cross-platform codebase means that Mozilla could form the basis of a new, OS-independent middleware base running anything from a stripped-down browser, such as *Galeon*, to a complete office suite.

Working with Macintoshes



Charlie Stross takes a look at how Linux interoperates with Macintosh systems — including a tour of Appletalk file and print services, and a study of filesystems.

“The Mac system has evolved at the margins of the business world, in consequence Linux compatibility is very good.”

There are two superpowers in the personal computing world: Microsoft and Apple. Once upon a time, Apple's early computers had the biggest share of the market; since then, the Mac field has steadily shrunk, and now occupies a niche status with graphic designers, video editors, and the like. However, Apple's core user base is extremely loyal — and in one very important respect they're the best allies the Linux world has got.

Microsoft products aren't good at coexisting with other platforms, but the Macintosh system has, like Linux, been forced to evolve at the margins of the business world. Consequently, Macintosh/Linux interoperability is as good as, if not better than, Windows/Linux compatibility.

Linux can act as a file and print server for Macintosh networks. It can act as a gateway for Macs accessing the Internet. It can read HFS and UFS (but not HFS+ — yet) filesystems. Using *mkisofs*, Linux can build Mac-compatible CDROM images. Linux machines can't easily act as clients for Macintosh servers, but there are some tricks you can do to make it work. And at a pinch, Linux (on a PowerPC machine) can run MacOS as a guest operating system.

In this feature we're going to cover all the bases — starting with network protocols, file and print services, file formats (and little things like Macintosh type/creator attributes, which Linux doesn't understand by default), then directory services

and a brief Linux hitch-hiker's guide to MacOS X.

Network protocols

You can use a Linux system as a file and printer server for Windows PCs via *Samba*. But less people are aware that you can do exactly the same thing for Macintoshes, using a couple of different packages.

Windows networking uses the NetBEUI protocol stack, a relic of the 1980's acquisition of a small networking company by Microsoft. Apple began developing the Macintosh architecture around 1980 (originally in the form of the Lisa project), and, as with Windows, the standards for networking adopted by Apple predate TCP/IP — in those days, TCP/IP (internet) networking was only used by workstations and weird military and academic systems.

AppleTalk was originally intended for transmission over serial cables, but proved compatible with ethernet. More recently, Apple have bitten the bullet and adopted TCP/IP — and in particular, have acquired the ability to tunnel AppleTalk packets wrapped up inside TCP/IP packets.

AppleTalk is a set of protocols analogous to the internet protocols. They include RTMP (routing table maintenance protocol), AEP (appletalk echo protocol), NBP (name binding protocol), ZIP (zone information protocol), and the higher level AFP (Apple filing protocol) and PAP (printer access protocol) which concern us. All of these are layered on

top of the Datagram Delivery Protocol (DDP) which replaces IP (the Internet Protocol) for transport (at least, unless you're dealing with AppleTalk protocols tunnelled over IP).

There are several AppleTalk implementations for Unix and Linux, but the most important are CAP (Columbia AppleTalk) and *Netatalk*. *Netatalk* is the more recent development and comes as standard with most Linux distributions; it's generally easier to configure and use than CAP.

On SuSE and Red Hat machines (and others), the *Netatalk* configuration files are stored in */etc/atalk*. There are several components to *Netatalk* — starting with a kernel module for handling the Appletalk protocol stack, the user-space *atalkd* AppleTalk daemon (which handles the various routing, name registration, zone lookup and ping protocols by binding to an ethernet adapter and encoding/decoding packets), and the *papd* and *afpd* (printer and file service) daemons that provide AppleTalk equivalents to the *lpr* and NFS protocols. If you're going to use *Netatalk* to provide file and print services, the usual procedure is to load the *appletalk.o* kernel module, then run *atalkd*, then start *afpd* and *papd* to provide services over AppleTalk. This is usually executed at startup (on SuSE, via the */etc/init.d/atalk* rc script), and sensible default options for a small LAN are installed by most distributions. By

default, *atalkd* assumes that you're dealing with a single AppleTalk zone – although you can configure it to act as a bridge between zones (if you have a couple of ethernet interfaces).

Printer configuration

Configuring a printer is relatively straightforward. First of all, you need to set up the *lpd* print spooler on Linux. Then you set up *papd*. *Papd*'s job is to accept AppleTalk print requests and dump the print jobs into the local *lpd* spool directory. *Papd* assumes that the printer you're using is a postscript printer (and may need a PPD – Postscript Printer Definition file – to set up parameters like sheet feeders and page sizes), but works acceptably if you tell *lpd* to invoke *Ghostscript* as an output filter and send the resulting bitmap to a non-postscript printer. Setting up a print queue goes beyond our scope here, but in the case of Red Hat or SuSE you can simply use the Red Hat control panel or SuSE's *YaST1* configuration tool to define a print queue and tell the system which driver to use. (Note that SuSE 7.2 and on tries to use *CUPS*, the *Common Unix Printing System*, by default instead of *lpr* or *lprng*; if you're going to provide print services to Macintoshes you probably want to manually select *lpr* and use *YaST1* – or hand-edit */etc/printcap* – to get it working.)

If the print queue *lp1* is attached to a local postscript printer (or a print queue emulating a postscript printer), we can tell *papd* to use it by adding to */etc/atalk/papd.conf*, the following:

Linux Printer:\

:pr=/usr/bin/lpr -Plp:\

:pd=/usr/share/lib/ppd/

HPLJ_4M.PPD:

The **pd** flag here indicates that we're going to use a PPD file stored in */usr/share/lib/ppd/HPLJ_4M.PPD* to set up the printer, which will be visible to Macs on our network under the name "Linux Printer". Anything sent to "Linux Printer" by a Mac will have the HPLJ_4M PPD file attached and get spooled to the Linux printer known as **lp** (defined in */etc/printcap*).

File server

File services via *afpd* (the AppleTalk Filing Protocol daemon) are harder to set up because there's more to do.

You need to specify which filesystems to export, how to authenticate remote users (indicating which volumes they're allowed to mount), and then how metainformation describing Mac files is to be stored on the server.

This latter point is an alien concept to those of us who've grown up using PC's or Unix systems. On Windows or Linux a file is a stream of undifferentiated bytes, distinguished only by its filename and any permission bits or attributes that are tagged on to it. The name, permissions, and other attributes are known as "metainformation" because they contain info describing a file, rather than being part of its contents.

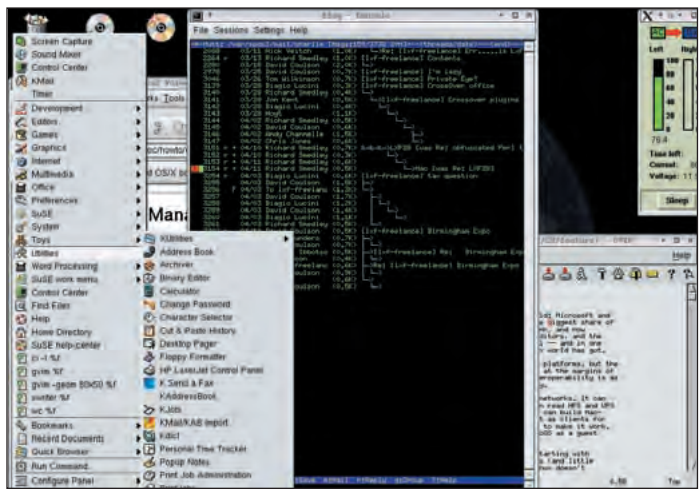
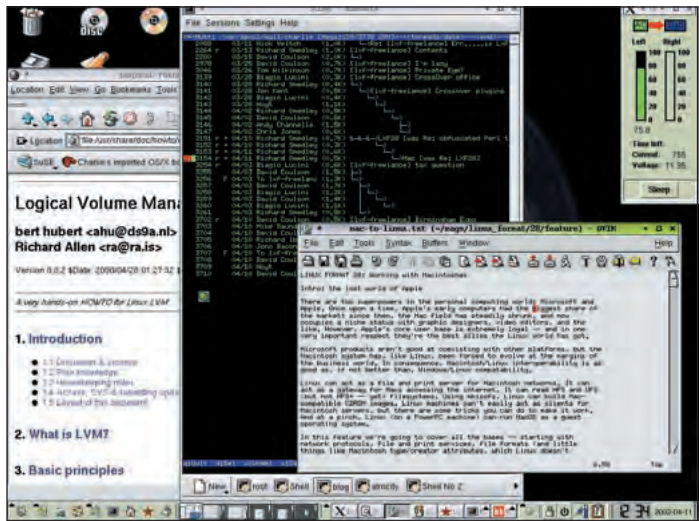
Since the early days of computing, we've had to deal with different types of file. For example, executable programs often store their data in different – incompatible – formats. Windows deals with this by assigning a three character suffix to different types of file (for example, .doc for *Word* documents, .xls for *Excel* spreadsheets, .txt for text files, and so on). Unix and Linux traditionally simply asserted that all applications had to be able to handle anything – but this broke down some time since, especially with the advent of email. We transmit files between computers via email and web protocols using MIME, a convention which assigns extra identifying metainformation to a file. When you MIME-encode a file (by, for example, sending an email attachment) you brand it with two pieces of information – the name of its creator application, and the type of this file (for example, creator: Word, type: RTF or Document or Template). Some Linux based systems have explicitly adopted MIME types; for example, KDE works out the MIME type of a file using a lookup table of file suffixes and MIME types, and this governs how the file is opened when you click on it. Windows works in an analogous way, as does MacOS X (which, like KDE, is a graphical environment running on top of a Unix system).

What most people forget is that Apple invented the type/creator metainformation concept, and handle it differently (and you need to know how Macintoshes store this information before you can set up an

AppleTalk file server).

The standard Apple filesystem is HFS, the Hierarchical File System. HFS differs from a Unix or DOS filesystem in that files are stored in two "forks" – the data fork (which resembles the traditional stream-of-bytes file) and the resource fork. The resource fork is a B-tree – it can contain extras such as icons, chunks of executable code, configuration variables, the file's type (a 4-character code) and creator (another 4-character code) flags, and so on. The resource fork in fact behaves rather like the Windows registry – except that there's a separate one for every file.

SuSE Linux 7.3 PPC on an iBook. A terminal window is open, running the *mutt* mail client; there's also a *Vim* session and *Konqueror* in the background, and *batmon*.



Obviously, if you serve up a Linux filesystem to a Macintosh, the Macintosh needs to be able to see resource forks – but files on Unix or Linux don't have them. Likewise, if a Macintosh saves a file to a Linux server via AppleTalk, the server needs to be able to keep the resource and data forks separate. So *Netatalk*

There's even a port of *OpenOffice* to PowerPC Linux, seen here running in the background. All the comforts of home...



Mac Interoperability



MOL (Mac-on-Linux) is a virtual machine environment for LinuxPPC that serves the same rôle as *VMWare* on Intel. Here we see a *MOL* session running under *XT11*, booting into *MacOS 9.2* (*MacOS X* is not yet supported).



creates an entire parallel directory tree (on each exported tree of directories) to store the resource metainformation. Each directory that the *afpd* daemon serves up gets a *.AppleDouble* directory created inside it. And each file that has a resource fork – usually created by a Mac client – has a shadow file in the *.AppleDouble* directory, with the same name and containing the resource information.

Configuring *afpd*

Afpd has three configuration files, usually held in */etc/atalk* on Linux systems. These are *afpd.conf* (which tells the server how to operate as a network daemon and where the other files are located), *AppleVolumes.default* (which specifies which directory trees are exported as Macintosh volumes), and *AppleVolumes.system* (which sets up the default mapping between filename suffix and Macintosh type/creator info).

The options to *afpd.conf* are documented in a man page; mostly they centre on whether to use DDP or TCP as the transport network layer, where the other files are located, and how to authenticate users. *AppleShare* requires users to identify themselves by username/password before they

can mount a volume (a guest account “nobody” is often provided).

AppleVolumes.default sets up the volumes that are exported. It follows a syntactic model similar to the NFS */etc/exports* file, where each uncommented line specifies a directory, the name it will be exported under, who (or what machines) are allowed to mount it, and whether any conversions are to be applied (for example, to impose Windows filename conventions for compatibility with *Samba*, or to do automatic CRLF translation for files of type TEXT. Almost inevitably, Linux systems come configured to export a user’s home directory – that is, if user “joe” has an account and password on the Linux server, then when they connect to it from a Macintosh client and enter their Linux username and password they can see a volume called “joe” containing their home dir.

The *AppleVolumes.system* file is slightly more complex.

As noted earlier, Macintoshes don’t pay any attention to the filename suffix when working out what icon to display on a file or what program to use to open it when you double-click on it. That’s because they store type/creator bits in the file’s

resource fork. By default, if *afpd* exports a file with no resource fork and no known type and creator, it’ll give it the type “????” and creator ‘Unix’ – but this is no use to a Macintosh. Files that a Mac client saves onto an AppleTalk volume will have resource forks that tell the Mac what to do with them – but what about other files?

To solve this problem, the *AppleVolumes.system* file contains a series of records that look like this:

.	“TEXT”	“txt”
ASCII Text	SimpleText	
.mf	“TEXT”	“MF”
Metafont	Metafont	
.sty	“TEXT”	“TEX”
TeX Style	Textures	
.psd	“8BPS”	“8BIM”
PhotoShop Document	Photoshop	
.pxr	“PXR”	“8BIM”
Pixar Image	Photoshop	
.sea	“APPL”	“????”
Self-Extracting Archive		
Self Extracting Arc		

The first field is the file’s suffix; the second and third are the type and creator flags, while the fourth and fifth are human-readable descriptions. For example, we can see from this that dot-files are served up as TEXT files with creator *txt* (*SimpleText*, the standard Macintosh system text editor). You can edit TEXT files using any application that can open TEXT files (for example, *BBEdit* or *Alpha*), but if the creator application is present it will be used by default.

You can do several things with *AppleVolumes.system*’s mappings to customize the way your Linux server interacts with the Macintosh clients. For example, suppose you don’t like *SimpleText* but prefer to use *Alpha*. *Alpha* has a creator flag of ‘ALFA’, so if we replace ‘txt’ with ‘ALFA’ in all the records in this file, *Alpha* will be used to edit text files you double-click on instead of *SimpleText*.

It can be hard to work out what type/creator flags to use for new types of file – but a tool in the *Netatalk* kit helps us; *ICDumpSuffixMap*. This is a MacPerl script. MacOS 8.1 and higher uses a tool called *Internet Config* to assign type/creator mappings to files received as MIME attachments in email, and *ICDumpSuffixMap* eats the *Internet Config* database and spits it out in the format used by

AppleVolumes.system.

Not all Macintosh clients will have access to the same programs. That's okay; you can have subsidiary .AppleVolume.system files in each user's home dir. These override the master file, so if you want everyone to use *Alpha* to edit text files but Joe prefers to use *SimpleText*, Joe can override *Netatalk*'s global settings.

Managing netatalk

Netatalk – both the network protocols and the *papd* and *afpd* services – is started from an init script at boot time; */etc/rc.d/init.d/atalkd* (on Red Hat) or */etc/init.d/atalkd* (on SuSE). It should come configured out of the box to work with user home directory volumes; the only thing you may need to do before running it for the first time (on a recent distribution, at any rate) is to configure an *lpd* print queue pointing to either a Postscript printer or a *Ghostscript* driver, then put the name of the print queue in */etc/atalk/papd.conf* (as above).

In use, *afpd* has an annoying habit of spraying .AppleDouble directories all over your filesystems. To keep this under control, use the *acleandir* utility. *Acleandir* cleans up the directory specified as its parameter; give it the **-r** flag and it cleans up all that directory's children as well. By default it removes "orphan" AppleDouble files, which do not have a corresponding data file. However, if you add the **-d** flag it will also remove orphan .AppleDouble directories (i.e. ones where all the files have been removed), and the **-a** "aggressive" flag can be used to force the removal of all .AppleDouble files, not just orphans. You probably want to set up a *cron* job that runs *acleandir -rd* / once a day or once a week.

There are a couple of other vital utilities that come with the *Netatalk* package. *afile* tells you the Apple type and creator of a file without checking the AppleVolumes.default files – it looks in the .AppleDouble resource file if one is present. Thus, if something is persistently showing up as the wrong type of file, you can use *afile* to work out what the Macintosh client (as opposed to *afpd*) thinks it is, and if necessary update your AppleVolumes.default file.

Sometimes a file shows up with the wrong type and creator for no obvious reason, and needs changing. Use *achfile* to do this. For example, to specify that a file called "silly.txt" is actually a Microsoft Word RTF file:

```
achfile -t 'RTF' -c 'MSWD' silly.txt
```

Finally, you can tell how busy your server is by using the **nu** (*Netatalk* users) command to list current logged-in users, and **macusers** for some more verbose feedback.

Timelord

Timelord is a time server for Macintoshes. The corresponding client, *Tardis*, can be found in some of the MacOS free software archives; *Timelord* is distributed with *Netatalk*. You can probably consider *Timelord* to be obsolescent – MacOS X uses the standard Network Time Protocol to keep the system clocks in synch.

Burning CD's

Macintosh CDROMs show up with fancy icons and working type/creator attributes when you load them on a Mac. Linux can burn data CDROMs; you use a tool called *mkisofs* to master an image of an ISO-9660 filesystem, then use *cdwrite* to burn the image onto a CD. Not many people know that Mac CDs are basically standard ISO-9660 CDs with some extra info – like those vital type/creator attributes – saved in a special record on the disk, and even fewer people realise that *mkisofs* can create perfect Mac CDs which also work perfectly on Linux, and on Windows at the same time if you use Rock Ridge and Joliet extensions as well as MacOS extensions! But it's true – you can use Linux to take files your Macs have saved into a directory via *Netatalk* and burn them into CDs that Macs will recognise correctly.

Burning Macintosh HFS CDs under Linux rates an article as long as the whole of this one, but you can get some basic pointers from the *mkisofs* manual page. If you want to burn the contents of */home/image* into a CD image stored in */tmp*, you might normally issue a command like:

```
mkisofs -R -o /tmp/disk.img /home/image
```

(Where **-o** specifies the output filename and **-r** indicates Rock Ridge (Unix) file extensions are to be used).

To add Macintosh HFS extensions, and tell *mkisofs* to get the Macintosh type/creator associations from the *Netatalk* .AppleDouble directories:

```
mkisofs -R -hfs --netatalk -o /tmp/disk.img /home/image
```

To tell *mkisofs* to label the disk "Macintosh backup", so that this is what it's named when you insert it into a Mac:

```
mkisofs -R -hfs --netatalk -o /tmp/disk.img -hfs-volid 'Macintosh backup' /home/image
```

There are a number of options that *mkisofs* can use to scan for type/creator associations, but if you're using *Netatalk* this is the way to go. We can even add Joliet extensions so that the long filenames will show up on a Windows machine, if we're feeling merciful:

```
mkisofs -R -J -hfs --netatalk -o /tmp/disk.img -hfs-volid 'Macintosh backup' /home/image
```

Isn't *mkisofs* fun?

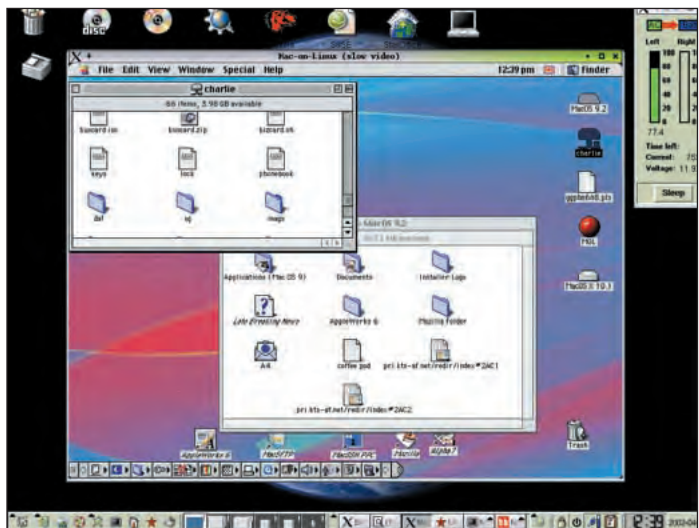
Accessing Mac files

Every operating system seems to come with a different filesystem and a different approach to storing data. Worse: every so often, the developers decide that their current solution is inadequate and bring out an upgrade.

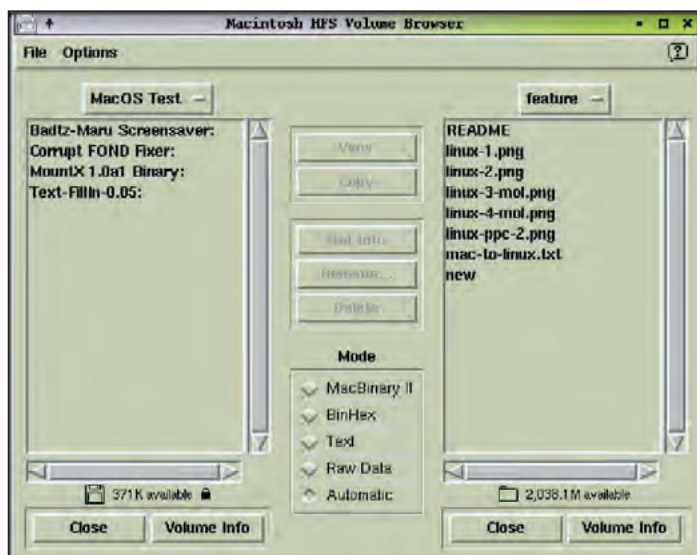
MacOS is no different. Original Macs used HFS, the Hierarchical File System, and this is well-supported on Linux. Subsequently, with System 8.1 Apple introduced HFS+, which supports larger hard disks and is more efficient. MacOS X in turn moves on to UFS, the Berkeley Unix Universal File System, which is also well-supported on Linux. But each

"You can use Linux to take files your Macs have saved via Netatalk and burn them onto CDs your Macs will recognise correctly."

MOL running with two folders open – one on the MacOS HFS+ partition that MOL booted from, the other on a Linux ext2 filesystem called "charlie", mounted via Netatalk from the Linux side of the system.



Mac Interoperability



Here's *xhfs*, a file manager in the tradition of *Midnight Commander*, that lets you browse HFS filesystems without mounting them. Use it to copy files to and from Mac floppies interactively.

system has its pros and cons, and different issues associated with getting files on and off it.

Firstly, Macintosh partition tables are not the same as PC hard disk partition tables. Luckily, Apple partition tables are easier to deal with; on PowerPC Linux you use the *pdisk* utility (instead of *fdisk*) to view or edit the partition table, and Apple's *HD Setup* tool (the MacOS hard disk partitioner) actually understands Linux *ext2* and swap partition types – the result of Apple's flirtation with MkLinux prior to MacOS X. Linux kernel 2.4 understands Macintosh partition tables by default, so you should be able to plug an external hard disk from a Mac into a PC running Linux and at least see what partitions are installed.

HFS provides filenames up to 31 characters long, type/creator meta-information, and aliases (symbolic links); it's more like a Unix filesystem than the DOS FAT filesystems used by Microsoft. However, HFS uses a B*Tree to store its data, and has certain limitations. On disks larger than 2GB, the minimum block size is 64KB – a 5-byte file will occupy an entire single block, or 64KB! This makes HFS extremely inefficient for storing lots of small files (which Unix-type systems like Linux and MacOS X have to do). In addition, HFS is very inefficient at handling directories containing hundreds or thousands of files.

Linux has supported HFS via a kernel module since release 2.2. To mount an HFS filesystem from, say,

`/dev/hda4` on a dir like `/mnt/hfs`, add the following to your `/etc/fstab` file:

```
/dev/hda4    /mnt/hfs    hfs
rw,noauto,user    0 0
```

Then type `mount /mnt/hfs`. Lo, the HFS filesystem will mount happily and you can read and write to it.

Before HFS support at the filesystem level, Linux users could use a set of tools called the *HFS utilities* to read and write files to HFS disks, including floppies. The *hfsutils* worked like the venerable DOS *mttools* – a set of standard commands were provided (named with the *h* prefix), such as *hls*, *hcop*, *hmkdir*, *hrm*, *hcd*, and so on. To start using them, type `hmount /dev/hda4` (or whatever device name corresponds to the filesystem you want to work with), then use the 'h' commands to move around the HFS filesystem, copy files to or from it, and so on. There's also a graphical shell called *xhfs* (screenshot, left) that lets you view the contents directly and copy files back and forth.

HFS+ is a newer version of HFS, with support for larger filesystems, finer block sizes, and larger numbers of files. It's standard on most newer Macintoshes.

HFS+ support for Linux is embryonic at the time of writing. While a filesystem driver will probably appear in the 2.5 kernel tree at some point, currently the only reliable way of manipulating files under HFS+ is to use the *HFSPlus tools* (available from <http://jridgell.org/programs/> among other places). These work like the *HFS utilities*, only the commands are prefixed with "hp" – for example, you use `hpmount` to start working with an HFS+ filesystem, then `hpcd` to change directory, `hpcp` to copy files on and off it, and so on.

Nobody has yet ported the *xhfs* shell to run with *HFSPlus tools*; however Jonathan Riddell has built a KDE IOSlave that allows the *Konqueror* file manager to act as a front end for HFS+ filesystems. (Confession time: I have not yet gotten this to work. However, if/when it does, this will be the best way short of full in-kernel HFS+ support of getting at your files on an HFS+ partition.)

MacOS X is a radical departure for Apple, and is probably best described as a descendant of BSD

Unix running on top of a Mach microkernel, with a non-X11-based graphical user interface and a classic MacOS emulation environment to allow for backward compatability. MacOS X is usually installed on HFS+, but you can also install it on UFS filesystems – Apple's only caveats are that this isn't stable enough to be fully supported as yet, and that the Classic emulator won't run properly if you try to use UFS. UFS filesystems are, however, fully supported by Linux. Just mount the filesystem with type 'ufs' instead of 'ext2' or 'hfs'.

Dinking with files

Once you've got at a Mac filesystem, you may still have problems dealing with files. The first thing to note is that Macintosh text files use LF (ASCII 13) to denote a newline, whereas Linux and Unixes in general use CR (ASCII 10) for that purpose. (DOS/Windows uses CR-LF pairs, and MacOS X really sets the cat among the pigeons by reverting to Unix orthodoxy.)

You can convert between Mac and Unix text files easily enough using Perl – see this month's Perl column for details – or with *recode* but what about other files?

As we've seen, Macintosh files have two forks. When we copy them onto a Linux system we risk messing them up. To deal with this issue, a number of encoding mechanisms have been developed. AppleSingle, AppleDouble, and MacBinary are all commonly-used binary encodings that keep resource and data forks separate – in the case of MacBinary both forks are stored in a single file, while AppleDouble stores the resource fork separately. For transfer via email, BinHex fulfils the same function as UUencoding or MIME Base64 encoding; it converts a Macintosh file into a simple ASCII format that can be reconstituted at the other end. Linux systems often use *megatron* to convert between these formats (and *Netatalk's* separate storage format).

Megatron is a tool with multiple front-ends to convert different files; what it does depends on whether it is invoked as **unbin**, **unhex**, **unsingle**, **hqx2bin**, **single2bin**, or **macbinary**. (When you install it, each of these

names is provided as a link to the common executable.) It's distributed by default with *Netatalk* these days, and you'll need it if you receive any .hqx suffix attachments (BinHexed files) in your email.

MacOS, like Linux and Windows, has a common file archive format – but it's not tar or zip. *Stuffit* archives are created using *Stuffit Deluxe* or *Stuffit Lite* from Aladdin Systems, a commercial software vendor (see www.stuffit.com). There is a free-as-in-beer Linux *Stuffit* decoder that allows you to unpack *stuffit* archives (suffix: .sit or .sea) on Linux for Intel; if you want to unstuff on Linux for PowerPC you'll probably end up having to use the free MacOS *Stuffit Expander* utility under *MOL* (see below).

MOL – Mac On Linux

If you have a Macintosh running PowerPC Linux, you may still need to run MacOS applications from time to time. The answer to your prayers is a tool called *MOL*, *Mac-on-Linux*. *MOL*, from www.maconlinux.org, is a GPL'd package that allows your Macintosh to fire up a virtual machine environment and boot MacOS in it, without having to do a complete reboot. On machines with New World ROMs and Open Firmware (read: less than two or three years old), you don't even need to make a copy of the ROM image. Just make sure you've got an HFS or HFS+ partition with MacOS 9.2.2 (or earlier) installed in it, and configure *MOL* by editing the `/etc/molrc` file. You will need to follow the detailed configuration instructions on the website, and you should make sure your MacOS partition is able to boot before you start in on *MOL*, or you're likely to have a frustrating time; there are lots of options to configure and the documentation is about what you'd expect for a release 0.9 product. In addition, *MOL* can't load MacOS X yet (although work is allegedly in progress on this front).

MOL can run in full-screen mode, using a different virtual terminal from X11, or you can run it in a window under X11 (although graphics performance is substantially slower if you do it that way). Sound support is available by way of a special driver, although it doesn't yet work on late-model iBooks. *MOL* can share an

ethernet device with Linux via the *sheep* driver; this lets you assign an IP address to *MOL* and use your Linux server as a router for TCP/IP connections from MacOS applications running under *MOL*. *MOL* can even mount Macintosh volumes exported via AppleShare by a copy of *Netatalk* running on the host Linux system (and in fact, this is the easiest way to give MacOS access to Linux filesystems).

MOL relies on some special kernel modules to run; for this reason, the first time you execute the *startmol* script after rebooting you need to do so as root. Thereafter, permitted non-root users can use *MOL* to run Mac applications under Linux.

MacOS X

The big imponderable in Linux-Macintosh connectivity is MacOS X. How do you deal with it – like a Macintosh, or like a Unix machine?

MacOS X is easier to deal with than traditional MacOS. For example, it can mount NFS-exported filesystems, and print to traditional Unix *lpd* print queues – Apple have even licensed *CUPS*, the *Common Unix Printing System*, for a future release. If installed on a UFS partition, MacOS X files can be grabbed by a Linux system on the same machine with ease.

However, it would be a mistake to assume that MacOS X is just like Linux. It may be Unix-based, but it's very different. For starters, the entire graphical user interface is different – it uses display PDF as an imaging system, virtually all applications can be scripted using AppleScript (an object-oriented scripting language that treats applications as objects that provide methods and treats files as instance data). While there are some traditional configuration files in `/etc`, most system configuration is carried out via *Netinfo*, a system introduced by NeXT that provides an LDAP-like distributed hierarchical database that subsumes and replaces everything from NIS+ to DNS.

Details of *Netinfo* are available from Apple; there's an introduction in the developers "Introduction to MacOS X" at <http://developer.apple.com/techpubs/macosx/Essentials/SystemOverview/SystemOverview.pdf> and there's a

(no-longer-supported) open source port of the Darwin *Netinfo* tools available from www.padl.com/OSS/NetInfoforLinux.html. As yet nobody seems to be working seriously on building *Netinfo* interoperability into Linux, which is a shame – it looks like a very powerful distributed directory service, similar to LDAP and NIS+ but more powerful.

Summing up

This has been a brief overview of the field of Mac/Linux interoperability. We've seen that a Linux box can be a useful network server for Macintosh clients, and a PowerPC Linux machine can interoperate with MacOS at least as well as an Intel Linux machine can work with Windows – better, if you consider that *MOL* is Free Software, but functionally equivalent to *VMWare*. Given that PPC Linux is considerably faster than MacOS X, and provides server features that MacOS X has yet to acquire (such as decent packet filtering and support for weird filesystems), Linux is just as worth considering as the heart of a Mac network as it is – with *Samba* – as the centerpiece of a Windows LAN.

With the release of MacOS X, the Macintosh world is now steering a convergent course with Linux and Unix. Unlike Microsoft, Apple reaps no commercial reward by building in barriers to interoperability with other systems. So over time we can expect to find ever tighter coupling between the Linux and MacOS X environments. It's going to be an interesting few years for those of us who think there's life beyond x86! **LXF**

“Linux is just as worth considering as the heart of a Mac network as it is – with Samba – as the centrepiece of a Windows LAN.”

Most of the usual applications have been ported to Linux PPC; here we can see *OpenOffice.org* running under KDE, along with a *GNOME* application (*gvim*). About the only stuff you won't find are commercial programs with binary-only releases – developers still seem to think that Linux is an x86-only world.



The power of KDE3

Jono Bacon greets the third incarnation of the all-encompassing K Desktop Environment, and tells us what's new and improved.

coming C++ toolkit gaining in popularity, was designed well and looked more attractive than many of the toolkits at the time, such as *Motif*.

KDE 3.0 was released on April 3rd 2002, sporting a number of new features and capabilities.

What is new?

Without a doubt, the main reason for the upgrade from KDE 2.x to KDE 3.0 is the switch to an updated Qt. Qt has been updated from 2.x to 3.x and has added a number of additional

features, bugfixes and performance improvements. Here is a summary of Qt improvements benefiting KDE 3.0:

- Better database support, including *MySQL*, *PostgreSQL* and *Oracle*. More databases can get support through a simple database driver structure.

- Explicit linking and plugins. The *QLibrary* class provides a platform independent wrapper for runtime loading of shared libraries.

- Unicode support for bi-directional and various other types of languages.

- Additional tools. *Qt Designer*, *Qt Linguist* and *Qt Assistant* speed development and maintenance.

One of the most prolific projects in the Free Software world is without doubt the K Desktop Environment; better known as KDE. Back in 1996 the project founder, Matthias

Ettrich, first called for coders interested in developing an easy to use and robust desktop environment for Unix-based systems – renowned for their power, but also for complexity due to their command line interfaces.

Back then Ettrich needed to make a decision about the choice of widget set (graphical buttons, scrollbars etc) to construct KDE with. After some discussion he decided on the use of Trolltech's Qt toolkit. Qt was an up and



■ Numerous bugfixes and speed improvements.

Not only have all these benefits been transparently passed on to KDE 3.0, but there have been a number of improvements that have been developed by the KDE team:

■ A number of bugfixes and speed improvements. Applications start noticeably faster in many cases.

■ Improvements to dialogs – support for different views, previews and filters.

■ Look and feel has been improved – redesigned icons, sidebar down the side of the K Menu, further fixes.

■ Printing has been vastly improved.

The framework consists of a print command/dialog; a printing manager; a job viewer for queue control and management; a Command Editor for cascading a series of external print job filters such as *enscript*, *a2ps* and *pamphlet* and GUI elements to config these filters; a wizard for auto-detecting and installing new printers; and a *CUPS* configuration tool.

■ Address book – new KDE address book library which provides a central address book to all KDE applications. Based on the vCard standard, and has provisions for extension by LDAP or database servers.

■ The new lockdown framework, which is essentially a permissions-based system for altering application config options. The kiosk framework supplements KDE's config framework with a simple application API which applications can query to test authorisation for certain operations.

■ *Konqueror* has had a raft of updates, the most notable being its improvement of JavaScript rendering. Plugin support also improved.

Should I upgrade?

A number of people have considered whether there is a real need to upgrade to KDE 3.0. In many cases KDE 2.x is working well for them and they are happy. Is there really need for the latest snazziest version of KDE?

Although these are valid concerns, an upgrade to KDE 3.0 is highly recommended. This is mainly due to the large upgrade to Qt 3.x, and also the sheer number of additions that make KDE 3.0 a more usable environment. Also, with the release of KDE 3.0, many developers of 2.x software are in the process of porting

their projects to the 3.x series. This porting path is a straightforward one and hence the transition to KDE 3.0 based software will be quick.

An important thing to note is that KDE 3.0 is not binary compatible with KDE 2.x. What this means in English is that KDE 2.x software will not run under KDE 3.0 and *vice versa*. This is an important element to consider as KDE 2.x software gradually gets discontinued.

Installing KDE 3.0

The KDE team have always worked on just releasing source distributions, leaving the binary packaging up to the independent packagers and volunteers. There are RPMs available for all of the major distributions and Debian volunteers have packaged up some DEBs – available externally from the Debian archive, or inside the archive when the bug-tracking process has finished.

If you do not find any packages for your distribution (check the CD/DVD!), or would like to take the 'purist' route to installing KDE 3.0, it is noted that you can install the source code, on this month's discs. See the *Getting KDE 3.0* box for more details.

To install the packages, you may require some further libraries that KDE 3.0 is dependent on. See your distribution's docs for the specific instructions of getting and installing these packages. For more details on what packages are required for KDE 3.0, see *Software Requirements* box. To install the main KDE 3.0 packages follow these instructions:

For RPM's:

If you are using Red Hat, Mandrake, SuSE, etc, you can use your distribution's graphical installer.

Alternatively you can install them from the command line with:

```
rpm -i package-name.rpm
```

For DEB's and Debian

If you would like to upgrade from within the Debian archive, simply issue the following commands:

```
apt-get update
```

```
apt-get install kde
```

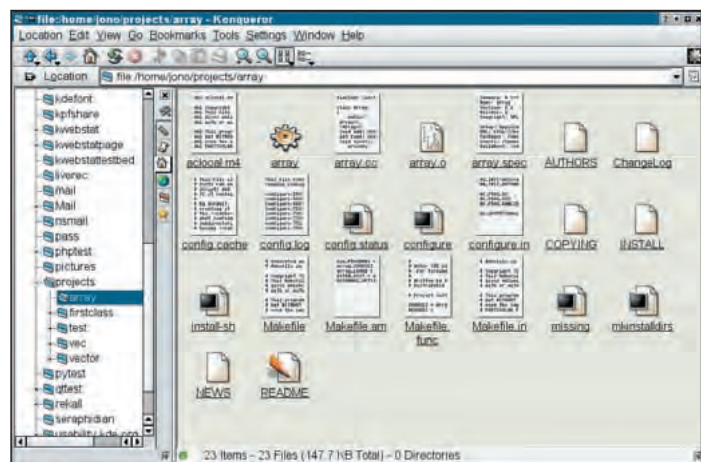
If you already have KDE 2.x, you will be able to upgrade with:

```
apt-get upgrade
```

If you would like to install the packages separately, you can install them with:



Konqueror browsing the web – now with working JavaScript!.



Konqueror as a file browser/manager – also improved.

dpkg -i package.deb

Please note that at the time of going to press the Debian packages were not yet available, due to the impending release of Woody. Check the latest release at [http://packages.](http://packages.debian.org/unstable/x11/kde.html)

“Konqueror, the full-featured kitchen sink of a browser has improved Javascript support – Javascript menus now work.”

[debian.org/unstable/x11/kde.html](http://packages.debian.org/unstable/x11/kde.html), or do a search for unofficial .debs.

Compiling the source

If you have some trouble with packages, or the packages are not available for your distribution, you can compile the source code specifically for your system. You will need to ensure you have the correct tools installed to compile the source and

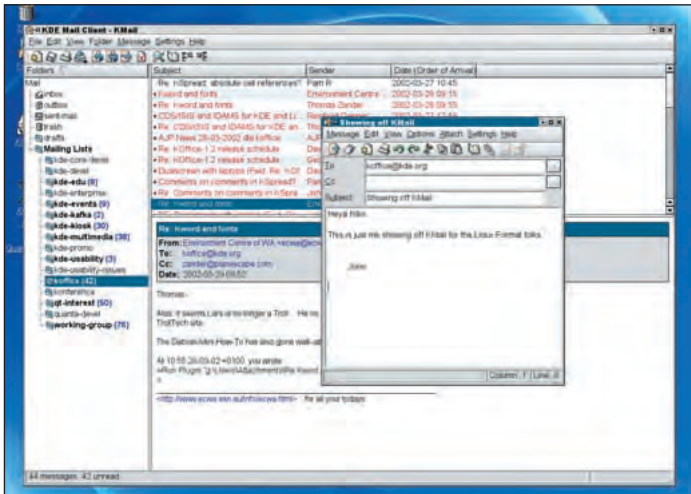


KDE

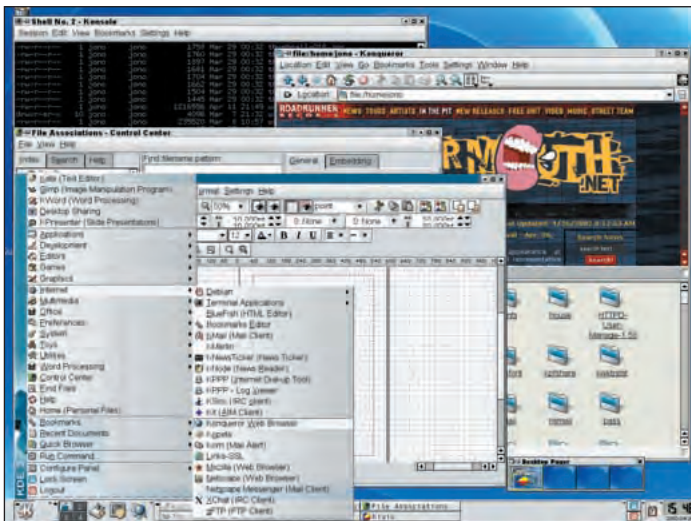


some dependencies – see the *Software Requirements* box for details.

The first thing you will want to compile after the tools and other dependencies have been installed is Qt. You can get Qt 3.0.3 from www.trolltech.com. Download the tarball to a directory where you want to keep your source code. I normally



KMail in full flow – fast, powerful and secure.



A full featured KDE session (you can't see the Qt improvements).

Getting KDE 3.0

Download for your distro

The KDE project have only ever released source code officially, although many volunteers and companies compile and package KDE for various distributions. The current source and some distro packages are on this month's discs. Others can be often found on the particular distribution's website or FTP, or they can be added to the upgrade path so when you next use your upgrading tool, KDE 3.0 will be automatically installed. Some of the following links may be useful:

RPM Archive (multiple distributions) – www.rpmfind.net

DEB packages for KDE – <http://packages.debian.org>

If you would like to get the latest source tarballs for KDE 3.0 to compile yourself, get them from: ftp.kde.org

There is also a list of mirrors that you can use, if the connection is slow, at www.kde.org/ftpmirrors.html

create a `/sources` directory:

```
mkdir /sources
```

(you need to be root to do this)

I then set the correct permissions

for my `/sources` directory:

```
chown jono /sources
```

Next, unzip and untar the Qt tarball:

```
tar zxvf qt-tarball.tgz
```

We now need to set some environmental variables for Qt. This is easily done using a shell startup file such as `.bash_profile`. Go to your home directory and load `.bash_profile` into your favourite editor. Add the following lines:

```
QTDIR=/sources
```

```
PATH=$QTDIR/bin:$PATH
```

```
MANPATH=$QTDIR/man:
```

```
$MANPATH
```

```
LD_LIBRARY_PATH=$QTDIR/lib:
```

```
$LD_LIBRARY_PATH
```

```
export QTDIR PATH MANPATH
```

```
LD_LIBRARY_PATH
```

Next you need to re-login to let the changes take effect for your shell. Then go back to the directory with the Qt source and enter:

```
./configure -system-zlib -qt-gif  
-system-libpng -system-libjpeg  
-plugin-imagfmt-mng -thread  
-no-stl -no-g++-exceptions
```

This will configure Qt for your system.

Now you can compile it with:

```
make
```

You do not need to do a **make install** like with many Linux programs, but if you get the error:

```
can't load library 'libqt.so.2'
```

or similar, you need to add:

```
$QTDIR/lib
```

to `/etc/ld.so.conf` (as root). You then need to run `ldconfig` (as root):

```
/sbin/ldconfig
```

Now Qt is installed, we can get the KDE packages installed. The KDE packages need to be installed in the following order: *Arts*; *kdelibs*; *kdebase*

After these packages we can install the others in any order we want. The compilation instructions for each package is the same. First unpack:

```
tar zxvf kde-package.tgz
```

You can then compile the package with the following commands:

```
./configure
```

```
make
```

```
make install
```

If you want to install KDE 3.0 to a specific location, make sure that when you configure each package, you

specify the location of the directory:

```
./configure --prefix=/opt/kde
```

(replacing `/opt/kde` with the directory where you want to install it).

Application updates

KDE 3.0 is without doubt a lot more versatile than KDE 2.x, with a raft of improvements and fixes, and I have already covered a lot of the main updates. There are however, a number of updates to individual applications which are worth pointing out. Many of these applications are ones you may already use with KDE 2.x which are improved further in KDE 3.0.

KMail (kdenetwork)

KMail is faster and more powerful. Its increased support for security in KDE 3.0 helps you plug it into your security policies, and its enhanced support for folders, expiration times and filters help *KMail* to become more suitable for reading mailing lists.

- SMTP authentication – useful for the increasing number of SMTP servers requiring login details

- SMTP over SSL/TLS

- Pipelining for POP3 – faster download times over slow connections

- On demand downloading or deleting without downloading of big mails on a POP3 server

- Creating/removing of folders

- Drafts/sent/trash folders on server

- Automatic configuration of the POP3/IMAP/SMTP security features

- Identity-based sent & drafts folders

- Expiry of old messages.

KOrganizer (kdepim)

KOrganizer is a useful tool for scheduling and appointment management, and it has been extended with a plugin interface so the program is easily extendable by developers. *KOrganizer* experienced fervent development leading up to the KDE 3.0 release and group scheduling, improved views, extended alarms *et al.*, are all useful additions.

- Plugin interface

- Group scheduling

- Split alarm daemon in a lowlevel and a GUI frontend

- Pinning contacts to appointments and TODO's

Noatun

The development of *Noatun* has been a rapid one and it is now a capable media player. Support for *Winamp* skins, *icecase* and *shoutcast* will be

useful for users, and it has benefited from improved stability.

- *Winamp* skin loader
- *Icecast/shoutcast* streaming
- Hide close status and tag displaying

Konqueror

Konqueror, the full featured kitchen sink of a browser, has gained a number of improvements in both its web browsing and file management/viewing facilities. The major improvement for web users will be the improved JavaScript support (most notable is that JavaScript menus work now on many sites) and this makes *Konqueror* a much more capable browser. Further support for MIME types; file previewing has improved.

- Additional IOSlaves
- Media player
- Improved Sidebar
- Animated icons/improved file info
- Improved JavaScript support.

KOffice

Although not officially released as part of the KDE 3.0 distribution, and still in heavy development towards the next release, *KOffice* has gained notable improvements across the board.

Porting to KDE 3.0

If you are a developer who has written a KDE 2.x application, porting your application is not a major issue.

The first thing to be aware of is the porting path from a Qt 2.x to 3.x application. For more details on this see <http://doc.trolltech.com/3.0/porting.html>. Qt 2.x and 3.x are largely source compatible, but for some classes in Qt 3.x that replace or extend those of Qt 2.x. The major areas to be considered are:

- *QFont* related code. *QFont* has changed a lot from 2.x to 3.x. The API has remained fairly similar to preserve most source but some enums have disappeared, such as *CharSet*.
- *QMultiLineEdit* has been replaced with the rich text engine, but is included in Qt 3.x for compatibility.
- The classes *QArray*, *QCollection*, *QList*, *QListIterator*, *QQueue*, *QStack* and *QVector* have been renamed.
- The *QSortedList* class is now obsolete. Consider using a *QDict*, a *QMap* or a plain *QPtrList* instead.
- *QRegExp* has been rewritten with support for Perl expressions.

Details on porting your app from KDE 2.x to KDE 3.x can be found in

Software Requirements

Big software has big dependencies

To install KDE 3.0 you need some basic software which is required for it to run. You can, however, add further software which KDE can make use of to make it even more powerful. This box gives details what you need and what additional software you can add. For details on getting the right packages for your distribution, see your distribution's documentation.

Required software

- Linux (almost every Linux distribution can run KDE 3.0);
- Qt 3.0.3;
- X Server and optional extensions (XFree86 4.2.x, which includes all the enumerated extensions, is recommended)

Required software to compile KDE

- gcc-2.95.x
- Automake
- Autoconf

Optional software

- XFree86
 - the RENDER extension for beautiful anti-aliased fonts
 - the DPMS extension for Energy Star display power management
 - the Xinerama extension for modern multi-head displays
 - the XVideo extension for enhanced video playback
- *GhostScript* for PostScript/PDF support (preferably 6.50 or later)
- *MySQL*, *PostgreSQL*, or *unixODBC* for database support
- *Berkely DB II* is highly recommended for *KBabel*, the KDE translation tool
- Python for scripting in some *KOffice* components
- Perl for scripting in *KSirc* and automating updates of configuration files for new releases
- *gzip* and *bzip2* for data compression
- *TeX* and *LaTeX* for document processing
- *CUPS* >= 1.1.9 for enhanced printing administration, options and usability
- *gphoto2* >= 2.0.1 devel for accessing images on digital cameras

- *SANE* for scanning support
- *lm-sensors* for monitoring motherboards
- *mttools* for accessing a floppy disk as floppy:/ from *Konqueror*
- *Java Virtual Machine* >= 1.3 for Java applet support; a recent version of *Lesstif* or *Motif* for Netscape Communicator plugin support
- *WINE* for executing certain MS Windows controls and applets
- *Crossover Plugin* for Quicktime, *Shockwave Director*, *Windows Media Player 6.4* and MS Office viewer support
- *OpenSSL* >= 0.9.6x for HTTPS, SFTP, SSH, VPN and more (versions 0.9.5x are no longer supported)
- *GnuPG* or *PGP* for email/document encryption/decryption
- *JSSE1.0.2* for Java applets requiring SSL/HTTPS (common with online banking) – included with a *JVM1.4*
- *PAM* for services (such as login) authentication
- *pppd* for dialup networking
- *libsmb* for browsing Windows/NetBIOS shares
- *libldap* for LDAP address book support
- *FAM* for efficient file/directory change notification
- *OpenGL* for some 3D screensavers and some 3D graphics programs
- *SDL* >= 1.2.0 for some *Noatun* plugins
- *libtiff* for viewing facsimiles
- *libmng*, *libpng* and *libjpeg* for viewing images
- *imlib1* for using *Kuickshow* for viewing a wide variety of image formats
- *freetype 2* for anti-aliased font handling and manipulation
- *PDFInfo* for enhanced PDF file browsing
- *mpeglib* for video playback (included with *kdemultimedia*)
- *ALSA* for more advanced audio support
- *cdtools/cdparanoia* for ripping audio CDs
- *LAME* for encoding MP3 files
- *libogg/libvorbis* for Ogg Vorbis encoding/playback
- *XAnim* for *aktion!*'s video engine
- *libaudiofile* for playing .WAV audio files

the *kdelibs* in KDE3PORTING.html or at <http://webcvs.kde.org/cgi-bin/cvsweb.cgi/~checkout~/kdelibs/KDE3PORTING.html>, changes include:


- The API for *kicker* (panel) applets has changed. The API takes positioning into account much more stringently now, and can notify an applet of positioning changes.
- The theming facilities are much improved. *KStyle* is used for many theming purposes and *KThemeBase* and *KThemeStyle* have been ported over to it.
- *KAccelMenu* has been removed – use Qt's facilities for this or, preferably, the XML-UI for building menus and actions.

Although KDE 2.x and KDE 3.0 are largely source compatible, the build system will need to be made aware of Qt 3.x and KDE 3.0 when you compile. The best way to get this working is to use *KDevelop* or

KAppTemplate to generate a working set of *Autoconf* and *Automake* scripts and then replace the ones in your KDE 2.x project accordingly.

Conclusion

KDE 3.0 is a big release for both developers and users, and I have looked at main features and upgrades that have occurred in the new release for both KDE and its associated applications. I have shown the most relevant updates and changes for you here, but there are a large number of changes I have not had space to cover; I suggest you read these at www.kde.org/announcements/changelog2_2_2to3_0.html.

The project continues to develop and evolve, and not only improves itself, but associated applications that run on KDE such as *KOffice* and *KDevelop* enjoy the benefits of the KDE developers' enthusiasm. 

Tutorials

Our experts offer help and opinions on a whole host of Linux applications

Your guide to getting things done!

Whether you are just starting out in Linux, or an experienced veteran, there's always more to learn. Every issue of *Linux Format* is packed full of practical advice, and nowhere is it more concentrated than in our tutorials pages.

Here you'll find expert guides to all sorts of things, from Basic Linux usage to understanding and deploying network solutions, from simple script coding to the complexities of Perl regular expressions, Java server apps and more. We aim to bring a good mix of tutorials to each issue, but if you have any suggestions for topics you'd like us to cover, why not contact us, by post, by email (linuxformat@futurenet.co.uk) or log on to our website and post your suggestions in our special forums? (www.linuxformat.co.uk). Hope to hear from you soon!

Nick Veitch EDITOR

How code is represented

Including code in magazines can be tricky, but we hope our notation will help it become clear. When lines are too long for our columns, the remaining text appears on the next line in a solid blue box:

```
procedure
TfrmTextEditor.mniWordWrapClick
(Sender: TObject);
otherwise, there is usually a gap
between lines:
begin
mniWordWrap.Checked := not
end;
Usually, you'll find the code on
our CD/DVD too.
```

THIS MONTH...

Apache

We focus on how to provide restricted access to parts of a website, and how to keep a record of the traffic to your site. As well as checking out personal web pages for hosting multiple websites **p66**

Perl

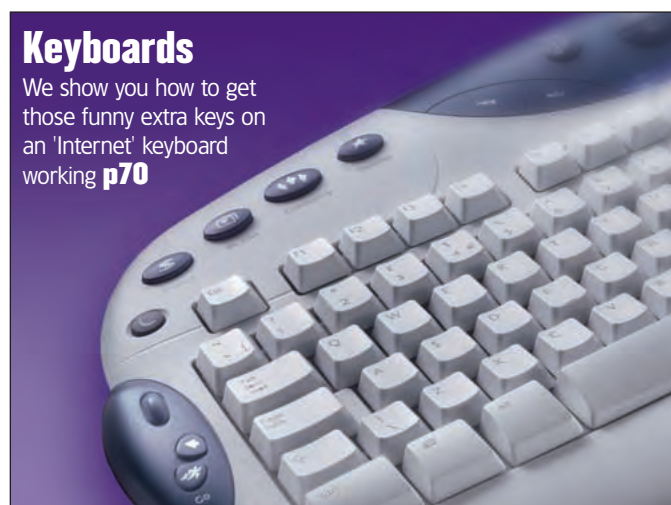
How much power can you get in a single line of code? We explore the art of the one-liner – and show you some handy Linux power-tools **p74**

Java

This month learn how to read XML documents in Java with our introduction to XML parsing and SAX – the simple API for XML **p78**

Keyboards

We show you how to get those funny extra keys on an 'Internet' keyboard working **p70**



Kylix

We raise the flag for some simple graphics with the TCanvas class – swap your pencils for some geometry calculations and join our art class **p80**

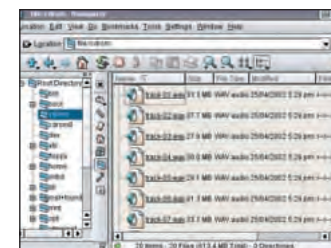
CGI

Worried about cross-site scripting attacks? We show you how to keep out the crackers. Remember what Mulder said: "Trust no-one" **p86**

TIP OF THE MONTH!

Have you ever wished that you could access data stored on a CD more flexibly under Linux? If so, then maybe the *CD filesystem* will help. This is a kernel module which creates a virtual filesystem for mounted CDs, listing all the tracks as normal files.

CDfs exports data tracks as raw ISO or HFS images, audio tracks as WAV files and video tracks as MPEG movies. Thus tracks can be easily copied, since they are just files; audio and video tracks can be played directly; and data tracks can be mounted via the loopback device. *CDfs* also exports the disc's boot image as a file and provides a track listing and CDDb ID for the disc in the */proc* filesystem.



Access your CD tracks as files with *CDfs*.

Other benefits include the ability to access more than one data track on a CD (including older sessions on a multi-session disc).

Mounting a CD under *CDfs* is straightforward – you simply need to specify the filesystem type as **cdfs**.

Flexible CD mounting

For example:

```
mount /dev/cdrom /cdrom -t cdfs
-o ro
```

By default, *CDfs* will export the sessions of a multi-session disc separately as files. You can mount the first session with, for example:

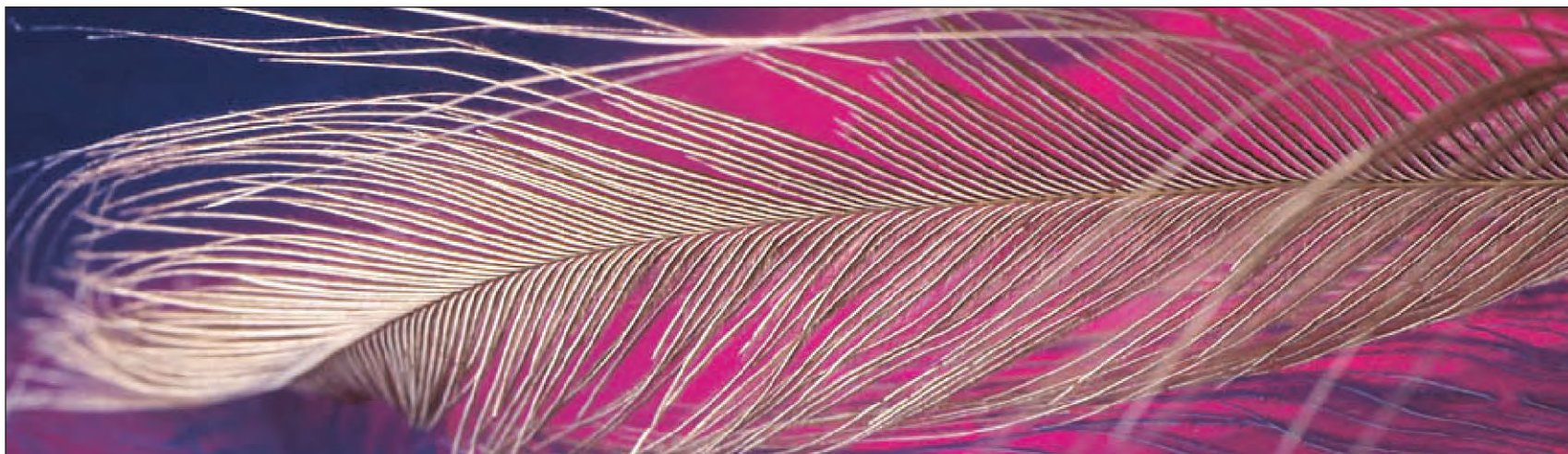
```
mount /cdrom/session_1-1.iso
/mnt/tmp -t iso9660
```

If you specify the **single** option to *CDfs*, then the multiple sessions will appear as one image – as per normal:

```
mount /dev/cdrom /cdrom -t cdfs
-o ro, single
```

For more information and to download *CDfs*, see:

<http://www.elis.rug.ac.be/~ronsse/cdfs/>



ACCESS CONTROL AND LOGGING

Apache webserver

This month **Chris Brown** examines hosting personal webpages and configures access control and logging.

Last month, if you can remember back that far, we left our *Apache* web server happily playing virtual host to the Carshalton Pregnant Mum's Leapfrog Team, The Sumo Wrestler's Glee Club, and the Hillman Minx Owner's Society. The latencies inherent in magazine publishing being what they are, I'm writing this prior to the publication of that tutorial, so the flood of hate mail from ardent Minx fans has not begun; nor have I yet received any irate letters from Carshalton.

In this, the third tutorial in our four-part series, we're going to focus on how to provide restricted access to parts of a website, and how to keep a record of the traffic to your site. First, however, we're going to look at one final mechanism for hosting multiple websites which we didn't cover last week – personal web pages.

Personal web pages

Imagine you're running a website at a school or college. To encourage your students to become citizens of the Wired Age, you would like to provide each with the opportunity to build her own website. You could do this, of course, by assigning each student a subdirectory of the **DocumentRoot**. So for example, if the document root was `/var/www`, student Jo Green might put her web content into the directory `/var/www/jgreen` which would be accessible through a URL like `www.somecollege.ac.uk/jgreen`. You'd need to be careful to assign ownership and access permissions on these directories so that students cannot modify one another's content. (Children can be so cruel, sometimes.)

There's a better way. You can tell *Apache* to look in a specified subdirectory of a student's home directory to find that student's web content. The required directive in `httpd.conf` is simply:

```
UserDir public_html
```

This says that every student with an account on the machine can put web content into a directory called `public_html` under their home directory. As the administrator, you can pick whatever name you like here, but all students must use the same sub-dir name.

Under this scheme, if Jo Green has a user account `jgreen`, and her home directory is `/home/jo`, then her web content is accessible via URLs like `www.somecollege.ac.uk/~jgreen/stuff.html`, which will cause the server to retrieve the file

`/home/jo/public_html/stuff.html`. Notice I've deliberately chosen a scenario where the user account name (`jgreen`) is not the same as the home directory name (`jo`). It's the account name, not the directory name, which you put after the `'~'` character in the URL.

Personal websites do not provide the same capabilities as fully-fledged sites created using the virtual hosting techniques we examined last month – they do not have a virtual host container directive associated with them in which to put user-specific config settings. For example, you can't log hits to each user's site in a separate file, and you can't separately specify which (client) sites can connect. But they provide a cheap-and-cheerful approach to providing private web space which requires minimal administration apart from creating the individual user accounts in the first place.

Who's that knocking?

Some websites are only too happy to make their entire content visible to anyone who cares to visit. Some aren't. *Apache* provides several different mechanisms to control access to the site, or to parts of it. The simplest methods limit access based simply on the IP address or hostname of the machine on which the browser is running. The directives which control this are **order**, **allow** and **deny**. In theory, there are many ways in which these can be combined, in practice there are only two sensible policies:

The first policy is to deny access to everyone by default, and to selectively allow access from specific clients. For example:

```
order deny,allow
deny from all
allow from 192.168.1.0/24
allow from goodguys.com
```

The **order** directive is important because it controls the order in which the deny and allow rules are applied. (The order in which the rules are written in the config file is irrelevant.) In the allow list, you can specify complete IP addresses or full machine names, or (as our examples show) you can specify subnets or domain names. The first allow line in the example allows access from all machines for which the top 24 bits of the IP address matches `192.168.1.0`. Since this particular network ID is part of the 'private' IP address space, which is never visible on the real Internet, the entry probably relates to our own local network.

The other, more open, policy is to allow access to everyone by default, and then selectively deny access from specific clients e.g.:

An aside...

Q. What's the difference between theory and practice?

A. In theory there is no difference, but in practice there is!

```
order allow,deny
allow from all
deny from badguys.com
```

Container directives

You can't put these directives at the 'top level' of the config file. (Well, you can, but you will get a delicious error message when you start the server – "allow not allowed here".) Instead, you put them into what are called "container directives" that limit the effect of any directives within the containers to specific parts of the site. Here's an example. Notice the syntax has an HTML feel:

```
<Directory /home>
order deny,allow
deny from all
allow from 192.168.1.0/24
allow from goodguys.com
</Directory>
```

This means that the **deny** and **allow** directives only apply to requests for pages in and below the /home directory. You can get fancier than this; for example, you can include regular expressions in the directory name and match against those.

Who are you?

Controlling access on the basis of the identity of the machine where the browser is running is a start, but controlling access based on the identity of the person sitting at it is much more useful. Unfortunately, it's also more work. Whereas the Internet has a global way of identifying a machine, via its IP address (or, equivalently, by its name), there is no equivalent way of identifying people. One day, perhaps, we'll have live, biometric identification of the user, verified against a universal database of retinal scans and tied in to globally unique personal identifiers. And we'll have perfected porcine aviation too. Until then, we have to deal with a myriad of user account databases in many formats, with no consensus of how to verify or even represent user identity.

How does *Apache* do it? Well, if you want to get really fancy, there are third-party modules for *Apache* which can verify user credentials against a whole variety of account databases – Kerberos, LDAP, NT domain controllers, NIS, SQL databases, and others. Here, we'll content ourselves with a look at how you can create a simple user database in a file (similar to the regular password file in Linux) and authenticate a user against it using the basic authentication mechanism built into the HTTP protocol.

Authentication and authorisation

Before we look at the details of how this works we need to understand that any access control mechanism has two parts to it – authentication, and authorisation. Authentication is the process of making sure you know who the user is (usually by requesting a user name and a password). Authorisation is the process of deciding if the user is to be allowed to do something. The two usually go together. You can't do authorisation checks unless you know who the user is, (unless there's a specific access rule for anonymous users). Contrariwise, it's a bit pointless doing authentication without following it up with authorisation (unless, perhaps, you'd just like a user name to put into a log file).

Suppose you administer the website www.goodstuff.com. Perhaps it has some unrestricted content that anyone can access.

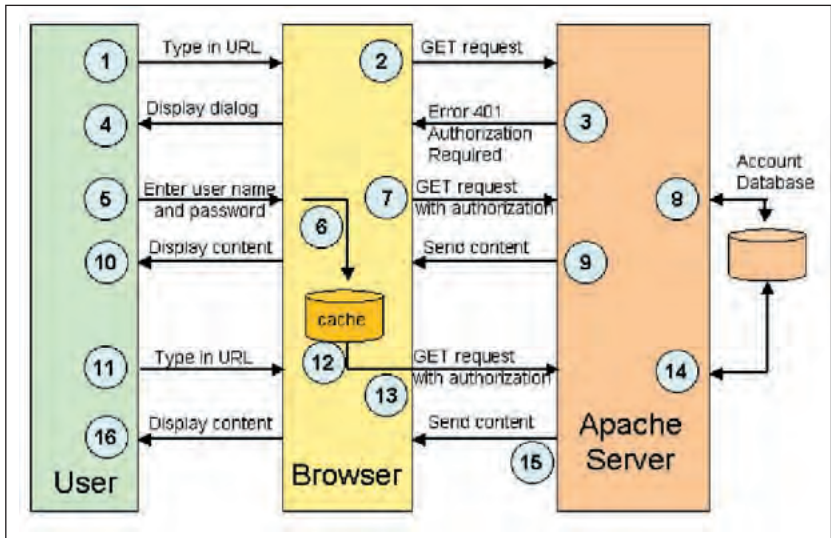


Fig 1: The sequence of events in authenticating into a protected realm.

But you also have a directory, let's call it *reallygoodstuff*, containing content which is restricted to users for whom you have established an account on your machine. To access that piece of the site, you require users to enter a name and a password. Effectively, they need to login to, or "authenticate into," that part of the site. Once they're authenticated, they should be able to move freely within that part of the site. You certainly don't want to make them re-enter a password every single time they access a resource within the restricted area of the site. The area of the website to which you gain access via a single login is called a realm, or sometimes a "sphere of protection".

Turning on authentication

OK, we have enough background. Consider this from `httpd.conf`:

```
<Directory /usr/local/apache2/htdocs/reallygoodstuff>
AuthType Basic
AuthName "The Really Good Stuff"
AuthUserFile /usr/local/apache2/conf/.htpasswd
require valid-user
</Directory>
```

The **Directory** container defines the directory to which our access controls will apply. The **AuthType** directive says that we're using basic authentication, against a flat (*i.e.* unindexed) file of usernames and passwords. The **AuthName** directive gives a name to the realm which we're protecting. In principle (though not in our example) there could be other directories elsewhere in the site which were part of the same realm. This name is also displayed by the browser in the dialog box it uses to collect your username and password – we'll come to that in a minute. The **AuthUserFile** directive names the file where the username/password data is held. Finally, the **require** directive says that only valid users (*i.e.* users who can supply a name/password pair matching one in the **AuthUserFile**) are to be allowed access.

Step by step authorisation

Fig 1 shows how the whole thing hangs together. The numbered steps go like this:

- STEP 1:** User enters the URL of restricted resource on the site.
- STEP 2:** Browser sends **GET** request for the resource to *Apache*.
- STEP 3:** Because the resource lies within a directory requiring an authorised user, the server returns a 401 "Authorisation Required" error to the browser. This error response includes the name of the realm for which authorisation is required.



- ◀ **STEP 4:** The browser displays a dialog box to the user, requesting username and password. (Fig 2 shows how it looks in *Konqueror*).
- STEP 5:** The user enters the username and password.
- STEP 6:** The browser stores the name of the realm, the user name and the password in a local cache.
- STEP 7:** The browser resends the **GET** request, this time with an Authorization line in the header, carrying user name & password.
- STEP 8:** The server checks these against the entries in the **AuthUserFile**. If there isn't a match, the server will return a 401 error again, at which point the user will see an "authentication failed" message from the browser. However, if the authorisation check succeeds ...
- STEP 9 & 10:** Server returns content and the browser displays it.
- STEP 11:** At a later stage, the user may enter the URL of another resource within the same realm. Here the browser is smart enough to realise that this request also requires authorisation, so:
- STEP 12 AND 13:** Browser retrieves the cached authorisation data and issues a **GET** request including the authorisation
- STEP 14:** The server again checks the data against the **AuthUserFile**. If successful:
- STEP 15 AND 16:** The server returns the content and the browser displays it.
- Notice that the user only has to authenticate once, but the browser authenticates every time it issues a **GET** request to a restricted resource.

Account creation

If you've followed things through this far, you might still be wondering how the account database (the **AuthUserFile**) on the server gets created. Well, it's created and managed by the command **htpasswd**, which is part of the *Apache* distribution. Here's a simple example of adding a user harriet to the file `/usr/local/apache2/conf/htpasswd`:

```
cd /usr/local/apache2/conf
htpasswd -m .htpasswd harriet
New password:
Re-type new password:
Adding password for user harriet
```

In practice, it would be unlikely that the web administrator would sit down at the server and enter **htpasswd** commands at the command prompt. More likely, the command would be issued "programmatically" from a server-side script. Typically, the website will have a page which says something along the line of "To see the really good stuff, you need to have an account with us". It will then present a form asking for a user name and a password and usually it will also capture other information such as your email address, and possibly some of your money. When this form is submitted, the server will run a server-side script to execute the **htpasswd** command that adds the user account.

Here's an extract from the `.htpasswd` file that **htpasswd** creates. It's basically like the first two fields of a regular Linux



Fig 2: The login dialog box.

password file – the user name and the hashed password.

```
chris:BYqbWdHZ3zNLQ
sandy:PbJL98/R.aao
harriet:$apr1$as.Lw/..$2BzoMwS6RT0o7c8K2x3fw1
```

Why, you may ask, is harriet's hashed password so much longer than the others? Well, it was hashed using *Apache*'s MD5 hashing function (that's what the **-m** flag on the **htpasswd** command specifies) whereas the entries for chris and sandy use the default Linux **crypt()** hash function (the one used for the Linux password file). Both are supported, and can be mixed in the same file.

Getting fancy with authentication

Basic authentication against a flat (unindexed) file of accounts is about as simple as you can get. For larger user populations there are *Apache* modules to allow you to hold the accounts in a proper database, such as Unix DBM files, or the free relational database *MySQL*, or a proprietary product such as *Oracle*. This is more efficient because they're indexed and can be searched much faster.

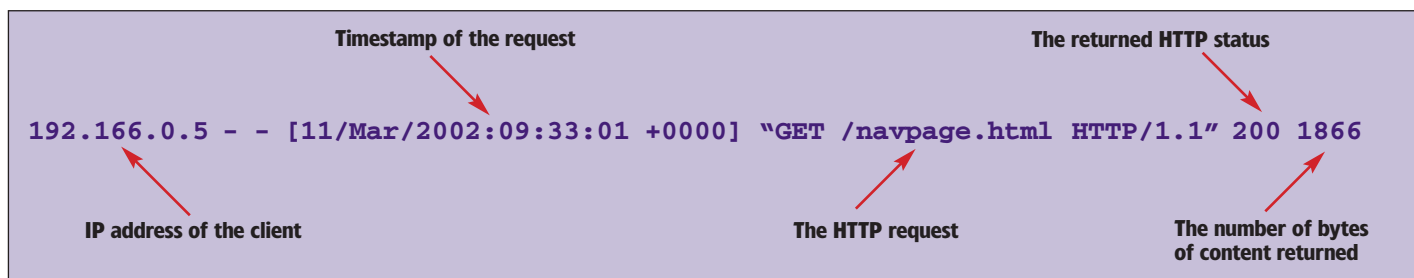
Apache will also do Message Digest Authentication, which uses a so-called challenge-response scheme for authentication which avoids passing the password across the network. However, not all browsers support this.

There's also a lot more flexibility available in the `httpd.conf` file than we've seen, to specify authorisation information. *E.g.*, we simply specified **require valid-user** which authorises access to any user who can authenticate against the account database. But you can also specify an explicit list of valid users here, or, more usefully, you can put lists of users into named groups (using a simple group file not unlike the group file in Linux) and allow access to one or more specified groups. This allows you to separate out the list of users who can be authorised into a realm from the list of users who can be authenticated to the server.

Index Error logging in Apache

Apache provides an extensive error logging facility. In my early days of using *Apache*, I rather ignored the error log file,

Fig 3: The anatomy of an entry in the access log.



preferring, for some strange egg-headed reason I now find hard to justify, to deduce the cause of problems by sheer intellectual prowess (hah!). Nowadays, I'll take all the help I can get, and will generally run *tail* on the error log at the first sign of a problem.

Error logging is very easy to set up in `httpd.conf`:

ErrorLog logs/error_log

tells the server where (relative to **ServerRoot**) to write the log.

The listing (*bottom of page*) shows three typical lines from my error log. The first item on each line is the date and time of the error. The second item is the severity of the error. These severity levels match the standard levels defined by the Linux logging daemon, *syslogd*. The third item is a description of the problem.

The first line describes a perfectly normal condition – *Apache* restarting after receiving a `SIGHUP` signal. You'll see that this is logged at a very low severity level [**notice**]. Next is a very common type of error, a request for a non-existent file. Typically this indicates a broken link somewhere on the site. Last is a much more serious error, which occurred as the server tried to start up and was caused by trying to run two different versions of *Apache* at the same time, causing them to fight over the use of port 80.

You can control the amount of stuff that gets logged with the `LogLevel` directive. For example, the entry:

LogLevel warn

will log errors at or above the [**warn**] severity. I would not recommend raising the **LogLevel** above [**error**] or you might miss something important.

... and the good news?

In addition to logging errors, *Apache* will also log each request which is received from a browser. This is also very easy to set up. In the default `httpd.conf` file, you will see two lines like this:

LogFormat "%h %l %u %t \"%r\" %>s %b" common

CustomLog logs/access_log common

The **LogFormat** directive specifies precisely what information will get logged. Each of the codes like **%h** and **%u** specifies a specific item to be logged. There are a lot of these codes. I won't bore you with a full list here, but just to give you the idea:

%h Displays the name of the remote host, if known, else its IP address
%t Displays the time of the request
%>s Displays HTTP status code returned for the request
%b Displays the number of bytes returned (excluding the HTTP header)

The word **common** at the end of the `LogFormat` directive defines a 'nickname' for this format. This one is called **common** because it actually defines what's known as the Common Log Format, which is a standard format defined by the World Wide Web Consortium, and is not *Apache*-specific. Using the Common Log Format is a good idea because there are lots of tools around to help you examine, filter, and summarise log files which conform to this format. The nickname is then referenced in the **CustomLog** directive which, in this example, tells *Apache* to write the access log to the file `logs/access_log`.

The access log is less varied than the error log – all the lines look the same! **Fig 3** shows the anatomy of a typical entry.

Logging within Virtual Hosts

If your web server is managing virtual hosts (and it probably is) you can if you wish (and you probably do) set up separate logs for each virtual site, by placing logging directives inside the virtual host containers, like this:

<VirtualHost 192.168.0.1>

ServerName www.leapfrog.co.uk

DocumentRoot /home/leapfrog/html

CustomLog /logs/leapfrog_access_log common

</VirtualHost>

What to do with logs

If you're running a production server, one of the things you'll definitely need to do with your log files is to rotate them – that is, to parcel (say) each day's logs into a separate file, enabling you to periodically purge the old ones. There's a program called *rotatelogs*, which is part of the *Apache* distribution (and not to be confused with the Linux utility *logrotate*) which will help with this. It relies on *Apache*'s capability to pipe the log into a specified program, which you do with an entry something like this:

CustomLog "| rotatelogs /path/to/logs/access_log 86400" common

One thing you probably won't want to do with log files is to actually read them. More likely, you'll use a log file analysis tool to pull out the statistics you're interested in. There are several free tools, such as *webalizer* and *analog*, and a shedload of commercial products such as *123LogAnalyzer*, *Wusage*, and *Summary*. These can pull out a variety of interesting statistics such as traffic volumes by hour, day, week etc., or a breakdown of accesses by the domain in which the client machine is named, or the operating system it's running. You can even look at the search engine keywords that were used to find your site.

Summary of Directives

Here's a summary of the directives we've looked at this month:

UserDir	Defines dir containing personal web pages
order, allow, deny	Controls access based on client machine
<Directory>	Defines container to restrict effect of the contained commands to a specified dir
AuthType	Specifies the type of user authorisation
AuthName	Names the "protection realm" being authenticated
AuthUserFile	Names the file containing <i>Apache</i> user accs
require	Specifies which named users can have access
ErrorLog	Names the file used to log errors
LogFormat	Defines the format of an access log
CustomLog	Enables logging to a specified file
LogLevel	Controls level of error messages logged

Next month we extend *Apache*'s functionality. **LXF**

Monitoring log files

Here's a handy trick. Bring up an extra terminal window and use it to run **tail -f** on the error log file. *Tail* shows you the last 10 lines of the file, and the **-f** flag says to keep going, and show any lines that get appended as the file grows. Of course you can do this with any file that gets written incrementally.

NEXT MONTH

In the final part of the series next month, we'll look at the way in which dynamically linked modules can be used to extend the functionality of *Apache*. We'll see how to build fancy directory listings, and we'll just dip our toes into the deep and murky waters of server-side scripting.

The error log

Every line tells a story

```
[Mon Mar 18 07:33:22 2002] [notice] SIGHUP received. Attempting to restart
[Mon Mar 18 07:42:59 2002] [error] [client 127.0.0.1] File does not exist: /home/index.html
[Mon Mar 18 13:00:42 2002] [crit] (98)Address already in use: make_sock: could not bind to address 0.0.0.0:80
```




MORE THAN ANYONE EVER WANTED TO KNOW ABOUT KEYBOARDS

Extra function keys, keymaps and Linux

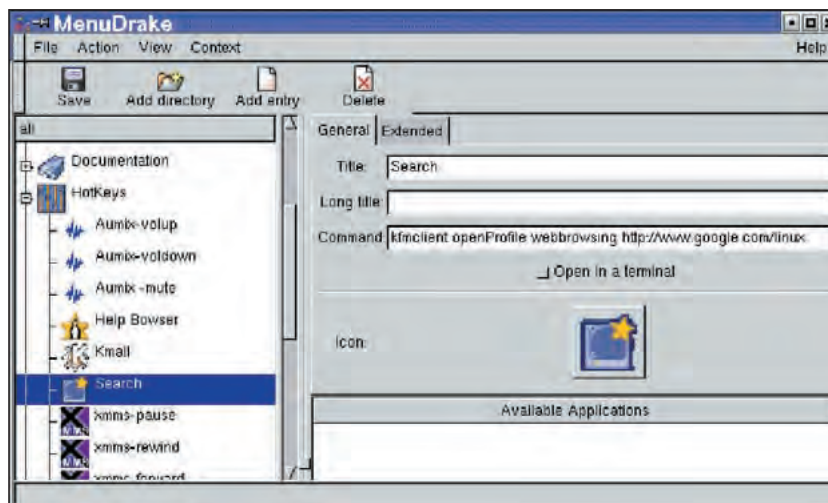
More than anything else, we hate to have hardware with features we can't use in Linux, especially those extra keys on the Internet Keyboard, so **Hoyt Duff** makes them work.

We're going to explore our keyboards with the goal of enabling the special keys found on our Logitech Cordless iTouch keyboard – the so-called "Internet Keys," which don't work by default. We will get them working with our KDE2 desktop primarily because KDE2 lacks an easy way to do it. You'll get enough info to apply these techniques to other keyboards and other window managers.

Here we have created an entry to open *Konqueror* in web browsing mode and go to Google. Anything that can be run from one command on the command line can go in "Command:".

How the keyboard works

Just to further confuse things, there are two sets of key mappings in use: one for the console and one for X11; they aren't even 100% compatible. Throw into the mix that all keyboards do not generate the same scancodes (some are even programmable), and it's a wonder that the keyboard even works at all.



CONSOLE As you press a key, the keyboard sends a scancode to the computer. When in scancode mode (RAW mode), a kernel driver sends the scancode directly to the running application. In keycode mode (MEDIUMRAW mode), it changes the scancodes to keycodes and sends those on. The keycodes are looked up in the keymap (XLATE mode) and are, for the most part, sent to the application as ASCII text (like the letter "a") or escaped ASCII sequences (like the Delete key). To make changes for console use only, you would edit the files in `/usr/X11/lib/X11/kbd`, editing the keymap you use. To see what keymap we are currently using:

```
cat /var/log/messages | grep keymap:
```

```
Feb 28 06:59:46 marvin keytable: Loading keymap: us
succeeded
```

So we know that the US keymap is loaded. Where is that keymap located? It's named `us.-something` so we can:

```
find / -name us.*
```

```
/usr/lib/kbd/keymaps/i386/qwerty/us.kmap.gz
```

Now we know where the console keymaps are, we find a plethora of interesting keymap files not only for different locales – many LXF readers are probably using the `uk.kmap.gz` file (the `.gz` means that files have been gzipped; use **gunzip** to uncompress them and **gzip** to compress them) – but for special keys as well. Of particular interest are the files found in `/include`; useful snippets of code to enable special keycodes, for things like the *Euro* and **Windows** keys. Have a look in the `kmap` file to see how these `.inc` files are included. All this is very interesting, but it does not help us when we are using our Internet keyboard and KDE2 desktop because keyboards are handled differently for X11.

X11 The X11 keyboard process is outlined in *The Linux Keyboard and Console HOWTO*. The actual keyboard handling is done through an extension to XFree86 called *Xkb* – and that can be incredibly complicated. There are several places in the X11 startup process where you can define keymappings. For example, Mandrake 8.x includes a few lines in `/etc/X11/Xmodmap` that force the assignment of the **Windows** key (it's mapped as F13). If you try to reassign that key *before* the Mandrake script runs, it

will be unset by the Mandrake script. Other alternatives would be `~/Xmodpmarc`, `~/xmodmap`, or an executable shell script in `~/kde/Autostart`. Try them all and use what works for you. The KDE *Control Center* has an option to choose a keyboard: look under *Peripherals/Keyboard*. If the choice is likely greyed out, the keyboard defined in `/etc/X11/XF86Config-4` is the one being used. Find it under the **Input Device** section. Our system shows:

Option "XkbRules" "xfree86"

Option "XkbModel" "pc105"

Option "XkbLayout" "us"

Two other options not used by default are: Option

XkbVariant The most common choice for this option is **nodeadkeys** – “dead” keys don’t type anything until you press another key – **winkeys** is also available. Option **XkbOptions**: A list of options can be found in `/etc/X11/xkb/rules/xfree86.lst`. For example, **ctrl:nocaps** disables the **Caps Lock** key.

Tools Provided by Linux

To begin mucking about on your system, you’ll need to be familiar with a few command line tools, all of which have very good man pages (with examples!! – all man pages should be as nice); please read the man pages.

xev Found in the X11R6-contrib RPM, this app will create a window (which needs to have the mouse focus to work as expected) and print in the console you launched it from the descriptions of X events that occur in that window, including key presses. For example, pressing the letter “a” will display:

KeyPress event, serial 28, synthetic NO, window 0x3e00002, root 0x4a, subw 0x3e00003, time 1339716214, (27,47), root:(31,67), state 0x10, keycode 38 (keysym 0x61, a), same_screen YES, XLookupString gives 1 characters: “a”

KeyRelease event, serial 28, synthetic NO, window 0x3e00002, root 0x4a, subw 0x3e00003, time 1339716371, (27,47), root:(31,67), state 0x10, keycode 38 (keysym 0x61, a), same_screen YES, XLookupString gives 1 characters: “a”

Now you have proof that a keypress has both a press and release event. The useful information here is that the keycode is **38** and the keymap identifies it as **keysym 0x61**, or “a”; the latter is blank if the keycode is undefined. Note that you will also see the events of your mouse moving and window focus changing, so be careful interpreting the output.

dumpkeys Displays current keymap (keyboard translation table)
getkeycodes Prints the kernel scancode (in hex) to keycode (in decimal) map.

showkey Displays the scancodes of the keypress. Most keys generate a separate code for press and release. Scancodes are shown by **showkey -s** and keycodes are shown by **showkey**.

setkeycodes Can be used to translate between scancodes and keycodes for use in the console only. For example, if your “moon” key produces **scancode e05f** (from *scankey*) and you want to assign it to **keycode 223**, you use the command **setkeycodes e05f 223**, then use **loadkeys** to define the function of the new key. This is for the console only and NOT for X. Note that while **showkey -s** in the console results in a scancode of **e05f** for the keypress, there is no keycode output for **showkey**, telling us that the key is undefined for the console. The output of **xev** (above) tells us that XFree has assigned this key a keycode of **233**. If XFree did not assign a keycode, we can use **xmodmap** to do so. **loadkeys** Loads a keymap that translates scancode to keycodes. **man 5 keymaps** shows the keymap format and also explains how your system can differentiate between control keys that are

pressed, and gives examples of assigning strings to keycodes.

xmodmap Edits and display the keyboard modifier map and keytable map that translate keycodes into keysyms for X. The man page includes some nice examples. The command **xmodmap -pk | more** displays the current keymap
setxkbmap This sets the keyboard mapping using the X Keyboard Extension. The *Xkb* map is made up from components found in `/usr/X11/lib/X11/xkb`.

kbd_mode Used to forcibly change the keyboard mode for RAW to XLATE and back

xkeycaps A frontend to **xmodmap** that is handy if your keyboard is supported (currently not including any ‘Internet’ keyboards).

Using **xev** allows you to explore the underpinnings of X events on your desktop.

Investigating your own keyboard

SCANCODES To discover what scancodes are sent by our Logitech Cordless I-touch Keyboard, we ran **xev** from an *xterm*, then pressed each “mysterious” key and making a list of the results. We see that the moon key generates:

KeyPress event, serial 28, synthetic NO, window 0x3e00002, root 0x4a, subw 0x3e00003, time 1339986154, (56,55), root:(60,75), state 0x10, keycode 223 (keysym 0x0, NoSymbol), same_screen YES, XLookupString gives 0 characters: ""

KeyRelease event, serial 28, synthetic NO, window 0x3e00002, root 0x4a, subw 0x3e00003, time 1339986161, (56,55), root:(60,75), state 0x10, keycode 223 (keysym 0x0, NoSymbol), same_screen YES, XLookupString gives 0 characters: ""

So now we know that the keycode for the key is **233** and the key is undefined (no characters). Make a complete list of the keys and their scancodes and save it for later.

USED AND AVAILABLE KEYCODES To see what keycodes are currently used, use **Alt-Ctrl-F2** to bring up a virtual console and **#dumpkeys > dumpkeys.txt** This will write a file named `dumpkeys.txt` that you can view in a text editor. It will list the



LinuxFormatTutorialKeyboards

« keycodes being used, show the escape sequences assigned to the function keys and the “compose” keys (what keycode is assigned when using the meta keys). This lets us see what keycodes are available.

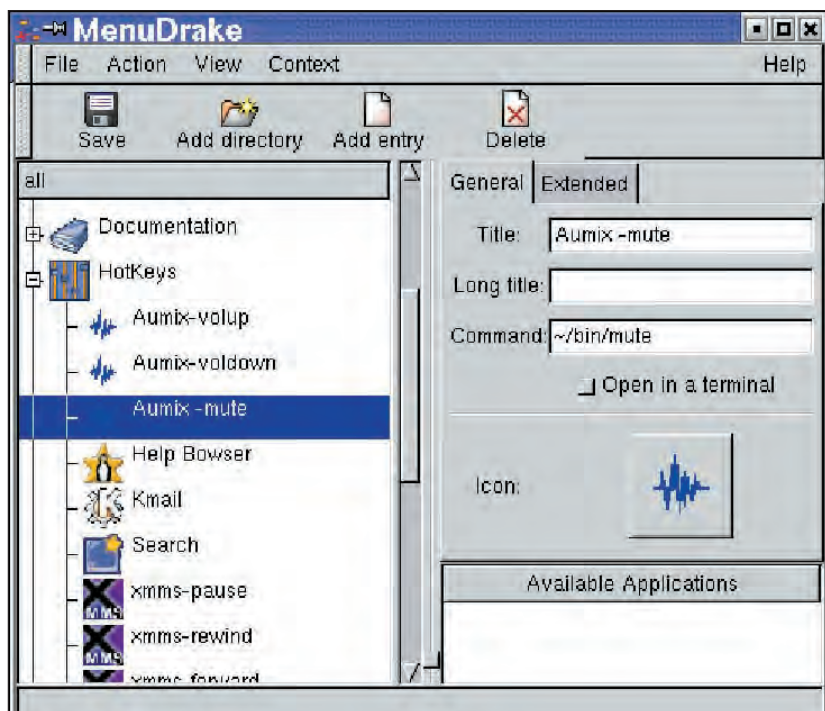
Enough theory, let's do it

Linux outdoes itself with providing multiple ways of accomplishing the same task. We strive to live up to the example. Examine them all and use what works for you.

METHOD 1: Once we have a list of each keycode from `xev`, we need to look in the file `/usr/X11R6/lib/X11/xkb/keycodes/xfree86`. Find the keycode in the right hand column and then write down its XFree86 code. For our “moon” key, it is **<I69> = 233**; next, make a backup copy of `/usr/X11R6/lib/X11/xkb/symbols/inet` and edit the file. We find the section that closely matches our keyboard – it's labelled **xkb_symbols “itouch”** – and edit it so that the XFree86 codes match our list. The “moon” key turns out to be **XF86Standby** and so on. For Mandrake users, we also have to change the XFree86 names as well since KDE/Qt won't work with them. For KDE, we suggest naming them F-something. Here is the original section of the file:

```
xkb_symbols "itouch" {
// Describes the extra keys on a Logitech iTouch keyboard.
name[Group1]= "iTouch";
key <I5F> { [ XF86Standby ] };
key <I1F> { [ XF86AudioMute ] };
key <I2B> { [ XF86AudioLowerVolume ] };
key <I2D> { [ XF86AudioRaiseVolume ] };
key <I22> { [ XF86AudioPlay ] };
key <I24> { [ XF86AudioStop ] };
key <I10> { [ XF86AudioPrev ] };
key <I19> { [ XF86AudioNext ] };
key <I32> { [ XF86HomePage ] };
key <I6C> { [ XF86Mail ] };
key <I65> { [ XF86Search ] };
key <I66> { [ XF86Start ] };
};
```

We can reference a script as well. Here we call a script that mutes and un-mutes *aumix*. Our scripts are kept in `~/bin`; yours can be anywhere accessible.



Here is the section modified for our model of the keyboard:

```
xkb_symbols "itouch" {
// Describes the extra keys on a Logitech iTouch keyboard.
name[Group1]= "iTouch";
key <I69> { [ XF86Standby ] };
key <I120> { [ XF86AudioMute ] };
key <I2E> { [ XF86AudioLowerVolume ] };
key <I30> { [ XF86AudioRaiseVolume ] };
key <I22> { [ XF86AudioPlay ] };
key <I24> { [ XF86AudioStop ] };
key <I10> { [ XF86AudioPrev ] };
key <I19> { [ XF86AudioNext ] };
key <I32> { [ XF86HomePage ] };
key <I6C> { [ XF86Mail ] };
key <I65> { [ XF86Search ] };
key <I66> { [ XF86Start ] };
};
```

And here is the section modified for use with KDE2:

```
xkb_symbols "itouch" {
// Describes the extra keys on a Logitech iTouch keyboard.
name[Group1]= "iTouch";
key <I5F> { [ F31 ] };
key <I1F> { [ F32 ] };
key <I2B> { [ F24 ] };
key <I2D> { [ F29 ] };
key <I22> { [ F21 ] };
key <I24> { [ F28 ] };
key <I10> { [ F27 ] };
key <I19> { [ F26 ] };
key <I32> { [ F22 ] };
key <I6C> { [ F30 ] };
key <I65> { [ F23 ] };
key <I66> { [ F25 ] };
};
```

Once the appropriate changes are saved, we open the control panel and go to Peripherals/Keyboard and look at the “Layout” tab. Here we need to uncheck “Disable keyboard layouts” and then select the Logitech iTouch keyboard. Once you “Apply”, test to see if the keysyms have been correctly assigned by using `xev`. You may or may not need to restart X.

METHOD 2: Once we have a list of each keycode from `xev`, we will use a text editor to create a file listing our desired conversions.

Save the file as `~/kde/Autostart/keycodes` and make it executable (**chmod +x**). You may execute the script from an `xterm` to initialise it; it will be run automatically whenever you restart KDE. Our file looks like:

```
#!/bin/sh
xmodmap -e 'keycode 162=F21' #Play/Pause
xmodmap -e 'keycode 178=F22' #Home
xmodmap -e 'keycode 229=F23' #Search
xmodmap -e 'keycode 174=F24' #Vol Down
xmodmap -e 'keycode 230=F25' #Run
xmodmap -e 'keycode 153=F26' #Forward
xmodmap -e 'keycode 144=F27' #Back
xmodmap -e 'keycode 164=F28' #Stop
xmodmap -e 'keycode 176=F29' #Vol Up
xmodmap -e 'keycode 236=F30' #Mail
xmodmap -e 'keycode 233=F31' #Standby
xmodmap -e 'keycode 160=F32' #Mute
```

To see if it is working, run `xev` and press the Internet keys.

Web resources

The world at your fingertips

Keyboard Scancodes by Andries Brouwer

<http://www.win.tue.nl/~aeb/linux/kbd/scancodes.html>

Linux Support for Funny/Functional Key by Rick van Rein

<http://rick.vanrein.org/linux/funkey/>

Remapping Keys, Mandrake User Org

<http://www.mandrakeuser.org/docs/xwin/xkeys.html>

The Linux Keyboard and Console HOWTO

www.linuxdoc.org/HOWTO/Keyboard-and-Console-HOWTO.html

Using those extra keyboard keys in X by Chris Jones

<http://www.linuxdude.co.uk/docs/Special-Keys/>

Internet Keyboards and KDE

<http://www.warpedsystems.sk.ca/article.php?sid=475&mode=8&order=0>

How to Setup International Keyboard in X Window with *Xmodmap* and *XKB* by Juraj Sipos

<http://www.linuxdoc.org/HOWTO/mini/Intkeyb/>

XKeyCaps Homepage

<http://www.jwz.org/xkeycaps/>

Xkb, the X Keyboard Extension

<http://www.tsu.ru/~pascal/en/xkb/>

METHOD 3: Create a `~/.Xmodmap` file. An example, in part:

```
keycode 162 = F21 #XF86AudioPlay
keycode 178 = F22 #XF86HomePage
keycode 229 = F23 #XF86Search
```

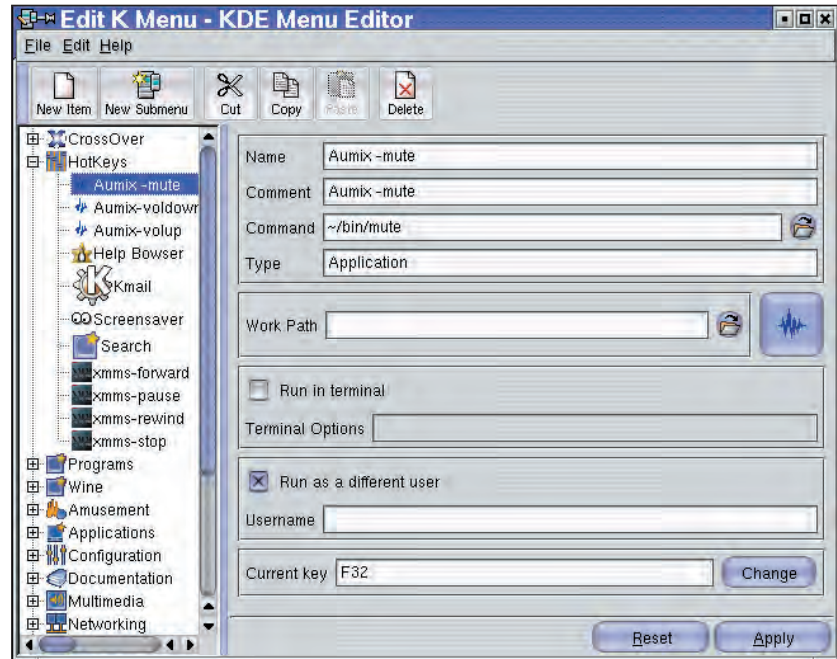
Again, we used the Function keys for KDE2; your window manager might just like the XF86 names. As before, test with `xev`.

Creating the Hotkeys

Now we are almost done. Of course, there are several ways to get it accomplished.

METHOD 1: Once our keyboard is actually generating keysyms from the Internet keys, we need to make them useful by associating them with some application by using *hotkeys*. Mandrake throws us a curve here: they use their own menuing system (*MenuDrake*) and not the KDE menu manager (*Kmenuedit*). The problem is that *MenuDrake* doesn't let us assign hotkeys. *Kmenuedit* will, but Mandrake won't use any menus *Kmenu* creates. Here's how we outfox this system. Use the Mandrake menu editor to create a new menu branch we will call Hotkeys. A right-click on the K-button and selecting "Preferences" should take you straight away to *MenuDrake*. If not, it's in the Configuration/Other menu. Create as many menu entries as you need. Name them as you like. See the screenshots for examples. What's in `~/bin/mute`? Here's a peek:

```
#!/bin/bash
let vol=`aumix -q|gawk '{vol=$3}'`
if [ $vol -eq "0" ]
then
aumix -L
else
aumix -v 0 fi
#thanks to bascule who wrote this script
```



Our `~/bin/mail` script looks like:

```
#!/bin/bash
/usr/bin/kmail --check
```


This opens *Kmail* and checks for new mail automatically!

METHOD 2: We can edit the `~/kde/share/config/khotkeysrc` file by hand if we understand the format. Open the file in a text editor; if no file exists, create one. Let's look at an example:

```
[Main]
Num_Sections=4
Version=1
[Section1] MenuEntry=true Name=K Menu -
Networking/WWW/konqbrowser.desktop
Run=Networking/WWW/konqbrowser.desktop
Shortcut=F16
[Section2] MenuEntry=true Name=K Menu -
Networking/Mail/KMail.desktop
Run=Networking/Mail/KMail.desktop
Shortcut=F23
[Section3] MenuEntry=true Name=K Menu -
Multimedia/Sound/Aumix.desktop
Run=Multimedia/Sound/Aumix.desktop
Shortcut=F32
[Section4] MenuEntry=true Name=K Menu -
Networking/WWW/Dillo.desktop
Run=Networking/WWW/Dillo.desktop
Shortcut=F25
```

Note that the "Num_Sections=" value must equal the number of sections, each of which are uniquely numbered. The `*.desktop` files are menu entries found in the `~/kde` file tree.

Loose Threads

We would very much like these Internet keyboard maps to be included by default since they are so popular. We would also like KDE to handle the XF86 keysyms (KDE3 does just that due to improvements in the *Khotkey* code). We would like to know why we can't get the **suspend** key to generate a keysym. And, it would also be nice to get *Kmail* to open in display :0.1 (our second head), but we haven't gotten that to work yet. But all that for another time. 

Kmenuedit can assign a hotkey to a new menu item. Press the Change button, select "Custom Key", press the desired Internet key. Here "mute" assigns F32. You must press "OK" to save. Now, pressing "mute" will execute the script in `~/bin` (you can place it anywhere convenient).

QUICK CODE

The art of the one-liner



How much power can you get in a single line of code?

Charlie Stross explores the art of the one-liner – and shows you some handy Linux power-tools

Perl is full of flexible commands – you can pack a lot of power into a single line of code. By way of an example, let's look at how we'd go about counting the number of words in a line of text. In a conventional procedural language like C, we'd iterate over each character in turn, checking to see whether it was a whitespace character; if it is, and the preceding character is non-whitespace, then we've reached a word boundary so we can add **1** to our count of words. In Perl, however, we use an array operator like **split()**, primed with a definition of a word boundary, and evaluate it in a scalar context: `print "there are ", split(/\s+/, $line), " words in [$line]\n";` **split()** scans a scalar variable (in this case **\$line**) looking for an expression (in this case the regular expression **/\s+/** – one or more whitespace characters). In an array context, it chops up the scalar and returns a list consisting of the pieces, using the expression to define the boundary between fragments. In a scalar context, it returns a count of how many pieces it found. You usually use **split** for tasks such as scanning files of records with fields separated by some regular delimiter character, but splitting a line of text into words is just fine.

Want to count the words in a file? Instead of calling **split()** on a single variable, we want to have it run in a loop, reading lines from standard input, and we want to add up the number returned each time we call **split()**. Like this:

```
while (<STDIN>) {
    $count += split(/\s+/, $_);
}
print $count;
```

Perl being Perl, we can leave out some of the details here and let Perl assume the default values. For example, if you don't give **split()** a scalar to scan, it assumes you want to scan **\$_**; if you don't give it a pattern to look for, it assumes you want to split on whitespace. And the **<STDIN>** construct (read a record from the named file handle) defaults to **STDIN** and sticks its output in **\$_** if you don't tell it anything else. All in all, this means we can reduce our count-the-words-in-the-standard-input program to:

```
while (<>) { $count += split; print $count, "\n";
```

And then run it from the shell prompt like this:

```
perl -e 'while (<>) { $count += split; print $count, "\n";' < some_file.txt
```

The **-e** (evaluate) option to Perl indicates that the following argument is a program to be executed – Perl can run commands specified this way, rather than stored in a named file.

Can we shorten this further? Sure: the **perlrun** manpage

shows us all the other juicy options Perl understands, e.g.:

```
perl -ane '$_ += $#F; print $c, "\n";'
```

The **-n** option makes Perl assume that your code (following the **-e** flag) is to be enclosed in a loop, reading from **STDIN**:

```
while (<STDIN>) {
    # your program goes here
}
```

-a (autosplit) flag tells Perl to automatically call **split(' ', STDIN)** on each line, stashing the results in a special variable called **@F**. You can use the **-F** flag to modify the pattern used to split fields:

```
perl -F'/W+/' -ane '$_ += $#F; print $c, "\n";'
```

This tells Perl to assume a loop around the program, automatically split each input line read from **STDIN** on the pattern **/W+/** (one or more non-word characters), and stick the results in **@F**. Our program then uses **\$#F** (count of highest subscript to **@F**) to increment **\$c**, a cumulative count of words, and print the result for each line. A little messier than our first attempt, but a lot shorter than **GNU wc (word-count)**, which weighs in at a couple of hundred lines of code in C!

Concord in one line

Counting words isn't the most immediately obvious use for Perl – after all, all GNU/Linux systems already have **wc**, right? But what if you want, for example, to count the frequency with which specific words occur in a file?

In a non-one-liner we'd do it like this:

```
while (<>) {
    @line = split(/\W+/);
    foreach (@line) {
        $concord{$_}++;
    }
}
while (($key, $value) = each %concord) {
    print "$key: $value\n";
}
```

%concord (concordance) is a hash; every time we meet a word in our input we use it as a key in **%concord** and increment the value associated with it (which therefore contains a running count of how many times we've met it). At the end we iterate over **%concord** and print out each word and the number of times we've met it.

This is a wee bit verbose. We can cut it down a bit:

```
perl -F'/W+/' -ane 'map { $c{$_}++ } @F;
END { while (($k, $v) = each %c) {
    { print "$k: $v\n"; }
}
```

It's still a one-liner (sort-of). We use an explicit **END {}** block to call out the instructions to be executed right before the script terminates – they won't be executed in the implicit loop block.

Here's a problem: the output from this program is a workable word-frequency list, but the words come out in no particular

order. We might want them sorted in dictionary order (by their first field), or on the basis of their frequency (the second field).

We can sort them by frequency at the shell prompt by daisy-chaining it in line with a program that will read its STDIN, and sort it on the basis of the second field. If we stash our concordance program into a Unix shell variable called **\$PROGGY**, then using a standard Unix pipeline we can and pipe them into sort:

```
$PROGGY < my_input_file.txt | sort -t : -k 2,
```

The **-t** flag tells *sort* to split fields on the **:** character, and the **-k** flag tells it to use the second and any subsequent fields (there aren't any) as the sort key.

The point to take away from this is that if you know the basics of Perl, you can roll one-liners to take the place of any random shell utility you happen to need but can't remember the name of.

Capital!

Want to capitalise all the filenames in a directory? Or convert them all to lowercase?

```
ls -l | perl -ane 'chomp; print "mv $_, uc($_), "\n";' | /bin/bash
```

This basically gets a list of filenames, one per line, using *ls*. Perl then bends these into commands of the form **mv filename.txt FILENAME.TXT** using the **uc()** (uppercase) command. Finally, we feed them to *bash*. For some reason there's no "uppercase rename" command in the Unix repertoire, but Perl one-liners let us add them easily enough.

A really cool thing to do with one-liners like this is to add them to your Unix shell environment. If you're using *bash* (or *Ksh*, or *Zsh*, or most contemporary shells) you can make use of aliases. A shell alias is set like this:

```
alias ln="ls -al $1 | less"
```

Thereafter, whenever you type the token **ln** as a command, separated by whitespace, the shell will expand the alias to **ls -al | less**, and **ln foo** will expand to **ls -al foo | less** (although not all shells support parameters to aliases – you may need to use a shell function instead).

Aliases and functions in *bash* can be loaded from the file `~/.bashrc`. So you can add a batch of Perl one-liners here. For example, add the following bits of bash/perl to your `.bashrc` file:

```
function ren_lower() {
    perl -ane 'chomp; print "mv $_, lc($_), "\n";'
}
function ren_upper() {
    perl -ane 'chomp; print "mv $_, uc($_), "\n";'
}
function ren_capitalise() {
    perl -ane 'chomp; print "mv $_, ucfirst($_), "\n";'
}
```

You can then (once you've executed your `.bashrc` or started a new subshell) use **ren_lower** in your shell scripts:

```
ls -l | ren_lower | bash
```

or

```
find . -type f -print | ren_capitalise | bash # make Windows users feel at home
```

Who sed what?

Perl is a working superset of the *awk* and *sed* text manipulation tools – we saw that in the capitalisation example above. But by using *sed*-style regular expressions we can achieve lots more.

For example, take ASCII files. On Unix and Linux, an ASCII file uses the `\n` (decimal 13, carriage return) character to signify the

end of a line. The Macintosh classic OS uses `\r` (decimal 10, carriage return) to signify the end of a line, and DOS/Windows uses a sequence `"\r\n"`, just to be different.

If you work with Mac or Windows users, you may want to add the following bash functions to your `.bashrc` to make swapping files containing ASCII text easier:

```
function to_mac () {
    perl -pi.bak -e 's/\n/r/g;' "$@"
}
function from_mac () {
    perl -pi.bak -e 's/r/n/g;' "$@"
}
function to_dos () {
    perl -pi.bak -e 's/\n/r\n/g;' "$@"
}
function from_dos () {
    perl -pi.bak -e 's/r//g;' "$@" }
}
```

In each case, the short script does a global search/replace on newline and carriage return characters, going in the appropriate direction (from Linux to Mac or DOS, or from DOS or Mac to Linux). We use the **-i.bak** flag to tell Perl to do in place editing – if you tell it to work on a file called *fred*, Perl will put its output into a file called *fred* and rename the original file to *fred.bak*. The **\$@** is a shell parameter that expands to a list of all the filenames specified as parameters to the shell function, so that if we say:

```
to_mac *.txt
```

The Perl script will be executed on every `*.txt` file in the current directory, re-creating them as Mac-compatible ASCII files and leaving the original versions with a `.txt.bak` suffix.

Perl's regular expressions can do a hell of a lot more than simply change line ending characters. For example, you can embed arbitrary Perl code in them using the **/e** operator:

```
$_ =~ s/foo/myfunc($1)/eg
```

Searches the variable **\$_** for the expression **foo**, and wherever it finds it, executes **myfunc()** (passing **foo** as a parameter).

This can come in handy if you've got a big website full of HTML files and want to manually update a link target. For example, if a lot of pages refer to an HTML page called "feedback.html" and you've just replaced it with the much-more-whizzy "guestbook.cgi" you can use Perl in combination with *find*, the Unix file search command:

```
find . -type f -name "*.html" -print | \
    perl -pi.bak -e "s/feedback.html/guestbook.cgi/g;";
```

Using an external *find* command is a pain in the neck. Why can't Perl do this itself? The answer is, it can – using **File::Find**. **File::Find** is a utility module for directory traversal and does pretty much what *find(1)* does – in fact, you can give the **find2perl** command a **find** command line and it will spit out Perl code using **File::Find** that does exactly the same work. It's not strictly suitable for one-liners, but it's nice to know that if you turn the above example into:

```
find2perl . -type f -name "*.html" -print | \
    perl -pi.bak -e "s/feedback.html/guestbook.cgi/g;";
```

Then there are no programs other than Perl involved in the loop.

(**File::Find** has one obvious deficiency – although you can use it to run other programs, there's no facility to add a callback subroutine in Perl that is executed on the *contents* of whatever it finds. So although we can execute the **s/foo/bar/g** command on the *name* of each file it locates, we need a second command (or a complex mess of file open/read/write/close code) to do it to the contents.)



LinuxFormatTutorialPerl

◀ Modules

In LXF 23 we met one of the most powerful Perl one-liners:

```
perl -MCPAN -e shell
```

The **-M** flag instructs Perl to import the named module (in this case **CPAN.pm**) and the **-e** command can then make use of anything imported from that module – in this case, the **shell()** (to search for and install modules interactively).

Some modules – notably **CPAN.pm** – export lots of subroutines that you can use in one-liners. For example, if you want to take a snapshot of your machine's Perl installation that you can replicate onto another computer, you'd do this:

```
perl -MCPAN -e autobundle
```

This writes a bundle file into the directory defined in **\$CPAN::Config->{cpan_home}/Bundle** on your system – typically in `~/cpan/Bundle` (in your home directory). A bundle file is a list of modules and version numbers, such as:

```
package Bundle::Snapshot_2002_04_03_00;

$VERSION = '0.01';
1;

__END__

=head1 NAME

Bundle::Snapshot_2002_04_03_00 - Snapshot of
installation on blueberry on Wed Ap

=head1 SYNOPSIS

perl -MCPAN -e 'install Bundle::Snapshot_2002_04_03_00'

=head1 CONTENTS

AnyDBM_File undef

Apache 1.27

Apache::Connection 1.00

Apache::Constants 1.09
```

Copy this bundle snapshot to another computer – then type:

```
perl -MCPAN -e "install
3 Bundle/Snapshot_2002_04_03_00.pm;"
```

And the CPAN module will do its second-best to install everything in the snapshot on your new machine. This is the recommended way of copying a Perl setup from one machine to another. (If you're feeling reckless, try **force install** to force CPAN to continue with the install if any of your modules fail to test out.)

Another handy module that gives us a bunch of commandline tools is **Config** – the Perl configuration module. **Config.pm** is basically an archive of system-specific information that describes everything the *Configure* program discovered when it was preparing to compile your local Perl installation; this is a detailed description of many aspects of your operating system. When you use *Config*, you implicitly import a hash called **%Config**, which contains a huge bundle of key/value pairs. Want to see a textual summary of the main configuration options on your system?

```
perl -MConfig=myconfig -e "print myconfig();"
```

Note the flag: **-MConfig=myconfig**. This is equivalent to:

```
use Config qw(myconfig);
```

Which explicitly imports **myconfig** from the module **Config**.

We can add other items to import explicitly:

```
use Config qw(myconfig config_vars config_sh);
```

is equivalent to:

```
perl -MConfig=myconfig,config_vars,config_sh
```

To get the entire configuration, try this:

```
perl -MConfig=config_sh -e "print config_sh();"
```

To get the Perl version:

```
perl -MConfig -e 'print $Config{version}; "\n";'
```

Or the system's identification (including operating system type, version, platform, CPU type, and when it was created):

```
perl -MConfig -e 'print $Config{myuname}; "\n";'
```

Want to know what the byte order (endian-ness) is on your system, or what name to use to invoke your C compiler?

```
perl -MConfig -e 'print $Config{byteorder}; "\n";'
```

```
perl -MConfig -e 'print $Config{cc}; "\n";'
```

Webbing around

LWP – **libwww-perl** – gives us a potload of utility modules, of which the easiest to use is **LWP::Simple**, a procedural front-end to the HTTP requests. For example:

```
perl -MLWP::Simple -e
'getstore("http://www.antipope.org/charlie/", "charlie.html");'
```

Fetches the URL "http://www.antipope.org/charlie/" and stores it in "charlie.html". And:

```
perl -MLPW::Simple -e "$x =
"http://www.antipope.org/charlie/"; \
print "Document $x is ", head($x)[1], " bytes long\n";"
```

Issues a **HEAD** request. This returns a list consisting of the object's MIME content-type, length, modification time, expiration time, and server.

We can have fun with the web and our `.bashrc` file – or other configuration files. For example, in `.bashrc`, add the following:

```
function getprint () {
    perl -MLPW::Simple -pe 'getprint($_); "$@"'
}
```

getprint() (in **LWP::Simple**) gets a URL and prints it on STDOUT (or, if it can't be retrieved, prints the status code and message on STDERR). This shell function expects to read a bunch of URLs as parameters, and fetches then prints them. So you can use a command like this:

```
getprint(http://www.linuxformat.co.uk/) > mypage-copy.html
```

On the bash command line to copy *Linux Format's* home page.

A bit more usefully, here's a nearly-one-liner that may be useful if you like to keep an eye on a bunch of web pages (such as slashdot, or the BBC news website):

```
function has_changed () {
    perl -MLWP::Simple -pe '$prefix = $ENV{HOME} . "/.urls"; \
    $mod = $_; $mod =~ s/\/+/g; \
    $time = scalar(stat "$prefix/$mod"); \
    $mt = head($_)[2]; \
    if ($mt > $time) { \
        print "CHANGED: $_\n"; \
        touch -m --date=$mt $prefix/$mod; \
    }
}
```

The idea here is that in your home directory (**\$ENV{HOME}**) you have a subdirectory called `"/.urls"`. In this directory you keep a bunch of files with names like `"http++slashdot.org+"`. (We change the forward-slashes to plus signs which are a bit easier for the Linux command line tools to handle). When **has_changed** is passed a series of URLs on standard input, it works out which file

Sorting

There's more to sort() than you realise

One topic that comes up repeatedly in Perl is: how do you sort a list of items?

Let's take an array of simple scalars like this:

```
my @list = (1, 2, 3, 4, 5, 6, 7, 8);
```

Each of these scalars is a simple integer number, and it's already sorted into ascending order. We can reverse it quite easily:

```
print reverse @list;
87654321
```

To insert some delimiters, we use **join** to turn **@list** into a scalar, with each element delimited by **||**, thus:

```
print "[",
  join("||", reverse @list),
  "];"
```

```
[8||7||6||5||4||3||2||1]
```

Now, let's randomise our list and run **sort** on it:

```
my @list = (7, 2, 3, 4, 1, 8, 5, 6);
```

```
print "[",
  join("||", sort @list),
  "];"
```

```
[1||2||3||4||5||6||7||8]
```

This demonstrates something important about Perl's **sort()** command. **sort** sorts numbers into ascending order, by default. You run it on a list, and it returns another list – sorted. Want to sort it the other way?

```
sort { $b <=> $a } @list;
```

```
[8||7||6||5||4||3||2||1]
```

The expression in curly braces is a sort **USERSUB** – a user-defined sorting test. **sort** works by iteratively applying a sorting test to each pair of variables in the array. The variables are visible within the sorting text as **\$a** and **\$b**; it's the job of the user sub to return **-1**, **0**, or **1** depending on how the elements in the list are to be ordered. The **<=>** operator, and its string equivalent **cmp**, are used to compare two values and return **-1**, **0**, or **1** depending on whether the left item is less than, equal to, or greater than the item on the right.

sort() assumes the following user-defined sorting test:

```
{ $a <=> $b }
```

If **\$a** is less than **\$b**, **<=>** returns **-1**; if it's greater, it returns **1**; and if they're the same it returns **0**. By swapping the order of **\$a** and **\$b** in the usersub (as in our sorting-it-the-other-way example above) we reverse the sorting order. Let's try another array:

```
my @list = (qw(red blue green mauve pink violet
yellow));
```

```
print ">=", join(" ", sort { $a cmp $b } @list);
```

```
=> blue green mauve pink red violet yellow
```

And the other way:

```
print ">=", join(" ", sort { $b cmp $a } @list);
=> yellow violet red pink mauve green blue
```

When sorting strings, it's important to remember that we don't sort in dictionary order – we sort in string order, depending on the local character set in use. (Undefined values sort before defined null strings, which sort before all characters in character-set order.) One side-effect of this is that **sort** is case-sensitive – and capitals go in a different sequence before lowercase letters, in ASCII/Latin-1:

```
my @list = (qw(red blue green MAUVE pink violet
yellow));
```

```
print ">=", join(" ", sort { $a cmp $b } @list);
```

```
=>MAUVE blue green pink red violet yellow
```

To avoid this, we need to canonicalise our comparison routine to lowercase:

```
my @list = (qw(red blue green MAUVE pink violet
yellow));
```

```
print ">=", join(" ", sort { lc($a) cmp lc($b) } @list);
```

```
=>blue green MAUVE pink red violet yellow
```

Arrays of simple scalars are easy enough to sort, but as we saw in the concordance example, sorting hashes so that they show up in the right order is a bit harder.

However, we can sort the keys to a hash on the basis of their associated value:

```
my @keys = keys %department_sales_volume;
sub sort_by_sales {
  $department_sales_volume{$b} <=>
  $department_sales_volume{$a};
}
for $dept (sort sort_by_sales keys
%department_sales_volume) {
  print "$dept: ", $department_sales_volume{$dept},
  "\n"; }
```

As you can see, here we're sorting an array (the keys to the hash) by comparing the value in the hash that's associated with each key. Once we've got the keys sorted into order on the basis of their values, we can print out a ranking of each department in our store.

Looking at our word-frequency program:

```
perl -F'/W+/ ' -ane 'map { $c{$_}++ } @F;
END { while (($k, $v) = each %c)
  { print "$k: $v\n"; } }
```

Let's re-write this and use **sort()** for our concordance to print in proper decreasing order of frequency:

```
#!/usr/bin/perl
while (<>) {
  chomp;
  @F = split(/\W+/);
  for $word (@F) {
    $frequency{$word}++;
  }
} # end of standard input
```

```
sub by_frequency {
  $frequency{$b} <=> $frequency{$a};
}

for $word (sort by_frequency keys %frequency) {
  printf ("%012s => %6d\n", $word,
$frequency{$word});
}
```

There are some much hairier uses of **sort()**, but we'll deal with them in another tutorial.

in .urls to look at (by putting **\$_** into **\$mod** and turning the slashes to plus signs), and checks its modification time (using the Perl command **stat()**). It then calls the **head()** method from **LWP::simple** and checks the modification time on the remote file. If the remote version is more recent, it prints "CHANGED" (followed by the URL) on standard output, then uses the external command **touch** to update the modification time on the file in **~/urls** to match the version on the web. In use, you just write a shell script in **bash** that uses **has_changed()** to detect whether a web page has been updated since the last time you ran it.

(NOTE: the date comparison method used here is primitive in the extreme. Doing this job properly isn't really a one-liner; if you want a fun exercise, try turning **has_changed()** into a real standalone Perl program with error checking, the ability to take account of different time zones and handle failed connections gracefully, and use a DBM database file instead of a subdirectory.) **LWP** isn't the only module for interesting web-related one-liners:

```
perl -MNet::Ping -e '$p = Net::Ping->new("tcp", 60); \
  $p->{port_num} = 80; \
  $p->ping("www.antipope.org") && \
```

```
print "www.antipope.org has a live http
port\n";'
```

Net::Ping lets us create ICMP, UDP or TCP packets and bounce them off remote hosts. In this example, we create a new **Net::Ping** object with a 60 second timeout, tell it to send TCP packets to port 80 (the standard well-known port for HTTP), and then **ping** **www.antipope.org** to see if it's alive and responding on that port. (You can actually use **Net::Ping** to write a dumb but easy-to-understand port scanner; just loop through all the available ports:

```
perl -MNet::Ping -e '$p = Net::Ping->new("tcp", 60); \
  for ($i = 0; $i <= 65535; $i++) { \
    $name = (getservbyport($i, "tcp"))[3]; \
    $p->ping("www.antipope.org") && \
    print "www.antipope.org has a live $name
port\n"; \
  }'
```

getservbyport() is a built-in Perl command that, for a given port number and protocol returns (in array context) the corresponding service name, aliases, port number, and protocol name. [LWP](#)

PARSING XML WITH SAX

Speaking Java

Learn how to read XML documents in Java with this introduction to SAX by **Richard Drummond**.



XML is the latest in a long line of buzz technologies that the pundits proclaim to be a cure for all the problems of software engineering. Or at least you could be led to believe that by the number of column-inches devoted to XML in the media and the number of vendors shipping products with XML support. But what is XML? How can such a simple three-letter initialism provoke so much excitement?

XML (the eXtensible Markup Language) is nothing more than a language for describing structured data in an open, human-readable and portable way. The beauty of XML is that it specifies the syntax for marking up data, but not the grammar. Therefore XML can be applied to any problem domain: you simply have to define an appropriate grammar to describe your data.

Java has two standard APIs for parsing XML: SAX (the Simple API for XML) and DOM (Document Object Model). Both of these APIs are just that: APIs. They provide a framework for driving an XML parser; they do not provide the parser itself. (This is rather like the way JDBC works. JDBC provides the framework for talking to an SQL server, but you need a driver for your specific database.) Luckily, the Java developer is spoilt for choice when it comes to choosing an XML parser (see the box *XML parsers for Java*) and the majority ship with built-in support for both SAX and DOM. In fact, with the release of *J2SE1.4*, Sun have integrated support for XML into the standard class library and both APIs are included, unified under the JAXP (Java API for XML) umbrella. The parser in *J2SE1.4* is *Apache's Crimson*.

SAX and DOM differ in approach. DOM parses a complete document, and builds a tree of Java objects to represent the data found there. SAX, on the other hand, provides an event-based API and parses XML sequentially, generating a stream of events as it encounters the tags and data in the document it is parsing. Your program can listen for these events, and so interpret the content of the document, by registering various handlers with the SAX parser. DOM is therefore the easiest to use, but has higher overheads in time and memory. SAX is fast, but requires that you do more work yourself. For parsing large data files, SAX is generally a better option, and is the one we'll discuss here.

Using SAX

The first step to using SAX is to instantiate a parser, and there are several ways of doing this. Remember at this point, though, that

SAX is a generic interface to XML parsers and that you want to make use of this to make your code independent of any particular parser implementation. The best way to create a SAX parser is with the class method `createXMLReader()` from the `org.xml.sax.helpers.XMLReaderFactory` class. You can pass the name of a parser's SAX2 driver class here or call this method with no arguments to instantiate the system default parser – as specified in the `org.xml.sax.driver` system property.

Have a look at this excerpt: the `main()` method of our example (have a look at the box, *The Project*).

```
public static void main( String[] args )
    throws IOException, SAXException
{
    // create the parser
    XMLReader parser = XMLReaderFactory.createXMLReader();

    // create the event handler
    IndexParser1 handler = new IndexParser1();

    // attach our event handler
    parser.setContentHandler( handler );

    // wrap up the input file
    InputSource input = new InputSource( openFile( args[0] ) );

    // ... and go.
    parser.parse( input );
}
```

This shows the basic procedure for using a SAX parser. First we instantiate the parser, then the event handler, then we register the event handler to the parser, and invoke `parse()` to process the specified document. (We'll talk about `InputSource` below.)

Here we call `createXMLReader()` with no arguments to create an instance of the default parser, which is returned as an instance of `XMLReader`. (Note that this has nothing to do with the `java.io` package's `Reader` classes.) If no default parser has been set or Java can't find it, then a `SAXException` will be raised.

If your Java runtime doesn't set a default SAX parser, you can do this manually on execution with, for example:

```
java -Dorg.xml.sax.driver=gnu.xml.aelfred2.SAXDriver
    IndexParser1
```

XML Parsers for Java

Where to find XML parsers, and the classes you need to instantiate them.

Name	Homepage	SAX2 driver class
Aelfred2	http://www.gnu.org/software/classpathx/jaxp/jaxp.html	gnu.xml.aelfred2.SAXDriver (non-validating) gnu.xml.aelfred2.XmlReader (optionally validating)
Crimson	http://xml.apache.org/crimson/index.html	org.apache.crimson.parser.XmlReaderImpl
Xerces2 Java	http://xml.apache.org/xerces2-j/index.html	org.apache.xerces.parsers.SAXParser

This tells SAX to use the GNU **Classpathx** parser (providing that parser is in your class path). See the table in the box *XML Parsers for Java* for the driver classes for popular parsers.

The big event

Your code can listen to SAX2 events generated by the parser by using a call-back mechanism. The SAX API defines several interfaces which you can implement in your code to hear about specific types of SAX2 event. For example, the principal interface is **ContentHandler**, which includes call-back methods that get invoked by the parser according to the content of the XML document being processed. One such method declared by the **ContentHandler** interface is the **startElement()** method, which gets invoked when the parser reads an opening tag; another is **characters()** which is invoked when data between a pair of opening and closing tags is parsed. SAX2 also declares the **ErrorHandler** interface to handle error events, the **DTDHandler** interface to handle DTD events and the **EntityResolver** interface to handle references to external entities.

Making life easier, the **org.sax.xml.helpers** package contains the class **DefaultHandler**. This is a convenience class which implements all four event interfaces. We can simply sub-class **DefaultHandler** and override the methods corresponding to the events we are interested in. This is the route taken in our example. Here is how we override the **startElement()** method:

```
public void startElement( String uri, String name, String
qName, Attributes attributes )
{
    // Output element name
    System.out.println( "Element: " + name );

    // Output attributes (if any)
    if( attributes.getLength() > 0 ) {
        System.out.print( "Attrs :";
        for( int i = 0; i < attributes.getLength(); i++ )
            System.out.print( " " + attributes.getValue( i ) );
        System.out.println();
    }

    // Output blank line after parsing an <indexedfiles> tags
    if( name.compareTo( "indexedfiles" ) == 0 )
        System.out.println();
}
```

When the **startElement()** method is invoked by the parser, it is passed the **name** of the tag being parsed and any **attributes** associated with it. The other parameters **uri** and **qName** are for handling XML namespaces, and we'll ignore them for just now. In the above method, we simply output the tag name to the console and walk through the list of attributes, printing the value of each.

To receive notification of the data associated with a tag, we implement the **characters()** method. The data read is passed by the parser as a character array, but only a sub-section of this array may be valid for the current invocation: the relevant sub-string is specified by the index **start** and the **length** parameters.

Here's the extract from our example

```
public void characters( char[] ch, int start, int length ) throws
SAXException
{
    // Output element data
    System.out.println( "Data: " + new String( ch, start, length ) );
}
```

The project

Or how come we're talking about this XML stuff now?

Nick and I are developing some tools for creating indices for the *Linux Format* coverdiscs and browsing them, and I thought it would be neat to provide a searchable index browser as a Java applet to include in the HTML front-end of the coverdiscs. We opted for XML as a data format simply because of the wide availability of tools to parse and munge XML. It was at this

point I remembered that we'd been wanting to do a tutorial on Java and XML for some time. The accompanying code on the coverdisc is a rough example which will parse an *LXF* coverdisc index. It's just an example at the moment, though, and simply prints out the data contained in the file, rather than doing anything useful with it. Wait for next month for that...

Note that the **characters()** method may be invoked multiple times per tag-pair – the SAX2 API doesn't specify that all the data should be passed in one go. For the XML fragment

```
<myTag>This is some data.</myTag>
```

the parser could generate multiple character events, passing the data in chunks as 'This is ', 'some' and 'data'. It could even generate events for each character of data. The upshot is that you shouldn't expect the data in one piece. A tactic for dealing with this is to cache the data from each character event and assemble them into a string when the relevant **closeElement()** event is received. We'll go into this in more detail next time, when we look at how we can make use of parsed XML data.

Getting input

The final piece of the puzzle is how to specify the document you wish to parse. The parser's **parse()** method is overloaded: one version takes the URL of the document as a **String**; the other expects the document to be wrapped up as an **InputSource**.

The **InputSource** class encapsulates information about the source of an XML document, and has several constructors. You can pass a reference to the document as a URL in a **String**, or provide it as an **InputStream** or **Reader**. These last two are the most useful – thanks to the flexibility of the **java.io** package.

Java provides us with the tools to instantiate an **InputStream** from a file, a URL, a pipe or even an in-memory character array. Thus, with the **InputSource** class, we can use all these types of stream as the source for our XML data.

In our example we want to handle XML stored locally as a file and remotely (specified by a URL), and we implement the following helper method to create the necessary **InputStream** for us. Usefully, this code also automatically handles gzip'd files.

```
private static InputStream openFile( String file ) throws
IOException
{
    // Use filename extension to determine if the file is gzip'd
    boolean isZipped = file.endsWith( ".gz" );

    // Construct reference URL from current directory
    URL ref = new File( System.getProperty( "user.dir" ) ).toURL();

    // Open file using ref as a context
    InputStream instr = new URL( ref, file ).openStream();

    // Handle gzip'd files
    if( isZipped ) instr = new GZIPInputStream( instr );

    return instr;
}
```

More on XML next month. [LXF](#)

NEXT MONTH

Next issue we will look at how you can make use of the data parsed by SAX rather than just dumping it on the console and how to map XML data in a file to Java objects in memory.



DRAWING WITH TCanvas

Graphics programming

PART 10: This month **Brian Long** takes you on a tour of the **TCanvas** class to see how you can do simple graphics in your Kylix applications

Graphics are an important part of many applications but so far we have only looked at adding various controls to *Kyl* forms. This month we turn our attention to doing a bit of drawing. Clearly there are various levels we could take this. Advanced graphics and rendering can be achieved using OpenGL but we will start things off looking at the humble **TCanvas** object.

The TCanvas class

A canvas object is used in objects that need to render an image, so a form has a canvas as does a **TPaintBox** component and a **TBitmap** object (among many others).

The canvas offers a variety of methods for drawing lines, rectangles, ellipses, chords, circle segments, polygons, Bezier curves and for writing text. It uses a pen object to draw lines and shape boundaries, a brush for filling areas with a colour or pattern and a font for writing text. The pen, brush and font objects have their own properties to customise their attributes and they themselves are exposed through **TCanvas** properties.

Flag waving

To get something straightforward out of the **TCanvas** class our first example will be to draw the Union Jack on the background of the form. You'd be forgiven if you thought that this would be quite straightforward (a couple of differently coloured crosses laid on top of each other) but the flag has very particular specifications that give it the correct look. We found information on how to

accurately draw the flag at: <http://www.jdawiseman.com/papers/union-jack/union-jack.html>.

We'll use code that produces a faithful rendering of a Union Jack but first we should see how to use the canvas. Whenever you need to update an item that has a canvas you can simply set the relevant properties and call the appropriate methods of the **TCanvas** object available through the **Canvas** property to draw whatever you need. For example, this draws a purple rectangle with a red border from point (10,10) to point (300,300):

```
Canvas.Pen.Color := clRed;
Canvas.Brush.Color := clPurple;
Canvas.Rectangle(Rect(10, 10, 300, 300));
```

In the case of a form or a paint box component you need to update the canvas in response to the **OnPaint** event, which indicates it needs redrawing. *CLX* sends the **OnPaint** event whenever a section of the form becomes invalid, such as restoring a minimised form or moving a window from in front of the form. You can see what area actually needs redrawing by checking the **ClipRect** property, potentially allowing you to optimise the form re-drawing and making the application more responsive.

Correspondingly our Union Jack will be rendered on a form's canvas by its **OnPaint** event:

```
uses
  Math, UnionJack;

procedure TForm1.FormPaint(Sender: TObject);
var
  R: TRect;
begin
  if ClientHeight * ClientWidth = 0 then Exit;
  //Fill form
  Canvas.Brush.Color := Color;
```

```

Canvas.FillRect(ClientRect);
//Ensure largest Union Jack of correct aspect ratio (2:1) is
drawn
if ClientWidth < ClientHeight * 2 then
  R := Rect(0, 0, ClientWidth, ClientWidth div 2)
else
  R := Rect(0, 0, ClientHeight * 2, ClientHeight);
OffsetRect(R, (ClientWidth - R.Right) div 2, (ClientHeight -
R.Bottom) div 2);
DrawUnionJack(Canvas, R);
end;

```

The **OnPaint** event handler first checks to see if the form has been narrowed down to nothing (**ClientWidth** would be **0**) or shortened to nothing (**ClientHeight** would be **0**). If either of these is true there is nothing to do so **Exit** is used to leave the event handler.

When we get round to drawing the Union Jack it needs to maintain a fixed aspect ratio (the width must be twice the height). Since the form can be resized to any proportions we will draw the largest flag possible in the centre of the available space. This means that as the form is resized there may be some blank space above and below the flag or to the left and right of it. The blank space will need to be refilled to erase parts of previously drawn flags of different dimensions, so we achieve this by drawing a filled rectangle over the entire form before starting to draw the flag.

The code sets the brush colour of the form's canvas to be the form's colour and then uses the brush to fill a rectangle that covers the form's client area (the **ClientRect** property returns this information as a **TRect**). **FillRect** does not draw the edges of the rectangle using the pen colour.

Next it moves on to identify the largest rectangle with the appropriate aspect ratio that can be drawn on the form; this is stored in the **TRect** record variable **R** set up by the **Rect** function. Once the size is identified the **TRect** is adjusted to ensure the rectangle is placed centrally in the form.

The final job is to pass the form's canvas and the size rectangle to the **DrawUnionJack** procedure, found in a separate unit called **UnionJack.pas**. **DrawUnionJack** is responsible for drawing a Union Jack on the specified canvas in the specified rectangle. It doesn't verify that the rectangle has the correct proportions but the **OnPaint** handler ensures this. The information shown on the aforementioned website explains the appropriate relationships between the placement and size of all the markings and the width and height of the flag.

After a pencil-and-paper session involving head scratching and strained attempts to recall some elementary trigonometry we came up with some code to do the job. Each part of the flag is drawn in turn using appropriate canvas operations and simple arithmetic, but there are a couple of trigonometric calculations that are required each time. To avoid repeating these unnecessarily, they are calculated in advance (in the unit initialisation section) and assigned to variables that are private to the UnionJack unit.

The code looks like this, but sections of it have been snipped for brevity. You can see the entire code in the *SimpleGraphics.dpr* project on this month's disc.

```

var
  Theta, SinTheta, Theta2, SinTheta2: Double;

procedure DrawUnionJack(Canvas: TCanvas; ARect: TRect);

```

```

var
  L, T, R, B, H, W,
  ThirdHeight, TwoThirdsHeight,
  VertOfs, HorzOfs, HorzOfs2: Integer;
begin
  L := ARect.Left;
  T := ARect.Top;
  B := ARect.Bottom;
  R := ARect.Right;
  H := B - T;
  W := R - L;
  ThirdHeight := T + H div 3;
  TwoThirdsHeight := T + H * 2 div 3;
  with Canvas do
  begin
    //Blue background
    Brush.Color := RGB(0, 51, 102);
    FillRect(ARect);

    //Border for the crosses of St. Andrew and St. Patrick
    Pen.Color := clWhite;
    Brush.Color := clWhite;
    //Get horizontal offset of border
    HorzOfs := Round(H / (10 * SinTheta));
    //Get vertical offset of border
    VertOfs := Round(H / (10 * SinTheta2));
    //Draw top left to bottom right diagonal border
    Polygon([
      Point(L, T), Point(L + HorzOfs, T), Point(R, B - VertOfs),
      Point(R, B), Point(R - HorzOfs, B), Point(L, T + VertOfs),
      Point(L, T)]);
    //Draw bottom left to top right diagonal border
    Polygon([
      Point(L, B), Point(L + HorzOfs, B), Point(R, T + VertOfs),
      Point(R, T), Point(R - HorzOfs, T), Point(L, B - VertOfs),
      Point(L, B)]);

    //Red cross of St Patrick
    ...

    //Border for the red cross of St. George
    //White vertical strip
    Brush.Color := clWhite;
    FillRect(Rect(L + W * 5 div 12, T, L + W * 7 div 12, B));
    //White horizontal strip
    FillRect(Rect(L, T + H div 3, R, T + H * 2 div 3));

    //The cross of St. George
    //Red vertical strip
    Brush.Color := RGB(204, 0, 51);
    FillRect(Rect(L + W * 9 div 20, T, L + W * 11 div 20, T + H));
    //Red horizontal strip
    FillRect(Rect(L, T + H * 2 div 5, R, T + H * 3 div 5));
  end
end;

initialization
//Get one diagonal angle, where width:height = 2:1
Theta := ArcTan(1/2);
SinTheta := Sin(Theta);

```



Figure 1: The flag background set up.

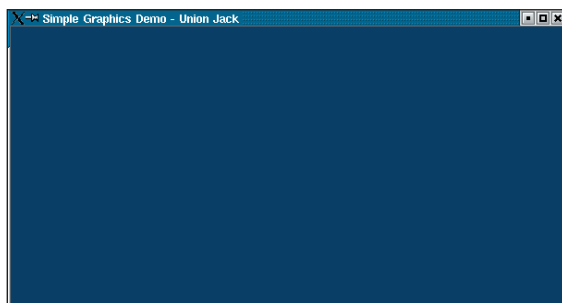


Figure 3 (far right): St. Patrick and St. Andrew taken care of.



Figure 2: The diagonal borders in place.



Figure 4 (far right): The border for St. George's cross.

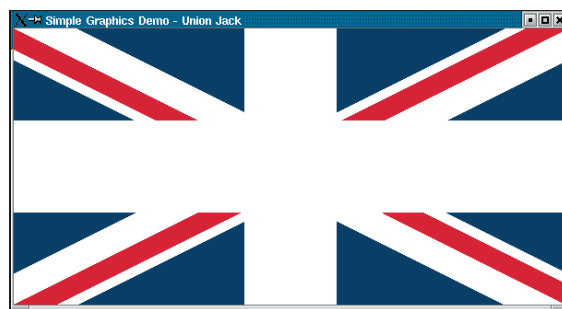


Figure 5 (right): The finished item.

```

<< //Get other angle
    Theta2 := ArcTan(2);
    SinTheta2:= Sin(Theta2);
end.

```

You can see the code declares some additional local variables for the left, top, right, bottom, width, height (as well as 1/3 and 2/3 of the height) of the bounding rectangle to allow briefer expressions to be written. Since most of the code involves calls to **TCanvas** methods or accessing **TCanvas** properties, a **with** statement is used to enter the scope of the canvas object before starting on the drawing code.

We'll take the drawing code a section at a time (the comments in the source should help you navigate it). Firstly the blue background is achieved by filling the entire rectangular area with an appropriate colour. This is specified to be Pantone 280, which equates to a **TColor** value with no red content, but with a green value of **51** and a blue value of **102**. **TColor** values can be created easily by the RGB helper function. As you can see, a **TColor** has the red value in the low byte, then the green value and then the blue value.

```

function RGB(R, G, B: Byte): TColor;
begin
    Result := TColor(R or (G shl 8) or (B shl 16));
end;

```

With not much done yet, all we have is the background of the flag (see **figure 1**). The next job is to get the thick, white diagonals which act as the border for the crosses of St. Andrew (a diagonal white cross, conveniently the same colour as the border) and St. Patrick (a diagonal red cross).

The pen and brush are set to white for this job and then some calculations precede the plotting of the required polygons. Polygons are drawn by supplying an array of **TPoint** records that indicate the vertices of the polygon. The Point function is helpful in creating a **TPoint** from a pair of X and Y co-ordinates. If you look at the result in **figure 2** you can see that each diagonal has six vertices, but the Polygon method requires one of points to be specified twice, once as the first point and once again as the last point.

Inside these diagonal borders are placed the crosses of St.



Andrew and St. Patrick (and later the cross of St. George will be drawn over everything else).

These two crosses are interleaved in an odd way if we look at it (see **figure 3**). If you examine the paths of the red sections you will see they are not symmetrical. The idea is that in each corner within the white border you find a section of a red cross and a section of a white cross (which will obviously not be explicitly visible). At either end of each diagonal the relative positions of these crosses are swapped, which leads to some messy drawing code. If they were not swapped we would be able to just draw two diagonal red strips, but as it is we have to do four individual sections

Each red quadrilateral must be drawn individually to get the required layout. Also, the official colour is Pantone 186, which is approximated using an RGB colour with a red value of **204** and a blue value of **51**.

With two of the crosses out of the way, the border for St. George's cross can be drawn. This is just a pair of rectangles and gives **figure 4**.

The final job is to draw in the St. George cross itself, involving nothing more than two further filled rectangles.

If you run the program you will find the form is resizeable and as you resize, the Union Jack is drawn as large as possible keeping the correct aspect ratio.

Optimising the drawing

The problem with this approach to drawing is that each individual change to the canvas is drawn on-screen individually, causing

many screen updates. In this case eleven things happen on-screen each time the flag is drawn (background fill, two diagonal white strips, four diagonal red quadrilaterals, two white strips and two red strips). With a more complex image things would get very sluggish and flickery.

To overcome this problem it makes sense to use a background bitmap (an off-screen bitmap). This is a **TBitmap** object that represents the image we are creating. The bitmap object is an in-memory matrix of pixels that we can set up using its canvas object. When the image is complete we can draw that on the form causing only a single screen update.

Another issue is the excess drawing done in this example, thanks to the form's **OnPaint** event handler executing a **FillRect** call to fill the entire client area. There was good reason for this statement – it was designed to fill the gaps left when the form is resized and the flag doesn't occupy the whole client area. However it does mean that a lot of pixel plotting takes place pointlessly (most of this fill area is overdrawn by the flag).

Additionally, the form takes responsibility for ensuring the correct aspect ratio is maintained, and that really should belong in the flag drawing code.

SimpleGraphics2.dpr takes all these things into account. The form has a **TBitmap** defined as a private data field, which is constructed and destroyed in the form's **OnCreate** and **OnDestroy** event handlers respectively. The **OnPaint** event handler simply sets the bitmap to be the same size as the form's client area and then draws a flag on it. The bitmap object is then drawn on the form.

```
type
  TForm1 = class(TForm)
  ...

private
  Bmp: TBitmap;
end;

...

procedure TForm1.FormCreate(Sender: TObject);
begin
  Bmp := TBitmap.Create;
end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  Bmp.Free;
end;

procedure TForm1.FormPaint(Sender: TObject);
begin
  if ClientHeight * ClientWidth = 0 then Exit;
  Bmp.Height := ClientHeight;
  Bmp.Width := ClientWidth;
  DrawUnionJack(Bmp.Canvas, ClientRect);
  Canvas.Draw(0, 0, Bmp);
end;
```

This project uses a unit called **UnionJack2.pas**, which is much the same as the original unit but contains a few updates. Firstly a new routine has been added that checks the aspect ratio of a rectangle passed in using a **TRect**. If the ratio is 2:1 as it

should be the routine returns **True**. If not it returns **False** and sets up more information.

```
function CheckAspectRatio(const SrcRect: TRect; out FlagRect,
  GapRect, Gap2Rect: TRect): Boolean;
var
  W, H: Integer;
begin
  Result := True;
  W := SrcRect.Right - SrcRect.Left;
  H := SrcRect.Bottom - SrcRect.Top;
  FlagRect := SrcRect;
  GapRect := Rect(0, 0, 0, 0);
  Gap2Rect := GapRect;
  if W <> H * 2 then //Check for incorrect aspect ratio
  begin
    Result := False;
    if W < H * 2 then //too tall
    begin
      FlagRect.Top := (H - (W div 2)) div 2;
      FlagRect.Bottom := FlagRect.Top + W div 2;
      GapRect := Rect(SrcRect.Left, SrcRect.Top, SrcRect.Right,
        FlagRect.Top);
      Gap2Rect := Rect(SrcRect.Left, FlagRect.Bottom,
        SrcRect.Right, SrcRect.Bottom);
    end
    else //too wide
    begin
      FlagRect.Left := (W - (H * 2)) div 2;
      FlagRect.Right := FlagRect.Left + H * 2;
      GapRect := Rect(SrcRect.Left, SrcRect.Top, FlagRect.Left,
        SrcRect.Bottom);
      Gap2Rect := Rect(FlagRect.Right, SrcRect.Top,
        SrcRect.Right, SrcRect.Bottom);
    end
  end;
end;
```

CheckAspectRatio defines three out parameters (they are much the same as **var** parameters but are used *only* to pass information out of a routine) that return the biggest rectangle the flag can occupy in the correct aspect ratio and the gaps either side, if there are any. The flag drawing routine can use this information to more efficiently paint its image on the canvas and fill in the rest of the area.

The start of the modified routine is shown below. As you can see, an additional background colour parameter is defined to specify the colour to fill in the extra space, and it has a default value of **clBlack** to save you specifying the colour unless you want something different.

```
procedure DrawUnionJack(Canvas: TCanvas; ARect: TRect;
  BkCol: TColor = clBlack);
var
  FlagRect, GapRect, Gap2Rect: TRect;
  GapsToFill: Boolean;
  L, T, R, B, H, W,
  ThirdHeight, TwoThirdsHeight,
  VertOfs, HorzOfs, HorzOfs2: Integer;
begin
  GapsToFill := not CheckAspectRatio(ARect, FlagRect,
    GapRect, Gap2Rect);
  //Set up brief access variables
  L := FlagRect.Left;
```



Figure 7 (right):
A much better tiled
effect.



```
T := FlagRect.Top;
B := FlagRect.Bottom;
R := FlagRect.Right;
H := B - T;
W := R - L;
ThirdHeight := T + H div 3;
TwoThirdsHeight := T + H * 2 div 3;
```

```
with Canvas do
begin
  Brush.Color := BkCol;
  if GapsToFill then
  begin
    FillRect(GapRect);
    FillRect(Gap2Rect)
  end;
  //Blue background
  Brush.Color := RGB(0, 51, 102);
  FillRect(FlagRect);
  ...
```

This gives a better effect at run-time with each redraw being somewhat quicker (noticeable if you quickly resize the form larger and smaller).

Tiling

If we don't want a single flag (or any bitmap for that matter) but instead want an image tiled across our form, then we can also readily accommodate this. *SimpleGraphics3.dpr* achieves this by setting up the flag in an off-screen bitmap in advance.

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  BkGrndBmp := TBitmap.Create;
  BkGrndBmp.Width := 200;
  BkGrndBmp.Height := 100;
  DrawUnionJack(BkGrndBmp.Canvas, Rect(0, 0,
  BkGrndBmp.Width, BkGrndBmp.Height));
end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  BkGrndBmp.Free;
end;
```

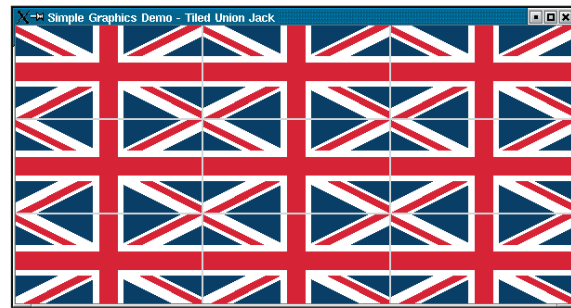
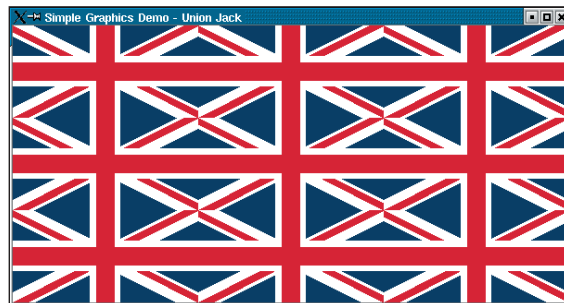
The **OnPaint** event is responsible for tiling this bitmap across the form whenever the form needs redrawing. This can be done very conveniently with the **TiledDraw** canvas method:

```
Canvas.TiledDraw(ClientRect, BkGrndBmp);
```

The downside with this method is that it offers no option to insert a small gap between each image (see **figure 6**).

So instead, the tiling code logic is implemented by hand and gives the result in **figure 7**:

Figure 6: A tiled bitmap,
but with no gaps...



```
procedure TForm1.FormPaint(Sender: TObject);
var
  X, Y: Integer;
const
  XGap = 2;
  YGap = 2;
begin
  //Tile the Union Jack across the form
  Y := 0;
  while Y < ClientHeight do
  begin
    X := 0;
    while X < ClientWidth do
    begin
      Canvas.Draw(X, Y, BkGrndBmp);
      Inc(X, BkGrndBmp.Width + XGap);
    end;
    Inc(Y, BkGrndBmp.Height + YGap);
  end;
end;
```

Scribble app

Of course we can use these principles to provide us a doodling opportunity (by way of a simple scribbling/drawing program). The project we are working towards is *Scribble.dpr* on the disc. It lets the user choose a pen width and colour and doodle over the form. A background bitmap is used to ensure the picture doesn't disappear when the form is covered (even by parts of its own menu).

The important parts of the code are the mouse event handlers (**OnMouseDown**, **OnMouseUp** and **OnMouseMove**), but let's first check the form events (**OnCreate**, **OnDestroy** and **OnPaint**).

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  DrawingBmp := TBitmap.Create;
  DrawingBmp.Width := Screen.Width;
  DrawingBmp.Height := Screen.Height;
  DrawingBmp.Canvas.Pen.Width := 1;
  DrawingBmp.Canvas.Pen.Color := clRed;
end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  DrawingBmp.Free;
end;

procedure TForm1.FormPaint(Sender: TObject);
begin
```

```
Canvas.CopyRect(Clip, DrawingBmp.Canvas, ClipRect)
```

```
end;
```

When the form is created a background bitmap is created as large as the screen (to cater for the form being resized or maximised) and the pen of its canvas is set to have a default width of **1** and a red colour. The **OnDestroy** handler ensures the bitmap is tidied away.

The **OnPaint** handler copies a portion of the background bitmap onto its canvas, as large as the form's exposed client area (there's no point copying pixels that won't be drawn). However you could alternatively write:

```
Canvas.Draw(0, 0, DrawingBmp)
```

Now let's see what the mouse handlers do. The user starts drawing by pressing the left mouse button down, so if that happens a form data field (**Drawing**) is set to **True** to indicate drawing is happening. Additionally the **LastPos** field is set to the current mouse location. The **Drawing** flag is reset to **False** if the user releases the left mouse button.

When the mouse is moved whilst drawing is ongoing, the cursor is moved back to the last recorded mouse position and then a line drawn over to the current mouse position. The new position is recorded in the **LastPos** field so we know where to move back to then time the mouse is moved.

```
type
```

```
TForm1 = class(TForm)
```

```
...
```

```
private
```

```
DrawingBmp: TBitmap;
```

```
Drawing: Boolean;
```

```
LastPos: TPoint;
```

```
end;
```

```
...
```

```
procedure TForm1.FormMouseDown(Sender: TObject; Button:
TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
if Button = mbLeft then
```

```
begin
```

```
Drawing := True;
```

```
LastPos := Point(X, Y)
```

```
end
```

```
end;
```

```
procedure TForm1.FormMouseUp(Sender: TObject; Button:
TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
if Button = mbLeft then
```

```
Drawing := False
```

```
end;
```

```
procedure TForm1.FormMouseMove(Sender: TObject; Shift:
TShiftState; X,
```

```
Y: Integer);
```

```
begin
```

```
if Drawing then
```

```
begin
```

```
DrawingBmp.Canvas.MoveTo>LastPos.X, LastPos.Y);
```

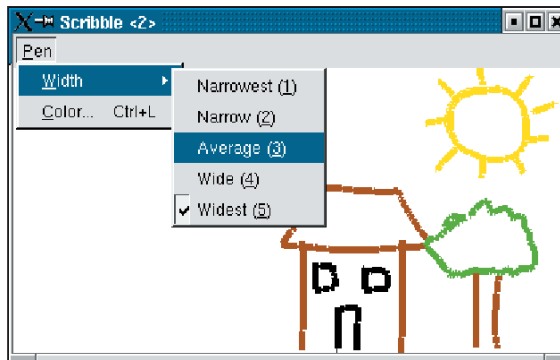


Figure 8: A doodle app.

```
LastPos := Point(X, Y);
```

```
DrawingBmp.Canvas.LineTo>LastPos.X, LastPos.Y);;
```

```
Invalidate;
```

```
end
```

```
end;
```

The form has a simple menu on it to adjust the pen width and colour. The colour menu item invokes a colour selection dialog to let the user choose any supported colour. The width menu item is a submenu with five menu items within it, allowing various widths to be chosen (see **figure 8**).

Since they act as a set of mutually exclusive items (like radio buttons) each of them had its **GroupIndex** set to **1** so they all reside in a group. Additionally their **RadioItem** properties were set to **True** and they all share the same event handler (**PenWidthClick** in the code below).

```
procedure TForm1.Color1Click(Sender: TObject);
```

```
begin
```

```
dlgColor.Color := DrawingBmp.Canvas.Pen.Color;
```

```
if dlgColor.Execute then
```

```
DrawingBmp.Canvas.Pen.Color := dlgColor.Color
```

```
end;
```

```
procedure TForm1.PenWidthClick(Sender: TObject);
```

```
begin
```

```
with Sender as TMenuItem do
```

```
begin
```

```
DrawingBmp.Canvas.Pen.Width := Tag;
```

```
Checked := True
```

```
end
```

```
end;
```

Each menu item's **Tag** property is set to the value that needs to be assigned to the pen width (**1** to **5** in this case). The event handler assigns **Tag** to the pen width and then checks the selected menu item. **RadioItem** being **True** ensures that when one of the menu items' checked properties is set to **True** all the others are set to **False**.

It is clear that this is a very minimalist application and it is left as an exercise for the reader to add more interesting features, such as an option for starting a new doodle and options for drawing lines and circles and rectangles. You can also offer to save the bitmap to a file chosen from a save dialog (check out the **TBitmap** methods).

Summary

We have now looked at some simple drawing operations using the **TCanvas** object. You can see more drawing in the *Clock* demo that is installed along with *Kyl* (look in *Kyl*'s demos directory). [LXF](#)

About Brian Long

Brian Long is a UK-based freelance trainer and problem solver for Borland's *Kyl*, *Delphi* and *C++Builder* packages. His website is at www.blong.com and he can be emailed at brian@blong.com

NEXT MONTH

Next month we'll look at some simple *CLX* animation. In the meantime, if there is something about *Kyl Open Edition* you want to see covered here, drop us an email and we'll try our best to incorporate it into a future instalment.

OUTWITTING SCRIPT ATTACKS

An introduction to CGI programming

PART 2: Dave Cross considers security – so should you. Protect your site from scripting attacks.

Running a CGI program on a web server that is connected to the Internet is actually quite a brave thing to do. You're giving anyone with an Internet connection permission to run a program on your server. You'd better be very sure that the program is secure and that it won't allow anyone to do anything that you don't want them to be able to do.

This is an area where many beginners' CGI tutorials are very weak and as a result there are a large number of web servers that are open to attack from crackers through CGI programs. I don't want to give the impression that CGI is inherently insecure. It is no more insecure than any other web technology and it's probably easier to make CGI secure. I just want to make the point that you need to consider security.

What can possibly go wrong?

Before looking at how we can increase the security of our CGI programs, let's just look at a few examples of what can go wrong.

In the first example you have a simple CGI script that gets the name of a file as a parameter and displays that file in the browser. A first attempt at writing this program might look like this.

```
#!/usr/bin/perl -w
use strict;
use CGI ':standard';

my $file = param('filename');

print header(-type => 'text/plain');

open FILE, $file or die "Can't open $file: $!\n";

while (<FILE>) {
    print;
}
```

In this case we're assuming that the 'filename' parameter will contain the full path to the file we want to display. I hope it's obvious why this is a very bad idea. What would happen if someone passed you a filename of `/etc/passwd`? They would get the contents of this file displayed on their browser. Now, I realise that the actual passwords in this file are encrypted, but they will see all of your usernames which gives them a foothold if they are trying to break into your server. And they can run something like *crack* against the password list to see if any of the passwords are particularly weak. All in all a bad idea. And remember that they can see any file on your system in the same way.

So here's a second version of the script. In this one we assume that the files we want to display are all in the same directory and we'll restrict the viewer to displaying files from that directory. Or, at least, that's the plan.

```
#!/usr/bin/perl -w
use strict;
use CGI ':standard';

my $dir = '/path/to/data/files/';
my $file = $dir . param('filename');

print header(-type => 'text/plain');

open FILE, $file or die "Can't open $file: $!\n";

while (<FILE>) {
    print;
}
```

At first this looks a lot more secure, but it isn't. True a cracker can't use a filename of `/etc/passwd`. But what about `../etc/passwd` or `../../etc/passwd` or whatever is needed to go back up the directory tree to find the file. We need to be far more careful about what data we accept from the user.

In the second example we want to display information about the users who are currently logged on the the server. For this we'll run the Unix *finger* command, capture the output and display it in the browser. The name of the user to run *finger* against is passed in a parameter. Here's the basic program:

```
#!/usr/bin/perl -w
use strict;
use CGI ':standard';

my $user = param('user');
my $who = `finger $user`;

print header(-type => 'text/plain');

print "Here are the results for user $user\n\n";

print $who;
```

Again, this is an obvious technique for getting the information back from a Unix command. But consider what might happen if the data passed back in the `user` parameter was `dcross; mail cracker@blackhat.com </etc/passwd`. The backticks in the program will simply pass the command to your shell. The correct *finger* command will be executed, but the *mail* command will also run which is probably not something that you want.

The last security nightmare that we'll look at now involves

something called “cross-site scripting” attacks. This problem again comes from being too trusting of our input data. In last month’s article we wrote a simple script which prompted the user for input and wrote an HTML page displaying the data that they had given us. One of the input fields that we used was a text input form which allowed the user to post any data that they wanted as their name. We then displayed that name (twice) as part of the data on the resulting page. Of course as long as people were just entering their name in this field there would be no problem, but see what happens when I enter my name as **Dave**

Cross<script>alert('Gotcha!')</script>. As the picture shows (below right), the JavaScript code that we’ve added to the name gets executed as the name is displayed. In this case the JavaScript is harmless but it’s quite possible that it could do a lot of damage. There was a good article on this subject on perl.com recently. It’s called “Preventing Cross-site Scripting Attacks” and it is at <http://www.perl.com/pub/a/2002/02/20/css.html>.

Trust no-one

We’ve now seen a number of ways that CGI programs can be vulnerable to attack from users, how can we protect ourselves from these dangers? The most important thing that you can do is to take a leaf out of Agent Mulder’s book and “trust no-one”. Never assume anything about data that you receive from a user. Always put it through the most vigorous checks before using it.

As an example, let’s go back to the file display example. You’ll remember that our major problem here was to prevent a cracker from displaying our `/etc/passwd` file. One solution that I often hear is that we could create a form which contains a drop-down menu listing all of the files that the user is allowed to see. It would be simple enough to build this list using Perl code like this

```
opendir DIR, 'path/to/files' or die $!;
print qq(<select name="file" size="1">\n);

while (my $file = readdir(DIR)) {
    next if $file =~ /^\. /; # skip '.', '..' and hidden files

    print qq(<option>$file</option>\n);
}

print qq(</select>\n>);
```

Anyone using this drop-down menu would only be able to choose one of the files that we wanted them to choose, so surely that solves the problem. Well, no, this version is just as unsafe as the previous versions. The key phrase is “anyone using this drop-down menu”. You have no guarantee that the request that goes to your CGI program has been generated from your form. Someone could copy the HTML from your form, alter it to allow them to enter whatever data they want and submit that request to your CGI program. Or they could even write a simple program that allowed them to create any HTTP request they want and submit that to your CGI program (Perl is a particularly good language for writing programs like that!)

So the upshot of all that is that you cannot use a cleverly designed form to protect your program. You have to assume that you’re always dealing with potentially dangerous data and handle it accordingly. Perl (of course) makes it easy to do this.

The first (and most powerful) tool that Perl gives you is taint mode. When Perl is running in taint mode it automatically distrusts any data that it gets from the outside world and won’t let you use that data in many ways until you clean it up. You turn

CGI Script Repositories

NMS – for safer scripts

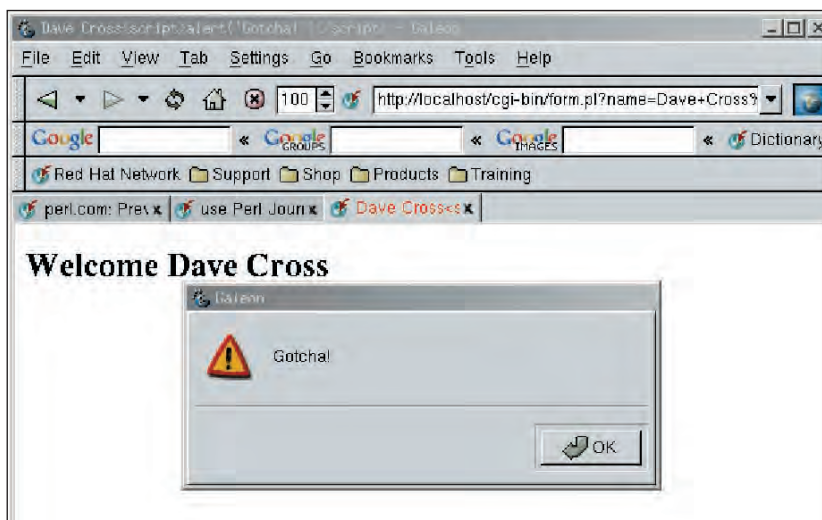
Most people who use CGI programs on their websites don’t write the scripts that they use. There are a large number of sites on the Internet that provide free scripts to download and use. One of the most famous is called “Matt’s Script Archive” and it contains a number of scripts written by a programmer called Matt Wright. Scripts from Matt’s site are in use all over the World Wide Web.

However you should not make the mistake of thinking that popularity and quality are the same thing. It’s long been known amongst the Perl community that the scripts found in most CGI script repositories are of very variable quality. I wrote an article called *Finding CGI Scripts for perl.com* which goes into this issue in some detail. You can read it at <http://www.perl.com/pub/a/2002/01/23/cgi.html>.

My conclusions were that the people most likely to use these scripts were exactly the people least likely to be able to judge the technical quality of the scripts and that this was leading to very insecure scripts being installed on a large number of web servers.

This is why the London Perl Mongers set up the nms project. The aims of this project are to provide a set of scripts which can be used in place of Matt Wright’s scripts. We didn’t choose Matt’s scripts because they were the worst, but simply because they were the most widely used.

The nms programs can be downloaded from <http://nms-cgi.sourceforge.net>. As I write we have replacements available for all of Matt’s scripts except *wwwboard* (which is a message board system), but I expect that to be available by the time you read this.



taint mode on by adding **-T** to your shebang line. Here’s an example of a simple (non-CGI) program which demonstrates how taint mode works.

```
#!/usr/bin/perl -Tw
use strict;

print 'Enter command: ';
my $cmd = <STDIN>;
chomp $cmd;

print ` $cmd `;
```

We prompt the user for a Unix command and print out the result of running that code. If you try to run this program and type in any command (e.g. **ls**) when prompted, then you’ll see this error:

```
Insecure dependency in `` while running with -T switch at
./taint.pl line 7, <STDIN> line 1.
```

This is Perl’s way of telling you that you tried to do something potentially dangerous with some tainted data. In this case we tried to pass a string to the Unix shell (using the backticks) without checking that the data only contained things that we want to pass to the shell.

The next thing that we need to know, of course, is how to go **>>**

A harmless cross-site scripting attack

◀ about cleaning the data. You do this using regular expressions. Here is our previous program rewritten to be taint-safe.

```
use strict;

$ENV{PATH} = '/bin:/usr/bin:/usr/local/bin';

print "Enter command: ";
my $cmd = <STDIN>;
chomp $cmd;

if ($cmd =~ m!^(/[w\-\+])$!) {
    $cmd = $1;
} else {
    die "Bad command: $cmd\n";
}

print ` $cmd `;
```

Fingering someone safely.

There are two changes in this version. Firstly we've set the **PATH** to a known value. This is because Perl in taint mode

distrusts the user's own path as it could potentially have been set to a dangerous value. Secondly we examine the contents of **\$cmd** before calling the shell. We check that the data in **\$cmd** contains only a specific set of characters. That set includes only word characters (alphanumerics and the underscore), spaces, a slash and a dash. This allows commands like **ls**, **ls -l** and **ls -l /home/dave** but excludes multiple commands like **ls; who**. Depending on your specific application the exact set of allowed characters may be different, but you should always keep the set as small as possible.

Having checked that **\$cmd** contains only our allowed characters we use a set of capturing parentheses to extract the value into **\$1** and then reassign this value to **\$cmd**. It is this action which untaints **\$cmd**. From this point onwards we can be sure that **\$cmd** only contains the data that we want it to contain. Perl also knows this and we can happily pass our data to the shell without generating an error. If **\$cmd** contains invalid characters then the regular expression match will fail and the program dies with an error message.

If you need some revision on Perl's regular expressions then you should look at the "perlre" manual page that was installed with Perl. For more information that you'll ever really need, see the book *Mastering Regular Expressions*, by Jeffery Friedl.

Now we can use taint mode to write a safer "finger" program.

```
#!/usr/bin/perl -wT
use strict;
use CGI 'standard';

$ENV{PATH} = '/bin:/usr/bin:/usr/local/bin';

my $user = param('user');

if ($user =~ m!^(/[w\+])$!) {
    $user = $1;
} else {
    die "Invalid user: $user\n";
}

my $who = `finger $user`;

print header(-type => 'text/plain');

print "Here are the results for user $user\n\n";

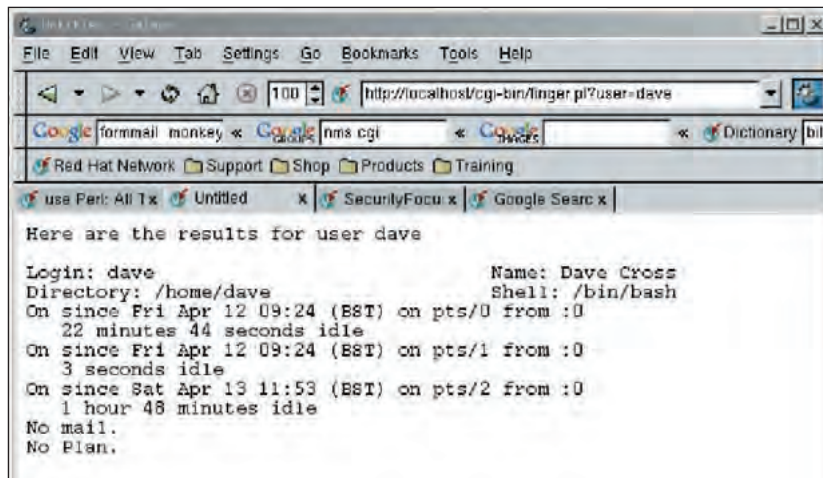
print $who;
```

We've made very similar changes to this program as we did to the previous program. We've set the **PATH** to a known value and we've untainted the value in **\$user** by checking it with a regular expression. Notice that in this case we can be far more strict in the set of allowed characters. As we're just looking for a user name we can just look for one or more word characters. The next figure shows the results of running this program on my local web server passing it the username **dave**.

For more info about taint mode, see the "perlsec" man page.

Other safety nets

Having fixed the problem with our *finger* example, let's take a look at how we'd solve the other problems we looked at earlier, starting with the file display script. To reiterate the problem, we have a directory that contains text files which we want to display to the user without them also being able to view our `/etc/passwd`.



Spam Attacks using FormMail

How to be an unwitting accomplice to spammers

One of the most popular CGI scripts on the World Wide Web is Matt Wright's *FormMail*. This script allows you to take the results of an HTML form and have the data sent in an email to a specified recipient. This is a very common requirement and Matt's script was one of the first freely available scripts to possess this functionality. Many ISPs automatically install this script on clients' sites.

The problem with this script is that it takes the address that it sends the email to from a form input. This is usually a hidden field in the HTML form but, as we've seen, that won't prevent a determined spammer from overriding that value and sending emails which originate from your server and annoy a large number of people.

This trick has apparently become very well-known amongst spammers and there are even automated *FormMail* detectors in existence which probe web sites looking for *FormMail* scripts that can be abused. The last eighteen months have seen a huge increase in the amount of spam email sent via unsecured *FormMail* scripts. Of course, this didn't go unnoticed by Matt Wright

and in July and August 2001 he released three new versions of the script in quick succession in an attempt to fix all of this problem.

Unfortunately the problem wasn't completely fixed. Whilst the latest version (1.9) is much better, there are still ways to use it as a spam relay. The problems are described in detail at <http://www.monkeys.com/anti-spam/formmail-advisory.pdf>. It's worth mentioning that this document is also uncomplimentary about the nms *FormMail* replacement but I should point out that since the document was written we have fixed all of the security holes in our script.

If you're using *FormMail* on your site then you really shouldn't be using Matt Wright's version. You should use the nms version instead. If you want to see how bad the problem is then try putting a file called *FormMail.pl* in your `cgi-bin` directory which does nothing but log the date, time and query string each time it is called. I run a script like this on a few of my domains and I get between five and ten probes each day on each of these sites.

Like the solution to our previous problem, we use a regular expression that matches our idea of a filename, and refuses to do anything if we're given anything that doesn't match that:

```
#!/usr/bin/perl -wT
use strict;
use CGI ':standard';

my $dir = '/path/to/data/files/';
my $file = param('filename');

if ($file =~ /\^[w\w\.]*$/) {
    $file = $1;
} else {
    die "Bad filename: $file\n";
}

print header(-type => 'text/plain');

open FILE, $file or die "Can't open $file: $!\n";

while (<FILE>) {
    print;
}
```

The program takes a familiar form. Only the regular expression has changed. In this case we're trying to prevent users from entering anything other than a filename in the current directory. We therefore insist that the filename starts with a word character which is optionally followed by any combination of word characters and dots. This allows filenames like "something.txt" or even "something.else.dat" but prevents "/something.txt" or things like ".././etc/passwd". Any time that you open a file with a name based on user input you should use these kinds of checks.

Cross-site scripting attacks

The final danger that we mentioned at the start of this article was that of cross-site scripting attacks where a user can insert JavaScript into data that you are going to display on a web page. Ways to get round this vary. If the data that you're displaying shouldn't contain any HTML at all then the brute force approach is to replace all < characters with the **<** HTML entity before displaying it to the browser. Here's how to make that change to last month's form processing program.

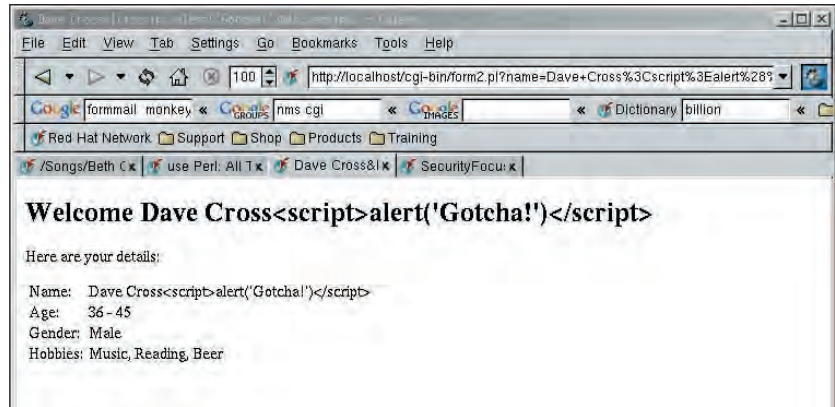
```
#!/usr/bin/perl -Tw
use strict;
use CGI ':standard';

my $name = param('name');
my $age = param('age');
my $gender = param('gender');
my @hobbies = param('hobby');

my $list;

if (@hobbies) {
    $list = join ', ', @hobbies;
} else {
    $list = 'None';
}

$name =~ s/</&lt;/g;
```



```
$age =~ s/</&lt;/g;
$gender =~ s/</&lt;/g;
$list =~ s/</&lt;/g;

print header,
    start_html(-title=>$name),
    h1("Welcome $name"),
    p("Here are your details:"),
    table(Tr(td('Name:'),
        td($name)),
        Tr(td('Age:'),
        td($age)),
        Tr(td('Gender:'),
        td($gender)),
        Tr(td('Hobbies:'),
        td($list))),
    end_html;
```

Attempted cross-site scripting attack fails.


It's a very simple change. We've simply run the transformation **s/</</g** against all of the user input variables to remove any potentially dangerous HTML tags. Notice that we've done it to all input and not just the data that comes from a text field. This is because, as I said earlier, we can't rely on the fact that the data has been entered via our form. Here's the result of trying to run a cross-site scripting attack on our new program (above). Note the absence of embarrassing JavaScript pop-up windows.

If, however, you want to include the ability for users to enter HTML in their data then you have a lot more work on your hands. You would need to keep a list of allowed HTML tags and attributes. Then you would have to parse the users input to work out exactly what they have tried to enter and remove anything that is not allowed. This is far from trivial and I don't have enough space in this article to go into any more detail. If you'd like to see an example of how it's done, please take a look at the guestbook script from the nms project (see *About CGI script repositories*).

Conclusions

I hope I haven't scared anyone away from CGI programming. These kinds of security issues exist no matter what kind of web technologies you use and they are actually easier to fix in CGI programs than they are in many other technologies.

CGI programming is fun and it is relatively easy to write programs that do interesting and useful stuff. You just have to be aware of everything that can possibly go wrong.

There are a number of good places to go to get more information on CGI security. One of the best is Lincoln Stein's *WWW Security FAQ* which is at <http://www.w3.org/Security/Faq/www-security-faq.html>. 

NEXT MONTH

In next month's article we'll look at some more advanced CGI concepts, like using cookies to store user data and using templating systems to separate your HTML from your Perl code.

Answers

If you are really stuck and the HOWTOs yield no good result, why not write in? Our resident experts will answer even your most complicated problems!

Experts this month

Whatever your question is, we can find an expert to answer it – from installation and modern woes to network administrations, we can find the answer for you – just fire off a letter or email and it'll all be taken care of.

LXF answers guy

David Coulson

is a networking and security guru with plenty of sysadmin experience to boot.



Richard Drummond

is an experienced programmer who can answer queries on a variety of subjects. A keen Debian user, he's also our resident Java guru.



Nick Veitch

is the editor of the magazine, and answers your easy questions! Or indeed anything to do with *Grub*, *LILO*, *netatalk*, *vi*...



Prompt service

Q I've enjoyed your magazine since I first saw it here in the US. Unfortunately the subscription price is too expensive for me, but I do pick up a copy to browse when I'm at Barnes and Noble. I always enjoy your review articles and HOWTO sections, and it adds to my Linux knowledge.

I have one question for you. I noticed that one of your reviewers, David Coulson, has an awesome shell prompt. Would you mind letting me (and the rest of your readers) know what his PS1 environment variable is? Keep up the good work!

Jeremy Turner

A The prompt actually sets two environmental variables, **\$PS1** and **\$PS2**; The latter being used when the shell is expecting more input, if you forgot to close a quote, or ended a line with a \.

The following should be inserted into your `/etc/bashrc`, although you may want to set it up as a function which is called from individual `~/.bashrc` files, so that all users don't end up with the prompt:

```
if [ "$USER" = "root" ]; then
  COLOR1='\033[00;31;47m]'
  COLOR2='\033[00;30;47m]'
  COLOR3='\033[00;33;47m]'
  COLOR4='\033[00;33m]'
else
  COLOR1='\033[01;32;46m]'
  COLOR2='\033[00;30;46m]'
  COLOR3='\033[00;34;46m]'
  COLOR4='\033[00;34m]'
fi

GRADO='\033[00m]'
GRAD1='\033[33;262;261;260]'
GRAD2='\033[260;261;262;333]'
COLOR5='\033[01;33m]'
COLOR6='\033[01;37m]'
COLOR7='\033[00;33m]'
COLOR8='\033[00;37m]'
```



A nice colourful prompt is pretty, but it's also useful for it to contain some helpful shell information.

```
PS1=$COLOR1$GRAD1$COLOR2`h`
'$COLOR3$GRAD2$COLOR4$GRA
D1\
$COLOR6' $(date +%l:%M:%P) `d`$
NONE`n'$COLOR5`u'$COLOR8`:'$
COLOR7`w'$COLOR8`$'$GRAD0' `
PS2=$COLOR1$GRAD1$COLOR3$
GRAD2$COLOR4$GRAD1$COLOR5
`>'$GRAD0' `
```

It is, of course, possible to adjust the colours to your own personal taste (or lack thereof), with the **COLOR** variables at the top of the script.

Lucent winmodem

Q Firstly I'd just like to congratulate the LXF team for a great magazine.

I recently purchased a Dell Inspiron 8100 1.2GHz, 512MB 133MHz SDRAM & 32MB nVidia Geforce2 – quite a combination. However after setting up a Red Hat 7.2 – Windows 2000 dual boot configuration I came across a disappointing development. Having installed Red Hat (earlier versions) on my old computer I was fortunate not to have the WinModem conflict problem! My Dell has unfortunately an integrated 56Kbps V90 Modem and 10/100 Ethernet. This leaves me in the unfortunate position of not having Internet access, which when running Linux is like having your dream car sitting in your garage and no insurance to take it out on the open road!

I was hoping you guys could point me in the right direction. Can

purchasing a PCMCIA card or USB port adapter solve my problem or is my dream car destined to never hit the information super-highway.

James McEvoy

A The 10/100 ethernet is either an Intel EthernetExpress Pro 100 card, or another popular laptop NIC. A quick check of `/proc/pci` (or the Windows device manager) should point you in the right direction. Red Hat should have figured out what the built-in NIC was during installation, so that should not be a problem.

As for the modem; Dell use the Lucent chipsets – one of the few which actually work with Linux. See www.physcip.uni-stuttgart.de/heby/ltmodem – although it may take a little creativity to make it work as you want, as rather than using `/dev/ttySx`, it will use `/dev/ttyLT`.

You may find some useful info at www.linux-laptop.net, as there will be someone else who has an Inspiron 8100, and has faced exactly the same problems as you. Even if it is all working happily, there are usually a number of optimisations and tweaks to help you get the best performance out of your laptop, along with support for hardware which you didn't think you had, such as sensors to monitor the temperature of the system.

Small X

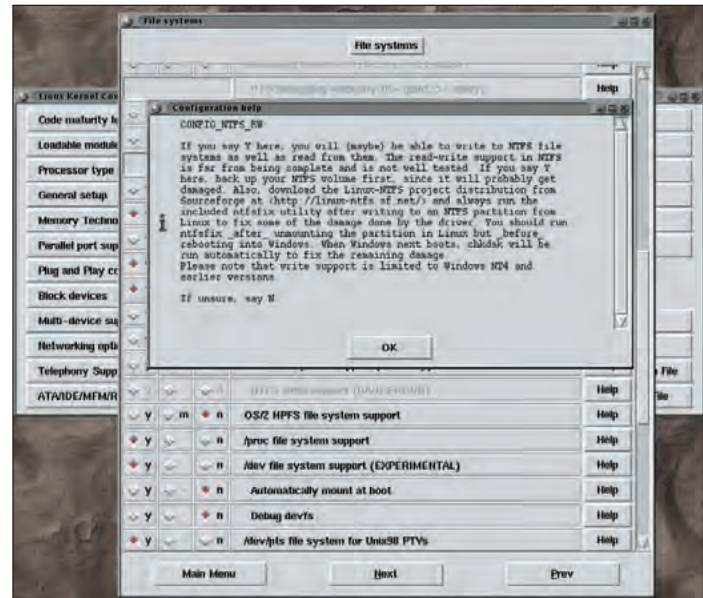
Q I have a couple of problems with Redmond Linux, from LXF25. Hopefully, you can give me some help.

The laptop in question is an Atlas Flyer 3400 (made by Hi-Grade), 14.1" TFT, 256MB RAM, PIII 1GHz, running both Windows 2000 Professional (NTFS partition) and Redmond Linux.

1. The most urgent problem is that I have some data on the Linux partition, which I want to move to the Win 2000 partition. The Linux partition is not visible from within Win 2000, but I can access the Win 2000 partition from Linux. However, I cannot move or copy the data. My user name has full admin access rights, but I am unable to change the NTFS partition rights to allow writing. I have also tried logging in as root. This allows me to change the rights, but when I try to confirm this action, I am told this is not allowed. The data in question is too big for floppy media, or anything else available to me and I need to move it. Any suggestions?

2. The second problem is that I cannot find the correct screen/resolution settings for this machine. Under Windows 2000, the video

card is shown as an SIS 600 and the monitor as a standard 1024x768. With 16MB of video memory, this allows 1024x768 to be displayed under Windows 2000 without any problem. During Linux setup, I used the SIS 600 option (probing proved this as being acceptable, i.e. it didn't hang), then tried all the options under Standard Monitors and also the LCD options. Many didn't work at all, but for those which displayed something, basically what seems to happen is that the resolution doesn't change. It always seems to be 640x480, no matter which settings are used. Some higher resolution settings are apparently accepted, but then display no differently. The result is that I have a kind of virtual screen, where only part of it is displayed and you have to move the mouse to the extremities in order to 'move' the screen to display the rest. I have also tried changing the screen settings from within Linux and this has proved unsuccessful too. Once or twice, it has resulted in my



Writeable NTFS support is available under Linux, but there is no guarantee that your filesystem will end up in one piece afterwards.

needing to reload Linux, so I don't want to change anything until I have sorted out my data problem above. Again, any suggestions?

Matthew

A Starting with your first question. Under Linux, the NTFS filesystem is read-only, and while there is experimental support for writing to an NTFS filesystem in the kernel, considering you will probably not want your Windows partition to go up in smoke, it's probably not something you want to experiment with. Saying that, the r/w NTFS implementation in 2.4 doesn't work with Windows 2000 anyway. The simplest solution, assuming it is possible due to disk space constraints, would be to create a small FAT filesystem on the drive, which will be accessible, and writeable, by both operating systems.

Under X there are two types of resolution. There is the physical display resolution, which is that being displayed on your LCD monitor, and there is the virtual desktop resolution, which is what you're finding when you're scrolling around. It is quite likely that the LCD panel will work well at 1024x768 under Linux, but 640x480 is the first resolution X finds when you start it up. You can use **Ctrl Alt +** and **Ctrl Alt -** to cycle through the different resolutions available to X by your video card, but without looking at your XF86Config file it's difficult to tell if that really is the case, or if X does not think your LCD panel is capable of such a resolution and is removing them from the configuration. However, since you have a virtual screen as default, it would be apparent that X thinks you can handle a higher

A QUICK REFERENCE TO: RPM

RPM is the package manager used by many distributions, including Red Hat, Mandrake and SuSE. Rather than having to compile everything from the source code, RPMs allow the installation of pre-compiled binaries of packages – ensuring that any dependencies are met before installation.

Installing an RPM downloaded from the Internet is done with the **-U** switch:

```
# rpm -U rpm-python-4.0.2-6x.i386.rpm
```

Before **RPM** will attempt to install the package, it will check existing packages which we have installed, and check that any required dependencies have been met. In this case, it complains:

```
error: failed dependencies:
libbz2.so.1 is needed by rpm-
python-4.0.2-6x
libdb-3.1.so is needed by
rpm-python-4.0.2-6x
```

In order to have this RPM install correctly, we need to install the package which provides **libbz2.so.0**, which is **bzip2**, and

one which provides **libdb-3.1.so**. Once these two dependencies have been met, the original RPM will install.

Of course, before we install an RPM, it's useful to know what it is. Performing queries on RPMs is done using the **-q** switch, and it's useful to know what dependencies are provided for by an RPM, which is done with **--provides**:

```
# rpm -qp --provides bzip2-1.0.1-3.i386.rpm
libbz2.so.1
bzip2 = 1.0.1-3
```

However, there are times when we know that dependencies have been met, either by a broken RPM, or by installing something from a source distribution, and we can have **RPM** ignore the dependencies of an RPM, and install it anyway:

```
# rpm -U --nodeps rpm-python-4.0.2-6x.i386.rpm
```

Naturally, if you really have not satisfied the dependencies for the package, then it's not going to work at all, so having **RPM** ignore what it requires usually causes more

problems than it cures.

While we can build from source, and install it without telling **RPM** about it, it is possible to build from the source code, create an RPM, then install it so everything behaves itself. Generally with RPM distributions, there are also **src.rpm** files, which are RPMs containing the source code for the particular package. The quickest way to build an RPM from a **src.rpm** is with **--rebuild**:

```
# rpm --rebuild bzip2-1.0.1-3.src.rpm
```

If the **src.rpm** does not exist, there is usually a 'package.spec' file within the standard source tar ball, although even if one does not exist, it's not a difficult task to create one by copying it from another package and adjusting it appropriately. Building an RPM from the spec file is very simple:

```
# cd /usr/src/package-1.0.1
# rpm -bb package.spec
```

This will produce a binary RPM in your **/usr/src/RedHat** directory – install like any other RPM.

FREQUENTLY ASKED QUESTIONS

FAQ How do I connect to the Internet with a modem under Linux?

There are many different ways to set up a dialup connection with Linux. Assuming we're after a standard PPP connection, then we can use something like *kppp*, *gnome-ppp*, *wvdial*, or one of the numerous distribution specific dialers. With something as basic as *kppp*, you simply enter the appropriate 'phone number, along with your user name and password, then try it out and see what happens. Of course, we can also do it by hand, by creating chap scripts and creating a */etc/ppp/* options for when we **ppp-on**:

```
/etc/ppp/options;
```

```
/dev/ttySO
```

```
crtscts
```

```
modem
```

```
lock
```

```
connect 'chat -f /etc/ppp/chat-script'
```

```
defaultroute
```

```
mru 576
```

```
mtu 576
```

```
/etc/ppp/chat-script;
```

```
ABORT 'NO CARRIER'
```

```
ABORT 'BUSY'
```

```
" ATZ
```

```
OK ATDT08452121666
```

```
ogin: username
```

```
word: password
```

```
ocol: ppp
```

FAQ Where is COM1 in Linux?

Linux uses */dev/ttySx* for serial ports, and COM1 is referred to as */dev/ttySO*. You can check to make sure your modem is really connected to */dev/ttySO* with a program such as *minicom*, where you can send **ATZ** to the modem and check to ensure that it responds with **OK**.

FAQ What is PAP and CHAP?

PAP and CHAP are different authentication methods used by PPP servers, and which one you need to use will depend upon your ISP. Nearly all ISPs use PAP, although users of BT Openworld's dialup services will need to use CHAP.

FAQ Will my WinModem work?

Depending upon the chip set used by the WinModem, it may or may not work. Lucent models generally work better than others, in so far as that they actually occasionally work at all.

www.linmodems.org has docs, and links to kernel modules, for a number of common chip sets, although your mileage may vary.

FAQ How do I use my cable modem?

Cable modems are generally connected to a NIC on the computer, so we need to use DHCP, rather than PPP, to connect to the Internet. There are many DHCP clients available, including *dhcpcd*, *dhclient* and *pump*, although generally a distribution will only have one of them installed. *pump* is usually the most successful with cable modems, so we can configure our NIC with:

```
# pump -i eth0
```

For a more permanent fix, you will need to reconfigure your networking within your distribution, in order for the DHCP client to be started at boot time, rather than having to do it by hand all of the time. If you have limited success with *pump*, then it's often best to try another client, such as *dhclient*, as some don't like working with certain DHCP servers.

FAQ Will my frog work with Linux?

Unfortunately under Linux, frogs are limited to 2400baud, and that is

assuming you can make them sit still long enough to negotiate the PPP connection. Often, an upgrade to a toad will solve the problem.

If you're actually referring to the Alcatel SpeedTouch USB DSL modem, used by the majority of ISPs in the UK, then it will work using the binary kernel modules distributed by Alcatel. You will need to compile *PPP-over-ATM (pppoe)* into the kernel, as well as obtain a number of user-space configuration programs, but many people do use this system for their Internet access. A fairly comprehensive HOWTO is at: <http://linux-usb.sourceforge.net/SpeedTouch/howto.html>

FAQ Can I share my Internet connection between both of my systems?

Assuming you already have a functional network, using private IP addresses, it's very simple to allow other machines to use your Internet connection. If you're running a 2.2.x kernel on the machine with the modem, then you will need to recompile the kernel with *ipchains* support, and install the *ipchains* package. For 2.4.x users, there is the choice of *iptables* or *ipchains*, although the former should be chosen

resolution, as the virtual size of the X display can not be changed once it is running, so X will always start with the highest you are capable of. To permanently change the resolution to something high, edit your *XF86Config*, and put the highest resolution in the list first, within the **Screen** section, as the first one it finds is the first it uses.

Mixed modules

After a few months struggling, I think the time has come to bother you with my problem. Currently I've got three kernels running which I can chose from by **LILO**: 2.2.16, 2.4.12, 2.4.18. Why so many? Well, because only the first one is able to load modules and gets ISDN and such running. I've tried everything

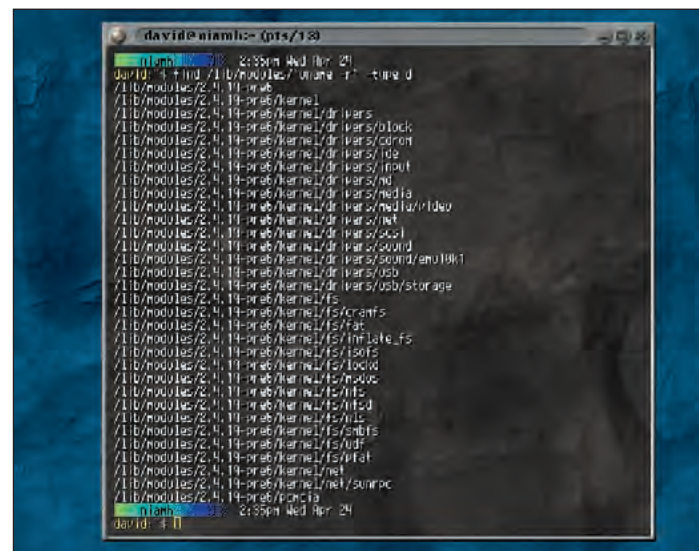
but now I'm getting desperate.

Where did I go wrong...?

Thanks for you wonderful magazine, it's helping me a lot understanding Linux.

Wim Wilts

You supplied us with a large number of .config files from the kernel, and a fairly comprehensive *dmesg* output from the kernel. The 2.4.x kernels require a large number of package upgrades, including *binutils*, *pppd*, and, of course, *modutils*. The */lib/modules/<version>* directory has changed very significantly between 2.2 and 2.4, so you need to ensure that you have the latest release of *modutils* installed, so that *modprobe* and *depmod* can actually find the modules. The latest *modutils* distribution is 2.4.15, which



The organisation of the */lib/modules* directory has changed quite a lot in 2.4, so you need a version of *modutils* which knows about it.

if you don't already have an *ipchains* configuration, or have the time to start again from scratch. We can share our connection by using IP Masquerading, which is done with one of the following commands depending on your choice of firewall:

```
# ipchains -A FORWARD -j MASQ
or
# iptables -t nat -A
POSTROUTING -o ppp0 -j
MASQUERADE
```

```
# modprobe ip_conntrack_ftp
```

We also need to enable IP forwarding, which is done with:

```
# echo 1 >
/proc/sys/net/ipv4/ip_forward
```

FAQ How do I secure my system against attackers?

Firewalling external connections is easily done with *iptables*, assuming you don't want to permit any incoming connections:

```
# iptables -A INPUT -i eth0 -j
ACCEPT
# iptables -A INPUT -i ppp0 -m
state --state ESTABLISHED,
RELATED -j ACCEPT
# iptables -A INPUT -i ppp0 -j
LOG
# iptables -A INPUT -i ppp0 -j
DROP
```

should solve the problems you are experiencing. You may also want to check up on the releases of the *ISDN4Linux* packages, and see if there are any updates to them for 2.4 users.

KOffice and Slack

Q I am running a Slackware 7.1 machine, 400MHz 256MB RAM, dual-booting with W98, NT4, W2K, etc.

I am trying to install *KOffice-1.1* because I feel this is the way to go rather than *StarOffice* because of open source reasons. I found that I had to install *Qt* before *KOffice*. So I downloaded *Qt-2.3.0.tar.gz* and installed that, but had to edit the *.profile* file in my root directory as per the install instructions of *Qt*.

This worked fine, but when I

attempted to install *KOffice*, I did not get beyond the *./configure* point, and received the following error message: "checking for KDE... configure: error: in the prefix, you've chosen, are no KDE headers installed. This will fail So, check this please and use another prefix! " However, I did not choose any prefix, so I presume that there must be a default one.

Ivor Eloff

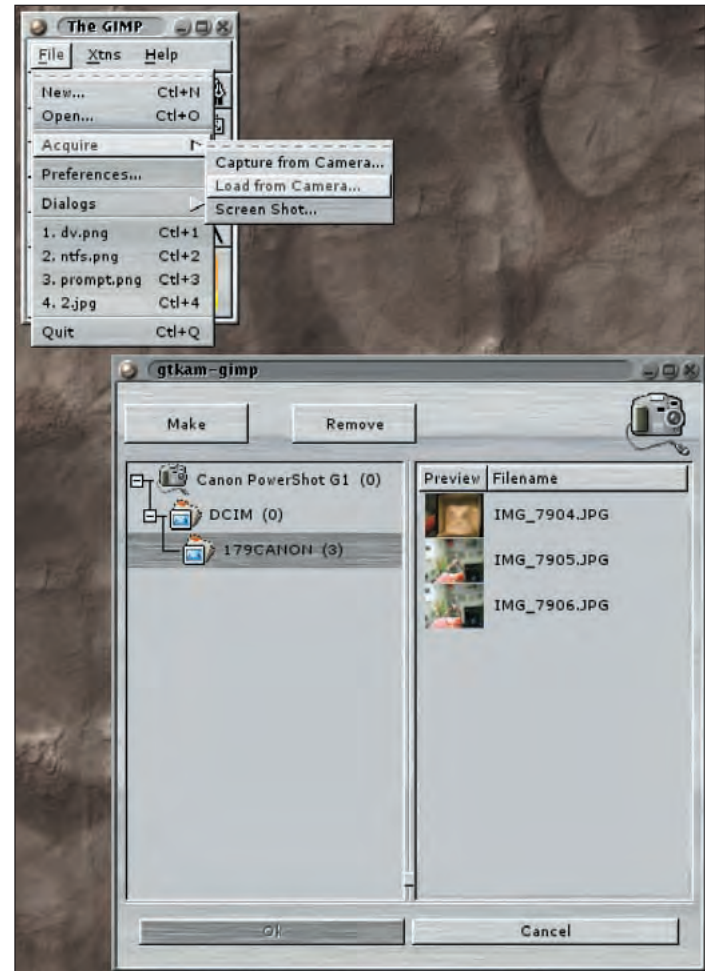
A Installing *Qt* only satisfies the *Qt* part of the *KOffice* requirements. As you might expect from it starting with a 'K', *KOffice* needs KDE to be installed, and at the very least, you will need to install *kdelibs* and *kdelibs-dev*, although you may need *kdebase* too. These can be downloaded from www.kde.org, from the Slackware FTP server, or from one of the numerous mirror sites, including download.sourceforge.net. You will need to specify *KDEDIR* and *QTDIR* within */etc/profile*, rather than just *~root/.profile*, as many other things rely on *QTDIR* and *KDEDIR* being set for all users.

Scanner blues

Q I have recently purchased an Epson Perfection 1650 Photo scanner to use in my SuSE Linux 7.3 setup. I made this decision after checking with the Epson web site to see if the 1650 was supported under Linux. It now seems that I may well have wasted £190 because, although I managed to get an rpm package from Epson, this would not install because it generated a dependency error **libgimp-1.2.so.0 is needed by iscan**. This seems to prevent my scanner from being recognised.

I have tried all the suggestions to do with setting up a USB scanner, as contained in the SuSE database but none work. Using *USBView* I can get the Vendor & Product ID but *modprobing* these still gives no result.

I have, in desperation, checked with SuSE but, because I am now out of my 90 day support, they refuse to give any help. A bit rich considering I have purchased all versions from them, ending with 7.3. I think it should be made abundantly clear to any wouldbe Linux purchasers, what hardware is supported.



The *GIMP* can use a variety of sources, including digital cameras and scanners, to acquire images.

Can you help in any way please. If not directly, perhaps with an article on installing USB scanners. The irony of this saga is that I managed to get an Epson Perfection SCSI scanner working really well but ditched it because it caused a problem in Windows ME. Look forward to hearing from you,

Michael Griffiths

A *libgimp-1.2.so.0* is provided by the *libgimp* package, and is part of the *GIMP* graphics software, which is likely used to capture the images from your scanner. Once you have *libgimp* installed, along with the other *GIMP* packages, then you should be able to scan something using *GIMP*'s 'Acquire' capability. With most USB devices of this sort, including digital cameras, there is generally no kernel module, as any software can access the devices via their */proc/bus/usb/* entry, much like the way */dev* works. As Epson have produced some plug-ins for this particular scanner, it is probably wiser

to contact them, rather than SuSE, in order to obtain trouble-shooting advice and configuration information, as SuSE will not be able to support third-party device drivers.

Camera obscura

Q I'm running SuSE 7.3/ Kernel 2.4.10, and my JVC digital video camera is connected to a Studio DV IEEE 1394 card, which should work using e.g. *Broadcast2000* or *dvgrab*. Well, the card is properly recognised but not usable. When I start *Broadcast 2000*, connect the camera and switch it on nothing happens. I read plenty of IEEE 1394 HOWTOs, looked at SuSE's support and read many postings from groups.google regarding the problem. Unfortunately I could not find any easy advice. Most stuff explains how to compile older 2.2x kernels to work with IEEE 1394. I'd be grateful for an advice "for idiots" like me :-)

J. Richter



« **A** The centre for IEEE 1394 development under Linux is at <http://linux1394.sourceforge.net>, which contains updated kernel modules and comprehensive HOWTOs for configuration. Looking at their Hardware Compatibility List, the support for JVC digital AV cameras ranges from 'Unknown' to 'Works Great', so you'll need to check up on your specific model and see if it works or not. Apparently both *Broadcast 2000* and *dvgrab* work well with the Linux IEEE1394 code in 2.4, although there is also *Kino* (www.schirmacher.de/arne/kino/), which seems to be a well maintained package.

For further DV editing info take a look at our *Roundup* on page 34.

Glibc config

Q Hehe, OK if I'd received an email with this title I know I'd run a mile... I have tried, sweated blood, tried and almost tired of it.

I now know that my early attempts to upgrade *Glibc* under RedHat 6.1 were flawed, but as the learning experience continued I was not deterred. My need for the upgrade eventually disappeared, but my fascination with it has kept me going. I am fairly well versed in many things Linux and Internet (to RFC level). I am a professional C++ & Real-Time programmer under Linux, a member of the Internet Direct (Indy) Open Source group (*Delphi/Kylix*) and have no qualms with recompiling anything and figuring out the whys & wherefores. It wouldn't be the first time I've been called a sad geek by my friends and been proud of it. :-)

As an incomplete list of resources, I have read the *Glibc2* HOWTO, built a Linux system from scratch (using the Linux-From-Scratch-HOWTO), hunted through newsgroups aplenty (thanks *DejaNews/Google*) and even had one-sided conversations on the *Gnu.org* mailing lists – they are busy fellows there, and I'd appreciate any response from them, including RTFM, but they have shown little inclination for how-to-install type questions over the past several years that I can see. The fact that at least one of these *Glibc* maintainers is an employee of a major distribution seems to suggest the lack of

divulgence is a subversive marketing ploy, but I leave my frustrations with this branch of GNU there.

I have recompiled *Glibc* many times without error, ensuring **make check** is performed cleanly, **ldconfig -v** is run after **make install** (both performed as single login) and ensured that *gcc* and all boot-required programs are statically linked to be on the safe side. I have tried upgrading the *Glibc* version, retained the *Glibc* version and downgrading the *Glibc* version (OK I didn't expect this last to work...). Every time (these days) I come up against a brick wall during boot up.

If I retain the RedHat 7.1 version of *Glibc* (2.2.2), I get dynamic load errors with *libtermcap* (it's linked with the same version!). If I upgrade the version, I get swap initialisation errors (overcome by removing the **-e** command line option) and syslog errors. Aaarrggghhh!

In the meantime, if I happen to crack the problem, I shall do my best to find the time to write an article for you, update the *Glibc2* HOWTO and post to a select few newsgroups/mailling lists. In the meantime, I'm just looking for pointers.

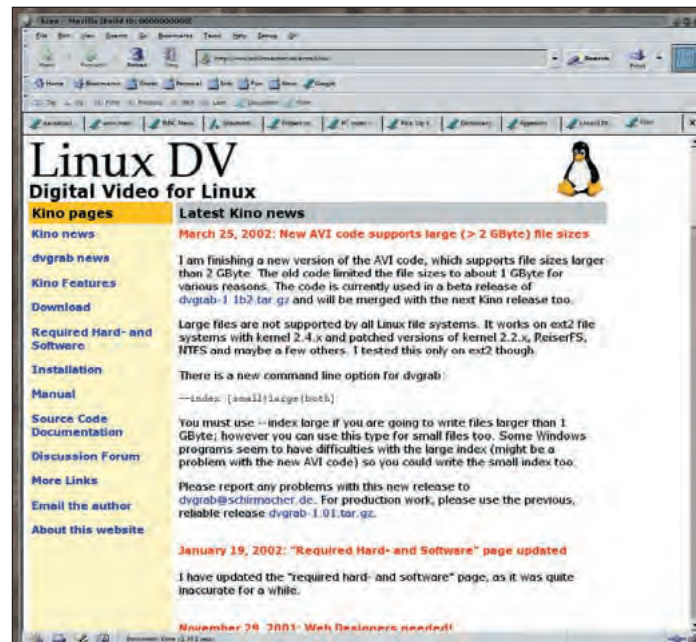
Keep up the great work in the magazine.

A. Peter Mee

A Compiling *Glibc* from source is akin to trying to put water into a paper bag using a spoon. The thing to remember is that the *glibc* version in RedHat 6.1 was 2.1, yet you're upgrading to 2.2. For reasons, which are probably best left out of the argument, dependencies under RedHat break rather spectacularly, so that if something needs *Glibc* 2.1, it really does want 2.1, and not 2.2.

Quite why *swapon* would want to break because of this is very unusual, and without some decent debugging output, it's pretty much impossible to even being to track the problem down. If *libtermcap* has issues with the RedHat *glibc* distribution, then you're probably a little bit stuck. Other than a mass RPM upgrade to 7.2, or indeed a reinstall with something which actually works, it's probably a losing battle which will waste far too much of everyone's time.

It's certainly possible to build *glibc*



Support for digital video cameras, over a Firewire interface, is possible under Linux, assuming the camera you are using is supported by the IEEE1394 kernel modules.

from source, without any problems, as those using Linux-From-Scratch, and indeed, many others who have upgraded their *glibc* distribution on a system which uses packages, but *glibc* is one of those things which you just don't want to push too far, in case you're stuck with a machine which won't do anything.

APM revisited

Q I read the letter from Andrew Hall on his problems after upgrading to Red Hat 7.2, and it occurred to me that I've had similar problems. In my case it was *apmd* deciding to put the box into standby because nothing was happening on the keyboard.

The symptoms fit – disk spinning down and box disappears off the network. Also the recovery method, plugging a keyboard or mouse in will cause the box to wake up from suspend mode, restoring system state and the network interfaces as if nothing happened.

Dave Addison

A Thanks for this helpful tip. Looking back at the original question this seems to fit the symptoms (as did the original answer about the changes in Red Hat device drivers). Hopefully one of these will be the solution Andrew is after.

If you don't need APM features at all, such as on a server, you should

disable them in the BIOS as well as disabling the Linux *apmd*. While useful on laptops, the reliability of this service is variable on desktops and other hardware. **LXF**

Submission advice

We are happy to answer all sorts of Linux related questions. If we don't know the answer, we'll find out for you! But in order to give you the best service, it helps a lot if you read the following submission advice.

- Please be sure to include any relevant details of your system. 'I can't get X to work' doesn't really mean anything to us if we don't know things like what version of X you are trying to run, what hardware you are running on.
- Be specific about your problem. Things like 'it doesn't work' or 'I get an error' aren't all that helpful. In what way does something not work? What were you expecting to happen? What does the error message actually say?
- Please remember that the people who write this magazine are NOT the authors or developers of Linux, any particular package or distro. Sometimes the people responsible for software have more information available on websites etc. Try reading the documentation!

We will try and answer all questions. If we don't answer yours specifically, you'll probably find we've answered one just like it. We can't really give personal replies to all your questions.

WRITE TO US AT:
Linux Format, Future Publishing, 30
Monmouth Street, Bath BA1 2BW or
email: lxf.answers@futurenet.co.uk

Coverdisc

Neil Bothwick is your guide through the wonders of this month's jam-packed Linux Format CD. Update your desktop or create your own GNU/Linux distro.



On the CD



Wherever you see this logo it means there's related stuff on the CD

Essential info

On page 105 we have grouped together essential info on the different types of packages on your coverdiscs – along with instructions for installing source packages.

Important notice

Before you even put the CD or DVD in your drive, please make sure you read, understand and agree to the following: The Linux Format CD/DVD is thoroughly tested for all known viruses, and is independently certified virus-free before duplication. We recommend that you always run a reliable and up-to-date virus-checker on ANY new software. While every care is taken in the selection, testing and installation of CD/DVD software, Future Publishing can accept no responsibility for disruption and/or loss to your data or your computer system which may occur while using this disc, the programs or the data on it. You are strongly advised to have up-to-date, verified backups of all important files. Please read individual licences for usage terms.

READ ME FIRST

Welcome to another issue and either a DVD or two CDs packed full of goodies for you to try. The main package this month is *KDE3*, which has just reached the full

release version. See our extensive coverage, starting on page 60, for full details of installing and using this fabulous desktop suite. If you want to install from the RPMs, it's worth checking your distro vendor site for updates first.

There's a lot more than just *KDE3* on the coverdiscs, though. We have the prime pickings of recent open source releases. The next few pages show you just a few highlights of them, so read on and then dive straight in.

System/ AcronisOSSelector

This commercial product was reviewed a few months ago, but we were unable to put a demo on the CD for various reasons. Now we have one so you can try it for yourselves. You may be asking, "Why do I need to pay for something that *GRUB* and *LILO* do for free?" *OS Selector* is more than a simple boot loader; it can also create and resize partitions.

Installation is either from Windows or Linux. To install from Windows, just double-click the **oss_s_e icon** and follow the prompts. Installing without Windows requires a boot floppy, but is almost as easy. Create the floppy with

```
dd
if=/mnt/cdrom/System/AcronisOSSelector/Linux/disk1.dup of=/dev/fd0
```

Replace /mnt/cdrom with the path to your CD/DVD drive if necessary. Then boot from this disk and follow the prompts. Although you don't need Windows to install or run *OS Selector*, you do need a FAT filesystem partition to install it to. If it doesn't detect one, then the installer offers to create one for you. The program needs 1.5MB of space on this partition for its files.

OS Selector is more than a boot manager; it also features a Disk Administration tool for managing partitions. This lets you create, resize, move, relabel, edit and delete

partitions. There are some limitations in this demo version, the main one being a sixty second delay on bootup. The Disk Administration section will allow you to try out the various options, but you cannot apply the changes to disk. Having said that, *OS Selector* is as easy to uninstall as it is to install. So if you try it and don't want to keep it, simply select "Uninstall Acronis OS Selector Trial..." from the Tools menu of the boot window and your previous boot loader setup will be restored.

Desktop/GKrellM

If you do everything from a console and eschew GUIs as unnecessary eye candy, especially those with themes, then skip to the next section, *GKrellM* is not for you. However, if you are one of those for whom the number and colour of LEDs is an important factor when choosing hardware, read on. *GKrellM* is a system monitor, more precisely it is a stack of system monitors that keep you up to date on almost every aspect of your system. The default setup displays CPU usage, process information, disk activity, network throughput, memory usage, laptop battery status, online activity and time and mailbox contents.

All of these are optional, and you can control the appearance and operation of the monitors you choose to use. Almost everything is

configurable, both in terms of the information reported and the style of the reports. Many of the monitors will open an information window when clicked on, to show further details about the aspect being monitored, and some of them are also able to launch commands.

When you tire of fiddling with the built in modules, there is the option to add plugins. Several of these have appeared on previous *Linux Format* coverdiscs, including plugins to monitors the CPU and fan via *lm_sensors* and to display a thumbnail of a webcam image in a monitor.

This is one of those programs you either love or hate. If you are into 'blinkerlights', you'll love being able to fiddle with so many ways of displaying so much information. If you have a more minimalist outlook, you'll hate it as a waste of screen space and CPU cycles.

Distros/Tomsrtbt

Sooner or later disaster will strike. You'll turn on your computer and Linux will fail to boot up. Some distros have a rescue mode available when booting from their CD, although in a major disaster this may not have what you need, or you may even have problems booting from CD. At times like this, the lowest common denominator of the floppy disk may be your last hope. *Tomsrtbt* is allegedly an



Blinkenlights galore. GKrellM tells you all you could possibly want to know about your system... and then some.

acronym for "Tom's floppy which has a root filesystem and is also bootable." although I'm not sure what language this actually works in. However, it is a good description. This is a packed Linux distribution on a single floppy disk, containing well over 200 commands and files.

To create the disk, unpack the gzipped archive and **cd** to the directory it creates. Then put a blank disk in the drive and type

```
./install.s
```

If there are any problems, consult the FAQ in the same directory. The file **tomsrtbt-2.0.23.ElTorito.288.img.bz2** unpacks to a 2.88MB image file that can be used as the boot image for a bootable CDROM. One possible use for this would be to back up your hard drive to CD-Rs or

CD-RWs and make the first disk of each backup set bootable with this image. For example, copy this file to a directory and you can create a bootable backup of that directory with

```
cd /path/to/backup
mkisofs -R -b tomsrtbt-2.0.23.ElTorito.288.img -c
boot.catalog -o ~/backupcd.iso .
cdrecord -v ~/backupcd.iso
```

Distros/Linux From Scratch

Prebuilt distributions, like Mandrake 8.2, from last month's *Linux Format* CDs and DVD, are certainly the easiest way to get Linux up and running on

standard hardware. There are times when you want something else. Either you are running non-standard hardware that the installer can't handle or you need a distribution for a specific purpose. Maybe you simply want to understand exactly what is going on instead of letting an installer shove a gigabyte or two of files onto your hard drive and configure it while hiding all the workings of the process from you. The easy to use front end of most distributions' installers becomes a handicap rather than a help in these situations.

Linux From Scratch is not a distribution in the normal sense of the word, in that it is not a complete click-and-go package. Instead it is a detailed explanation of how to go about building your own Linux setup, well, from scratch really. In addition to detailed instructions on every stage of the process, the core packages you will need are included, as tarballs of course. Obviously, it will take longer to setup up a box to run Linux like this than simply banging a Red Hat or SuSE disc in the drive and rebooting, but at the end of the process you will have a far greater understanding of how it all works and how all the pieces fit together.

For this reason, Linux From Scratch is not only of interest to those who want to build their own distros. Even if you are perfectly happy to run your prebuilt distribution, the information in Linux From Scratch can help you to gain a better understanding of how things work, which is always a good starting point when you find yourself in the situation of trying to find out why things don't work.

We have included an earlier version of Linux From Scratch on a previous Linux Format DVD, but have had several requests to put it on the CD too. That's hardly surprising as building your own lightweight distro is ideal for lower specced hardware, which most likely will not have a DVD drive.

Internet/Mozilla et al

Mozilla is covered elsewhere in this issue, so there's not much to say about it here except that the first release candidate of *Mozilla 1.0* is in the Internet directory of the CD. Mozilla is more than a standalone browser though. Its rendering engine is used by other browsers, such as *Galeon* and

KDE3

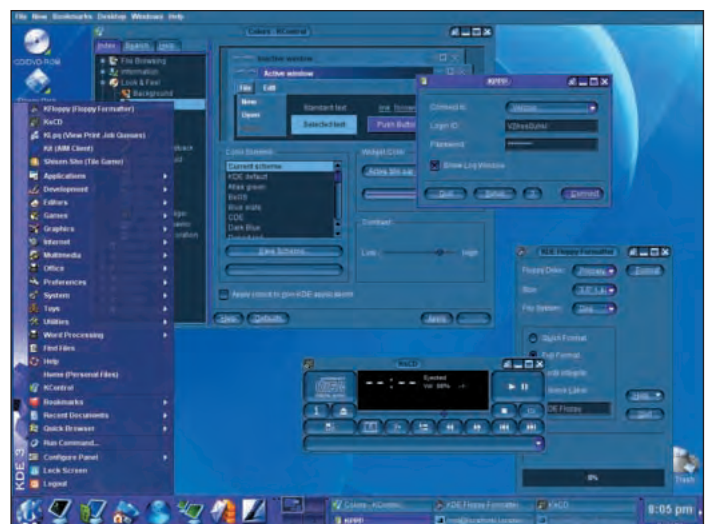
Where to find your binaries

KDE 3 has finally reached release status and we bring it to you, both on the DVD and the CDs (see the feature on p60).

You'll find KDE3 in Desktop/KDE3 on the DVD or the first CD. The disc contains all the KDE 3 files in source tarballs and RPM packages for Red Hat, Mandrake and SuSE, including SuSE PPC. In each case this is for the latest version of each distribution. You should check the **README** file in each distribution's directory, and it may be worth checking the versions against any currently available on your distribution's home site, in case any updates have

been released. If you are installing from the source files, you will need to have the latest version of *Qt* installed first, this is in Desktop/Qt.

The CD contains the tarballs and the binary RPMs, there wasn't room for the source RPMs. The CD is also missing most of the localisation files. Only GB English, French, German and Spanish were included. If you need a different package, you can download it from [ftp://ftp.kde.org/pub/kde/stable/3.0](http://ftp.kde.org/pub/kde/stable/3.0). Unless you're reading this magazine in the newsagents, in which case you should put it back and pick up the DVD issue!



You don't like the default look of KDE? That's no problem, we have plenty of alternative themes for you to try.

Skipstone. Both of these attempt to provide a less resource hungry alternative to the heavyweight *Mozilla*, while still retaining the benefits of all the work done on *Mozilla's* HTML rendering engine, *Gecko*. I'm not about to try to tell you which is better for you to use. They are all on the CD, try them for yourself.

Web browser is rather a misnomer for these programs as they can be used to display HTML documents from any source, not only the web. For example, the index files to the various sections of the *Linux Format* coverdiscs are in HTML. You navigate the CD or DVD using a "web browser" while going nowhere near the web. *Mozilla's* interface can be changed to suit almost any circumstance, and we have an example of this on the CD. *MozillaKiosk* uses *Mozilla* to display HTML pages but nothing else. There is no *Mozilla* GUI, all control is via links in

the HTML itself. This makes it ideal for kiosk information application, where you want to display only the relevant information without confusing the user with all the buttons and gadgets of a web browsers GUI.

Desktop/KDEthemes

Whether you have installed the latest *KDE3* or you want to stick with your old *KDE2* setup, there may come a time when you want a different look to your desktop. This directory contains several different themes for *KDE2* and/or *KDE3*. They range from simple wallpaper changes to complete themes that change the appearance of gadgets and menus.

Server/evoBB

Evolution BB is a web-based messaging forum or bulletin board



LinuxFormatCoverdiscCD

« system which lets you set up your own web-based discussion area on any web server that supports PHP and an SQL database. It's not limited to a single type of SQL database, and currently works with *MySQL*, *mSQL*, *MSSQL*, *ifx*, *iBase*, *PostgreSQL* and *Sybase*. Any version of PHP from 3 onwards is sufficient, and it works with Windows as well as Linux web servers. This means you can test it on your *Apache* system and still have it work if your web hosting provider uses, say, Microsoft's *IIS*.

Installing the forum software is an absolute doddle. After unpacking the tarball, you simply upload one directory to your web server and point your browser at <http://www.your.webspace/forum/install>. At least, that's the theory. The readme failed to mention that you first have to give some database details in the `connect.php` file. It needs to know the

database name and the username and password needed to connect to it. One quick edit of the file later, and the install script ran perfectly, not only setting up the system but giving the options to create several groups and forums. After the initial installation, you must delete the install directory from the server, or someone could use it to delete your forums.

All further administration is done through the admin URL, after you have logged in with your admin password. Despite the early version number, this appears quite a polished program. The help link takes you to some fairly detailed user documentation, which is quite unusual for an early version – documentation tends to be left until later in the development cycle. There is no documentation for admins, beyond the simple installation README, but everything seems fairly self-explanatory anyway. **LXF**

What goes where?

A quick guide to the coverdiscs' directories

To make it as easy as possible for you to find the type of software you are most interested in, we have divided the contents of the CDs and DVD into twelve directories.

These are the directories the discs are divided into and the type of software each one contains:

MAGAZINE

This contains programs and other files mentioned in various articles in each month's magazine. Look here for versions (maybe demos) of reviewed software or programs referred to in tutorials and features.

DESKTOP

A wide range of programs for general desktop usage of a Linux machine, from full blown windowing environments like KDE and GNOME to small utilities.

DEVELOPMENT

Anything relating to software development. This includes compilers, libraries, classes, debugging tools and development environments.

DISTROS

As you would expect, various Linux distros. These can be full distributions, that are only on the DVD or mini-distros that can also fit on the CD.

GAMES

What more is there to say, the Games section contains... well, games.

GRAPHICS

Paint, image processing, movie players, video capture. Anything to do with capturing, manipulating or viewing graphics can be found here.

INTERNET

The Internet directory contains client side internet and network software. This includes the obvious candidates like web browsers and email clients, as well as some more arcane Internet tools.

OFFICE

Various types of productivity software are included here. This is mainly word processors, spreadsheets, databases and accounting software, but there are some other programs that also fit here.

SERVER

This is the complement of the Internet section. It covers everything involved in providing services over the Internet or a network, as opposed to using them.

SOUND

Once again, a clearly defined category. Anything to do with capturing, playing processing or converting sound or music will go in here.

SYSTEM

This is where you will find updates and enhancements for various system components as well as various utilities to improve your usage or the security of your system.

ESSENTIALS

The Essentials directory contains various programs and packages that are often required by items on *Linux Format* coverdiscs but aren't always included as standard with distributions. It also contains the latest kernel sources. The DVD's Essentials directory contains the Linux Documentation Project too.

» CD CONTENTS AT A GLANCE

Disc A – KDE3

Disc B

Desktop

Advance

Affiche

Bochs

CdrLin

Devolution

Easymenu

fax4CUPS

Flasher

g-page

GarminGPSTool

Gdtkfft

Gedafe

Genes

GKrellM

Gnome-mlview

GOfax

HaywardFortunes

iftop

JIM

Kedge

Kleds

KMyMoney2

KPreg

KSambaKonqiPlugin

KVim

Lintad

LoFiMo

LogAnalysis

Logwatch

MuPO

Replacelt

Rigel

Skylander

SkyTVguide

Unpack

Personal information management (PIM) system

Clone of the MacOS-X "Stickies" application

Portable x86 PC emulation software package

Java-based GUI for *cdrecord*, *mkisofs*, *cdda2wav*, etc.

Quick, simple and powerful window manager for X.

Generate simple and advanced multi-level menu systems

Acts as a CUPS backend for a serial fax modem

Monitors changes to files and flashes a console LED

Send text messages to pagers or SMS enabled PCS phones

Download and save data from Garmin GPS receivers

Adds anti-aliased font support to GTK+-1.2

Application independent web front-end for databases

Personal genealogy database application

GTK+ and Imlib-based stacked monitor program

Tree-oriented XML editor for GNOME

Extensible fax solution that works with Hylafax

A collection of fortunes

Real-time bandwidth information on a specified interface

Personal Information Manager written in Java

Edge-sensitive virtual-desktop switching with your mouse

KDE Panel app for keyboards without LED's

KDE personal finance program.

Helper for completing web forms and other programs

SAMBA plugin for the properties dialog of Konqueror

A port of the Vim GUI to KDE 2/3

Fax and voicemail application

LoFiMo monitors log files in realtime

Extracts data for any of the recognised log messages

Analyzes and reports on system logs

Powerful PIM application for desktop and PDA systems

String replacement in text files with several features

Personal information manager for X

Modern astrology software for Linux/KDE2

Perl script to grab TV listings from www1.sky.com

Automates the process of extracting source tarballs

Development

Abbot

ArgParse

Batik

Blastic

borZoi

CCache

cdb

ClanLib

CodeGuide

ConfigAPI

David

File-Scan

FireString

Gambas

Greq

Libsafe

Itrace

ncc

Onyx

OOPS

Optik

RussianTeaHOWTO

SISC

Spread

SQlite

TinyQT

Scripted Java GUI testing framework

Process complex command-line parameters in shell scripts

Toolkit for Scalable Vector Graphics (SVG) images

A classic Basic interpreter

Elliptic curve cryptography library for developers

A fast compiler cache

A package for creating and reading constant databases

Game SDK currently supporting Windows, Linux and BeOS

An IDE that detects errors in your code as you type

Read, write, syntax check and erase configuration files

C/C++ code editor for GNOME

Used to make multiplatform virus scanners

A library to make string handling easier in C

Graphical IDE based on a Basic interpreter

Build simple GTK+ dialogs from within a shell script

Defends against buffer overflow and format string attacks

Debugging program which runs a command until it exits

Generates program flow and variable usage information

Stack-based, multi-threaded, general purpose language

A comprehensive bug tracking system

Command line parsing library for Python

How those Russian hackers keep going

Extensible Java-based Scheme interpreter

For developing reliable, robust distributed applications

C library that implements an SQL database engine

Compact QT subset for embedded use

Distros

BootEverywhereLinux

LinuxFromScratch

Minimalist (i386) Linux distribution on a single floppy

Build your own custom Linux system

NetstationLinux Tomsrtbt

For diskless thin clients terminals using x86 hardware
The most Linux on one floppy disk

Games

Epiphany
Ferox
GtkBalls
KWappen
Malom
MathWar
NebulaCards
PipeNightDreams
SpaceHulk
Tes
Teutron
XRick

Multiplatform clone of the game Boulderdash
Small MUD client based on the Qt library
Simple but awesome logic game
A colorful KDE2 board game
"Nine Men's Morris" game for SDL.
Educational flashcard maths game
Networked, modular card game engine using Java
Similar to the classic PipeDreams, using SDL
Board game in the world of Warhammer 40000
Multiplayer card game of visual perception
A fun game, akin to Tron or Nibbles
Clone of Rick Dangerous

Graphics

AutoTrace
EyeFract
Gimp
GQview
JPEG2PDF
K-3D
PDFMap
SDL-3D
SDLCam
tgif
TyGeMo
VTQLServer

Converts bitmaps to vector graphics
Julia and Mandelbrot set (fractal) generator
The GNU Image Manipulation Program
An image viewer for X
Converts a collection of JPEG images into a PDF album
3D modeling, animation and rendering system
Create very high quality maps in PDF format
Engine for displaying 3D Studio objects (.ASC) files
SDL GUI for the Philips Webcams
Vector-based drawing tool
TCL/Tk frontend to the photopc digital camera software
Server and clients for videotext access

Internet

AdvancedTFTP
ArchiveMail
Bookmark4U
Business-Associates
Datasphere
Encompass
EnergyMech
Fetchmail
FlashPlugin
FreelPdb
FTPRider
FXmakersIRCServices
Galeon
GetBinNews
Googleware
GSFDM
GTKmsn
JavaYEncDecoder
kShowmail
L2TP
LinuxSMS
Metis
Mikrop
Mozilla
MozillaKiosk
NewsGrab
rbot
Retchmail
Skipstone
SmsSend
tinc
WorldPrint

Client/server implementation of the TFTP protocol
Archives and compresses old email in mailboxes
Provides a comfortable bookmarking environment
Handles XML interface of the Amazon Associates program
Conferencing system that overcomes some of IRC's problems
Web browser using the GtkHTML engine for speed and size
Advanced IRC bot with many features
Remote-mail retrieval and forwarding utility
Flash Player plug-in for Mozilla and Konqueror
Manage, assign, track and audit IP address assignments
Set of Python scripts that build an FTP search engine
For those who spend their life on IRC
GNOME web browser based on the Mozilla rendering engine
Download binary files split in a newsgroup
Regularly send queries to Google
A new Internet downloader
Chat client that is compatible with the MSN protocol.
Decodes files encoded with yEnc
KDE tool for watching for email on POP3 servers
Layer 2 Tunneling Protocol VPN client/daemon
Perl script to send SMS to GSM phones
Collect information from web servers and spiders sites
Scans incoming email using third party virus scanners
The Mozilla 1.0 release candidate
Kiosk-style web browser based on Mozilla
Downloads and uudecode/ydecode binary files from USENET
IRC bot written in Ruby
Fast POP3 mail downloader
GTK+ web browser that embeds Mozilla via libgtkembedmoz
Send free SMS to any GSM
Virtual Private Network (VPN) daemon
Filter for Mozilla, HtmlDoc, and Netscape PostScript output

Office

Achievo
DataVision
EasyMailings
MimerDesk
PhpMyLibrary
RequestTracker
SMTM
Twiggi
WebGUI

Web-based project management and tracking tool
Database reporting tool similar to Crystal Reports
Generates mailing labels from an ASCII address file
Groupware environment for a wide variety of uses
PHP/MySQL web-based library automation application
Industrial-grade trouble ticketing system
Perl/Tk ticker, profit/loss calculator, and chart tool
Web-based groupware application using PHP4 and MySQL
Lets average users build and maintain complex Web sites

Server

Anti-censorshipTools
Apache
AtlasWebSite
autoresponder
BigBrotherLogAnalyzer
CryptNETKeyserver
DNSTools
EvoBB
Eyebrowse
GeekLog
IP-Atlas
Jigsaw
mcLinksCounter
PHP
PHP-FormMail
phpTest
Portmon
PowerDNS
PVote
SiteSeed
ToDoListPHP
V-webmail
VAMPWebmail
WebCleaner
XMail
ZenWeb

Abstracts services in an uncensored, reliable, fast manner
The world's most popular HTTP server
A web portal system that does not require a database
A simple email receptionist
HTML access logger and log analyzer
OpenPGP (RFC2440) compliant public key server
Browser-based DNS configuration and administration
Forum software free for use on your own web site
Web-based mailing list archive browser
Weblog software for performance, privacy and security
Plots you or any other host on the globe
W3C's leading-edge web server platform
Manage the links and counts the clicks on your web site
Widely-used general-purpose scripting language
PHP script conversion of the original FormMail.pl
Web-based testing program
Monitor network services and send notifications
High performance authoritative nameserver
PHP voting system using MySQL
Web development and content management system
PHP scripts that create a web-based list of things to do
Fast, customizable and standards-compliant webmail
Easy-to-configure, usable, web-based email client
Filtering HTTP proxy
Internet and intranet mail server
A system for building entire web sites, not just pages

Sound

Alsa
Autools
Beep
DarkIce
ERec
Ermixer
FFTScope
GLSVG
GnomeRadio
Honeyd
Jooky
LAoE
Maudio
MCDPlayer
MCfoo
RhythmBox
SpeexVoiceCodec
Spwave
Streamripper
Streamtuner
Tutka
WhiteNoise
XMMS-Iris
ZoltanPlayer

Provides audio and MIDI functionality to Linux
Play, record, generate, modify and analyse audio files
beep does what you'd expect: it beeps
Icecast, Icecast2 and Shoutcast live audio streamer
Records multiple raw audio streams from the soundcard
Sophisticated OSS mixer with a lot of useful features
Visualization plugin for AlsaPlayer
glsvg is an SVG/SMIL player using GLX.
FM-Radio tuner written in gtk+
Small daemon that creates virtual hosts on a network
Frontend for MP3 players which can run as a daemon
Graphical audio sample editor
Simple audio mirroring device (a sort of audio pipe)
Really small CD player with a curses-like user interface
Advanced and adaptive ogg/mp3 jukebox server
Music management software, inspired by Apple's iTunes
Patent-free compression format designed for speech
Speech file editor supporting several sound formats
Records shoutcast-compatible streams
Perl/GTK+ SHOUTcast browser
Tracker style MIDI sequencer
Turn your computer into an ambient random noise generator
Advanced OpenGL Visualization plugin for XMMS
A music playing daemon with its own mini HTTP server

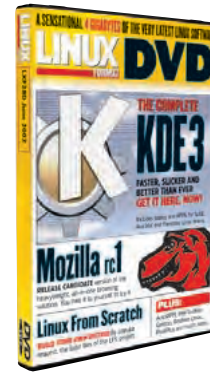
System

AutoRPM
CAPI4Linux
CastellanSentry
CDfs
Configure
CrashRecoveryKit
Dobackup
iBackup
Imprison
IPTablesTutorial
OMNIDrivers
Panoptis
PCIUtilities
QuarantineFirewall
RottLog
RunQ
SNARE
Socat
ssync

RPM management and installation
CAPI driver for Fritz!PCI, Fritz!Classic and Fritz!PnP
Detects login attempts on SSH and advises via a GUI popup
Exports all tracks and boot images on a CD
Eases the task of configuring many packages
Tool that can recover your crashed PC.
Automated incremental backups of multiple servers
Automates (with cron) the backup of system configurations
Imprisons processes to jail
How to install and set up iptables and netfilters
Support for over 400 printers
Tool to detect and stop DoS/DDoS attacks
Various utilities dealing with the PCI bus
Firewall with masquerade, TOS and traffic shaping
Replacement to Red Hat's logrotate
Computer performance management tool
System intrusion Analysis and Reporting Environment
Relay for bidirectional data transfer
Minimalistic filesystem synchronization utility

Coverdisc

Neil Bothwick is your guide through the wonders of this month's jam-packed *Linux Format* DVD. Hold on to your hats, here we go...



Once again, we have plenty of bonus programs for our DVD readers. In addition to including many programs in alternative package formats, for ease of installation, we have over a hundred programs that aren't on the CDs at all. This includes a complete distro, the high performance *Beehive*. There is also *Gentoo*, an extremely useful file manager. Don't think it's all boring productivity stuff — you get several games that don't appear on the CDs, some useful resources for developers, the latest beta of the *Opera* web browser and plenty of other useful, interesting or just fun-to-use software. Read on for a few highlights then load

up the DVD's index file for the full listing.

Desktop/Gentoo

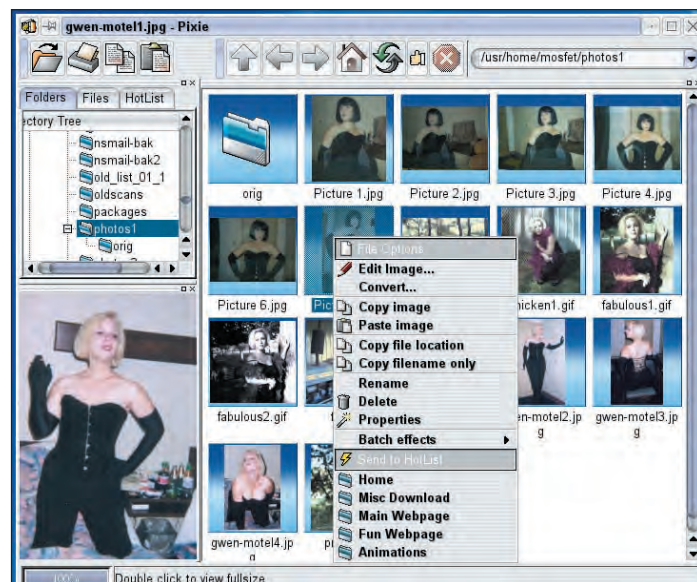
Creating a DVD and two CDs each month involves more than a little file management. With 5-6GB of files to sort into directories, I need a filemanager that is quick and easy to use, while letting me see what is going on. I've been using *Gentoo* for the job this month, and it seems to fit the bill very well.

The layout is the classic two-pane look. It reminds me of *DirWork*, a file manager I used on the Amiga for several years. That's by no means a bad thing; *DirWork* was a very capable file manager, if a little uninspired visually. The same can be said of *Gentoo*. The GUI is somewhat spartan, but it lets you get on with things without distracting eye candy. *Gentoo* is extremely configurable, and the range of options is immense. This isn't the sort of program you set up in a few minutes. It's more a case of the setup evolving to meet your needs: you'll add or change the way things are done as you learn how to use it to best effect.

In such a situation, it is important that the default configuration, your starting point, is useful. Thankfully, *Gentoo* does have a very workable set up out of the box. You don't need to do any configuration to use it, but you will want to as you explore its possibilities and discover better ways of doing what you want to do.

Distros/Beehive

We have featured this distro before, but it was several months and a number of updates ago. There is another reason for featuring it now. Apparently this is the last version of *Beehive* that will be available for free download. *Beehive* are switching to a



Organise your pictures with Pixie Plus (these belong to the developer).

subscription system to cover their server and bandwidth costs.

This is not a distribution for the Linux newcomer, as they say on their web site. "*Beehive Linux* is not for the inexperienced, or those new to linux/*nix. *Beehive Linux* is for people that know what they're doing and want to get the job done as well as possible in the least amount of time."

So, if it's not particularly easy to install, then why bother when there are so many distros that are? Quoting *Beehive* themselves again, "*Beehive Linux* is a distribution made by system administrators, for system administrators. Its intent is to provide fast and clean setup of workhorse servers and workstations. If you're looking for wizards and whizbang gizmos, you are in the wrong place. If you want to set up servers with the services you and/or your users need, you are in the right place". In other words, this is a fast lean distro for those who know what they're doing and want to get it done as quickly and with as little fuss as possible. That doesn't mean that *Beehive* is only for

servers, as they go on to say, "*BlackBox* and *KDE* are included - this is not the primary focus of *Beehive*, but, hey, every admin needs a workstation as well, right?"

An ISO image of *Beehive*'s single CD is included on the DVD.

Graphics/PixiePlus

Do you have a digital camera? If you've had one for any length of time, your hard drive is probably getting as cluttered with photographs as the bottom drawer of the sideboard used to in pre-digicam days. Fortunately there's a solution to help you keep things organised (on your hard drive, not your sideboard). *PixiePlus* is a file management program especially for images and photos. The file browser uses thumbnails of each image so you can quickly find the picture you are looking for.

PixiePlus is not just an image cataloguing program; the file browser is also able to move pictures by drag and drop. There is also a range of image processing tools. It is not a



Wherever you see this logo it means there's related stuff on the DVD

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.

Extra packages

Even with two cover CDs, space is limited if we want to bring you as wide a choice as possible, especially if one CD is entirely given over to a single package or distribution. Therefore we normally only include packages in tarball form on the CD, while the DVD also contains RPM or Debian packages whenever they are available.

replacement for *The Gimp*, but it does provide the basic functions most often needed with digital camera photos, such as scaling, cropping and altering brightness, colour saturation and contrast. In all, there are some thirty image processing effects available in *PixiePlus*. Many of these effects can be applied to a batch of images, saving time and effort.

Once you've found the picture you want and made any adjustments necessary to improve it, you may want to share it with others. *PixiePlus* will automatically scale pictures when printing, so that they fit the page, and dither the colours if your printer driver is one of those that don't handle this.

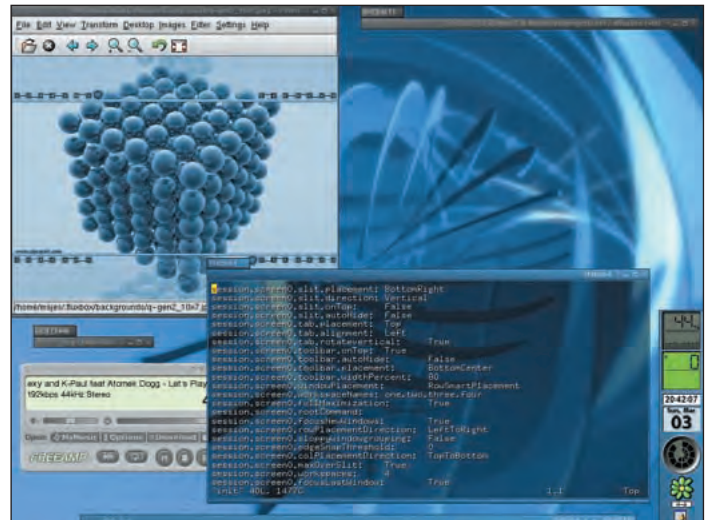
This version has been ported to *KDE3*, but it will work with *KDE2* too. Note that the RPM in here is only for SuSE. If you use a different distro, install from the tarball not the RPM. You will need the various *KDE* development RPMs installed to be able to compile from source. *KDE3* users can use the ones in the Desktop/*KDE3* directory of the DVD.

Desktop/Fluxbox

With all the coverage that *KDE* is getting this month, it's easy to forget that there are other window managers, and that not everyone wants such an all-encompassing environment as *KDE* offers. Many people prefer a more lightweight approach or have no choice if they are using older hardware. Lightweight doesn't have to mean basic or limited, though, as *Fluxbox* shows.

The *Fluxbox* window manager is highly configurable and has many useful features, including such things as configurable window tabs, an icon bar (for minimized and iconified windows), configurable titlebars (placement of buttons, new buttons, etc.) and a native, integrated key-grabber. It has *KDE* support, so you can still run applications written for *KDE*, but the *GNOME* support is currently incomplete.

Lightweight doesn't mean boring to look at either. *Fluxbox* supports themes, and there are many of these available on their web site at SourceForge. We've included several on the DVD, too. (The *fluxbox-theme-*.deb* files are for Debian; other distros should use *sidpack.tar.bz2*.) Because any new window manager is installed in addition to your existing set up, you will be able to try out both *KDE3* and *Fluxbox* this month while still keeping your old *KDE2* set up as a fallback, although I doubt anyone will continue using *KDE2* after trying these two excellent alternatives.



Fluxbox is a lightweight yet highly configurable window manager.



FunkYou is a complete DJ mixing and control console.

Sound/FunkYou

FunkYou is a DJ's mixing console. It mixes, pitches and cross-fades input from MP3s, WAVs and audio CDs. It features a jingle player, a jog dial, loops and pre-listening with a second sound card. It also has an integrated light controller, able to control lights via an RS-232 (serial) port or DMX. *FunkYou* was designed for use in a working environment, so ease of use was a prime consideration; it even supports a touch screen.

Internet/NetComics

There are certain rituals that have to be observed when you arrive at the office in the morning: getting your first cup of coffee, checking your email, logging onto IRC – anything to delay the moment when you have to actually do some work. One of the

more important tasks is to check various online comic strips. You can't seriously be expected to start work until you've read *Helpdex* while drinking that cup of coffee. Of course, this means that everyone in the office is hitting the same sites at the same time, so here's how you can get ahead.

Netcomics is a set of Perl scripts that will download comic strips and store them for later viewing. You can run it as a cron task and have the strips waiting for you when you arrive at the office. *Netcomics* has plenty of options. You can choose, for example, to download only Saturday's and Sunday's strips on Monday and leave Monday's until the next day, to avoid "humour overload"; you can specify which comic you want to download and when; whether to archive them; and a whole raft of other options. Just make sure you leave enough time to actually get some work done. **LXF**

Finding a particular program

How to find that needle in the haystack

With around three hundred packages on the DVD every month, regular buyers of *Linux Format* will soon build up a huge collection of programs. This is great – until you need to find a particular package. Trawling through DVD after DVD would be very time consuming, and, after all, computers are supposed to help us save time, or so they keep telling us. In the essentials directory of the CD you'll find a subdirectory called "CSV". As the name implies, this contains CSV (Comma-Separated Values) files listing the contents of every CD and DVD since issue 21 (we hope to add the earlier ones at some time). These CSV files can be imported into almost any database program on the planet, or you can use

the CSV Reader project from the Kylix tutorial to read and search them.

Each line contains five fields, the name of the disc the program is on, the title of the package, its path, a description and a link to the program's home page (if available). The DVD index files contain only those programs that appear on the DVD but not that month's CD, so DVD users should search both sets.

As these are plain text files, you don't need to go to the trouble of importing them into a database for to perform a simple search. The command `grep -i packagename Essentials/CSV/*` will output a list of all the occurrences of the named package on all CDs and DVDs.

LinuxFormatCoverdiscDVD

» DVD CONTENTS AT A GLANCE

Desktop

ArgyllCMS	Experimental ICC-compatible color management system
CDargs	Adds bookmarks and a simple filebrowser to the cd command
Darkstat	Ntop-workalike network statistics gatherer
Fluxbox	Lightweight and highly configurable window manager
Gentoo	File manager using GTK+ and the two-pane concept
gkleds	GKrellM plug-in to monitor the Num/Caps/ScrollLock keys
GnomePredict	Real-time satellite tracking program for GNOME
GTKtalog	Easy-to-browse CD-ROM database
GtkThemeSwitch	Command line utility to switch GTK themes on the fly
KAlarm	Display messages or execute commands at scheduled times
KDEPopper	Messaging over Samba
LEDblink	Notifies certain events by blinking keyboard LEDs
MOL	Virtual machine which runs MacOS on top of LinuxPPC
PandaPDFGenerator	Make PDFs on the fly for the Web
PictureMonitorApplet	Displays a picture in the panel from the 'Net or hard drive
Rolodap	Multiuser contacts management system using an LDAP server
SensorsLCD	Sends lm_sensors data to a LCDproc server
Waffle	Waffle's A Fast File Level Explorer
WeatherPlotter	Gather, track and look at historical weather data
XMLblaster	XML-based MOM (Message-oriented Middleware)
ZaurusUnitConverter	Unit converter designed for the Zaurus SL-5500 PDA
ZavalFileSearch	Fast file search on SMB shares and local FTP servers

Development

ApplDevelopmentLib	Library of useful programming tools written in C++
AddressParserLibrary	Complete parser for RFC822 addresses
Bond	Rapid application development tool
BondDB	Object-oriented wrapper for PostgreSQL
FOX	Popular C++-based toolkit for GUI development
GNUScientificLibrary	Collection of routines for numerical computing
Libdnet	Portable interface to low-level networking routines
Libpng	Routines to create and manipulate PNG graphics files
Libtext	Library for performing operations on growable text
LinuXMultiMediaProject	
	Generic, powerful API for all kinds of multimedia
OpenCyc	Open source version of the Cyc technology
phpmailer	Email transport class with multiple file attachments
PHPToolbox	Makes creating database-powered Websites easier
Regina	Implementation of the ANSI Standard for the Rexx language
RPC-XML	Perl classes for implementing XML-RPC services
ScriptBasic	ScriptBasic is an interpreter for the BASIC language
XMLtools	High-level tools to help using XML in Python
Zoinks	Programmer's editor and development environment

Distros

2DiskXwindowLinuxSystem	Small Linux distribution
Beehive	Fast, simple, secure, i686-optimized Linux distribution
GibraltarFirewall	Debian-based router/firewall distribution

Games

aBridge	Play bridge online with other real people
BatallaNaval	Multiplayer, networked BattleShip game
GNOME-Maze	Get the Tux out of the maze
PackAttack	Move falling boxes into a row to get the best high score
PythonChess	Play chess against the computer
RARS	The Robot Auto Racing Simulator
Reaper	Graphics-intensive 3D spaceship game

Graphics

AxiomPhotoNewsGallery	
	Photo and news gallery written in PHP
BINS	Photo album generator with internationalization support
ImageMagick	Automated and interactive manipulation of images
IMGV	Cross-platform, open source image viewing application
PixiePlus	Efficiently browse, manage and view large numbers of images

VISCACameraControlLib

	Control library for VISCA-compliant video cameras
vphotoalb	Web application to display photos organized in collections

Internet

BKFTP	Easy FTP command line access and a minimal graphic UI
Freenet	Peer-to-peer network
IRCServices	Services for IRC Networks
Keystory	Gathers data on the usage of OpenPGP signatures
LycoSMS	Send free text messages to any UK GSM mobile phone
NetComics	Retrieves daily comic strips from the web
nget	Command line NNTP file grabber
OpenVPN	Robust and highly configurable VPN daemon
Opera	Alternative, lightweight, X11-based web browser
PPWizard	Powerful and easy-to-use HTML preprocessor
SmartCacheLoader	Configurable web grabber with special smart cache support

Office

ChicoDigitalWebTool	Create a tree structure of web pages without knowing any HTML
Gquotes	Financial monitoring tool that fetches quotes from the web
HyperCal	HyperCal is a web-based calendar.

Server

Apache-ContentHandler	Framework for creating mod_perl and CGI web applications
BitmechanicSpindle	Web indexing and search tool
BSDftpd-ssl	RFC2228 "FTP Security Extensions"-compliant FTP server
CapibaraURLCloakingKit	A reliable URL cloaking system
LDAPtoDNSgateway	Exposes part of an LDAP directory using DNS
Mailman	Manage email discussion lists, like Majordomo and Smartmail
PHP-FileUploadClass	Upload images and text files through a web browser
PortFwd	Forwards TCP connections and UDP packets to remote hosts
Resin	Java application server and web server
TR-IRCD	Ircd and services programs for IRC networks
UMPLforXmail	Web frontend for managing Xmailserver
WebPackageSurfer	PHP-based package management system

Sound

Bewdy	GUI to visualize Maaate's capabilities
Funkyou	MP3-mixing, light controlling, samples and a cross-fader
Grip	CD player, CD ripper and MP3-encoder for the GNOME desktop
MPEG_Maaate	MPEG audio analysis toolkit
Rexima	Curses-based interactive mixer
Rio500	Kernel module, static library and command line utilities
SoftBeep	Redirect beeps from internal PC speaker to a soundcard
tk500	Utility program for the portable Yaesu VR-500 receiver
Xtunes	Comprehensive digital music system

System

BootScriptor	Interactive booting from a CD-ROM drive
GimpPrint	High quality printer drivers for Unix/Linux
grsecurity	Complete security system for Linux 2.4
HoratioAuthenticatedAccess	A firewall authentication tool
IPBandwidthWatchdog	Pcap-based traffic monitor
LinuxKernelPatches	Some useful and interesting kernel patches
Nessus	Remote security scanner for Linux/Unix
NomadII-DriverUtilities	Driver for the Creative Nomad II, IIc and II MG
PPart	Generate system recovery bootable CDs from hard disks
rsync	Replacement for rcp (and scp) that has many more features
ShorelineFirewall	Iptables-based firewall for Linux systems
Syslinux	Boot loader that operates off of MS-DOS floppies
Tomsrtbt	The most Linux on one floppy disk ever
UtilLinux	Suite of essential utilities for any Linux system
Webmin	Web-based interface for system administration
WOL	Wakes up boxes with wake-on-LAN Ethernet cards

Help wanted

This month we ask you to put another feather in Apache's cap; give a boost to a Java 3D engine; talk statistics with Python; or get your computer reading the classics.

We return to good old-fashioned coding this month, though you can be sure that we will have more huge general projects again in the future. For non-coders with (computer) time on their hands we do have one ambitious project in the form of an attempt to audio-enable the massive Project Gutenberg.

We start with another attempt to live with the obfuscated standards of the Windows world.

Andrew Oliver writes:

"The POI project consists of APIs for manipulating various file formats based upon Microsoft's OLE 2 Compound Document format using pure Java.

OLE 2 Compound Document Format based files include most Microsoft Office files such as XLS and DOC.

POI stands for Poor Obfuscation Implementation. Why would we name our project such a derogatory name? Well, Microsoft's OLE 2 Compound Document Format is a poorly conceived thing. It is essentially an archive structured much like the old DOS FAT filesystem. Redmond chose, instead of using tar, gzip, zip or arc, to invent their own archive format that does not provide any standard encryption or compression, is not very appendable and is prone to fragmentation.

POIFS is the oldest and most

stable part of the project. It is our port of the OLE 2 Compound Document Format to pure Java. It supports both read and write functionality. All of our components ultimately rely on it by definition. Please see the POIFS project page for more information.

HSSF is our port of the Microsoft Excel 97-(2002) file format (BIFF8) to pure Java. It supports read and write capability. Please see the HSSF project page for more information.

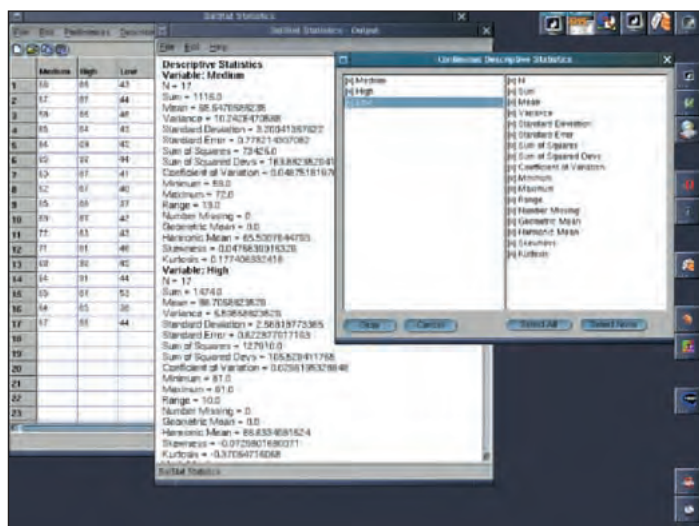
HDF is our port of the Microsoft Word 97-(2002) file format to pure Java. It supports read and write capability. Please see the HDF project page for more information. This component is in the early stages of design, so jump in!

HPSF is our port of the OLE 2 property set format to pure Java. Property sets are mostly used to store a document's properties (title, author, date of last modification, etc.), but they can be used for application-specific purposes as well. Currently HPSF supports read functionality only. Please see the HPSF project page for more information.

We primarily need people to help us document, test and develop the binary ports. However we also have a logo contest and other ways to help as well. So please check the POI project out at <http://jakarta.apache.org/poi> and join the developer's mail list by emailing poi-dev-subscribe@jakarta.apache.org

Salstat

Statistical analysis for scientific data



Salstat showing a resizable datagrid and some of the possible outputs.

Alan Salmoni writes:

"This is a package for the statistical analysis of scientific data (like SPSS). Currently it is in the Alpha stage – it starts, looks pretty, saves and loads ASCII files, does some descriptive

statistics, but not much else – which isn't surprising as I have only been coding for under three weeks. There is also a page on Freshmeat for it (and I even have a subscriber!). By the end of the week, it should be doing

simple tests like t-tests.

I am looking for help – as it is written in Python (and wxPython), Pythonistas are welcome (the wxPython skills are not needed, as the interface is almost finished). However, I am mostly looking for good coders with experience of statistical analysis (things like multi-factorial analysis of variance, multiple regressions and the such like). Knowledge of Python is unimportant, as long as they can communicate their knowledge to someone who knows Python.

The package will also run on Windows, any Unix that has wxPython ported across, and Mac – when the Mac port of wxPython is completed – so it could be a truly cross-platform application! The licence is the GPL, so it is free in the *libre* sense. The home site has lots more info, downloads (such as they are), some docs and Linux screenshots (on *WindowMaker*) for the trusting!"

<http://salstat.sunsite.dk>

View from aBridge

Get online for a rubber

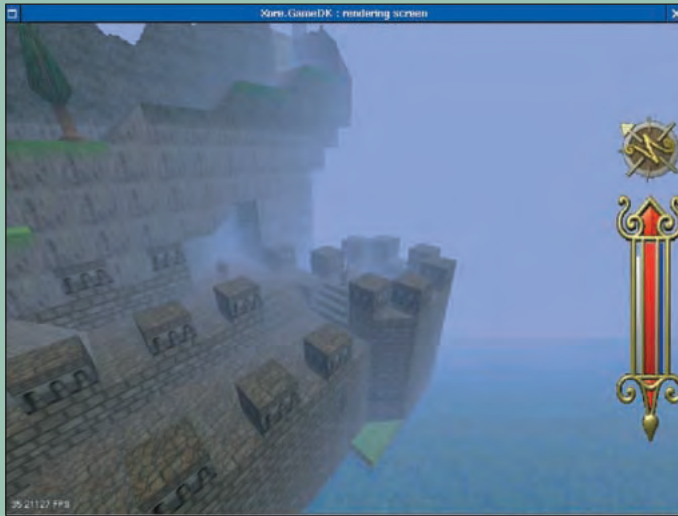
aBridge, the online bridge game, does happen to need a little help for adding new features, but the specific reason we're mentioning it here is that one of the LXF staff feels that there are never enough people online, at convenient times, to play against. Therefore we ask you to lay down your (coding) tools for a while, pour yourself a gin 'n' tonic and join the Reviews Editor online in a game or three.

www.abridgegame.org

Meanwhile, if YOU have a project that's in need of anything from artists and beta-testers to web-designers and, er, something beginning with Z, we want to hear about it. Email us now at linuxformat@futurenet.co.uk and give us some details of your project, and what sort of help you are looking for. Please include plenty of info about the project!

OpaleSoya

Java 3D gaming engine



Arkanae – based upon Opale.Soya, and nearly complete.

Opale.Soya II, is a fast, high-level and OO 3D engine for Java designed with 3D games in mind. You may have tried it out from last month's DVD. The project uses GL4Java (<http://jausoft.com/gl4java/>) to access native OpenGL.

The base of the 3D engine works quite well, but complex features (BSP tree, or the editor) are not finished, or not enough tested.

In fact, what Jean-Baptiste Lamy, the project originator, really claims to need is a maintainer for the project.

The "job" requires some knowledge of OpenGL/GL4Java and of Java (object-oriented programming, including concepts like event), and a lot of motivation!

Opale.Soya has a serious TODO list, take a look and then roll up your sleeves and dive in:

- Testing the modeling package – in particular, optimised faces whose vertices are not in the same coordinate system have never been tested;
- Morphing;



Fancy creating a level? Go to the site and find out how.

- Support standard file format (VRML, X3D, 3DS ?) in addition to Java serialization;
- Better animation package, in order to animate lights, fog & co;
- Landscape: connecting Heightmap3D to create infinite landscape;
- New FX:
 - Sky sphere,
 - Bumpmapping,
- Improve the Editor – in particular:
 - Add a window displaying the texture for selecting texture coordinates,

- Allow to break down optimised groups of faces;
- Java 1.4 NIO optimisation and fullscreen;
- Add more tutorials;
- Port to other OSs (MacOS, FreeBSD, ...);
- ... (open list :-)

You might also like to check out *Arkanae* – a near-complete 3D rôle-playing game using Opale.Soya. <http://opale.soya.tuxfamily.org/join.php/> <http://arkanae.tuxfamily.org/>

Listening to Gutenberg

Talking books and great music

Mike Eschman, of Engima

Technologies, has been sponsoring a project to audio-enable books from Project Gutenberg, using *emacspeak*/*ViaVoice* technology. From his working production facility he can manage 9–12 books a year without funding. Funding would enable more books to be audio-enabled, but so would donating computer time on Linux systems with ftp access to the Internet.

"All offered works are indexed by a search engine and a Table of Contents. The books are printable. The product packages a chapter of a classic Science Fiction, Action, Novel or Play with licensed music @ mono/11025/.mp3 as an overture and sometimes an epilogue. Each Chapter or section gets 30 minutes as an entertainment.

We have packaged 20th Century jazz and classical music appreciation as a thematic partner to the study of the written work," says Eschman.

"The product can be delivered in .crd, .mp3 and .wav. *Emacspeak* on a laptop acts as a portable reader. Batch processing for *ViaVoice* is operational. We also have a working mix-down desktop. We expect vcd with scrolling text and dolby 5.1 sound (first one, then the other) this year."

This collection fetches a one-time fee per book, paid by user institutions. These institutions have unrestricted, non-exclusive rights to a book. These funds are used to support the team creating the Gutenberg audio library.

To help edit books you will need *ViaVoice* – the *ViaVoice* used is the

Outloud! component included in the free Linux download of *emacspeak*.

Texts are edited to incorporate *ViaVoice* command lines which set the voice characteristics. The modified .txt file is then processed by a batch procedure which creates an audio file in .au format. Using a tool such as *audacity*, you listen to the audio product, marking problems (typically strange intonation or mispronounced words) on a print out of the .txt file. Changes are made and the batch process run again, as many times as necessary to create a useable audio product. Only then do you proceed, by exporting an initial .wav file.

The file is then processed through *sox* to obtain a mono 22050 .wav bandwidth limited to the 80–7,800Hz

frequency spectrum with white noise masking added. The new file is then processed by *lame* to create an .mp3 for (Internet) broadcast and download.

Alternatively, if you have a good voice, download a book and read it out, recording your reading as a .wav or .mp3 file.

Help is needed editing gutenberg books for *ViaVoice* processing – as described above; with reading and recording Gutenberg books "the old-fashioned way," as an alternative; and providing processing for books and/or providing ftp storage of audio books. Donations of recorded music by the artist holding the copyright of the recorded performance are also welcome.

<http://www.engima.com>

User Groups

Your local Linux User Group needs you! LUGs worldwide are full of members keen to help with your problems, discuss ideas and generally natter about all things Linux. We have collected a load of information here so you can find the LUG closest to you. You can find lots more information online at: www.lug.org.uk or <http://lugwww.counter.li.org/groups.cms>

1 Hampshire

URL www.hants.lug.org.uk
Contact Hugo Mills

2 Bristol & Bath

URL www.bristol.lug.org.uk

3 Scottish

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

4 Oxford

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

5 Kent

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

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.doncaster.lug.org.uk
Contact Andy Smith

12 Moray

URL www.moray.lug.org.uk
Contact Stewart Watson

13 West Wales

URL www.westwales.lug.org.uk
Contact Dan Field

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 <http://gl.lug.linux.co.uk/>
Contact John Southern

20 Surrey

URL www.surrey.lug.org.uk
Contact Jay Bennie

21 Cambridge

URL www.cam-lug.org

22 Devon & Cornwall

URL www.dclug.org.uk
Contact Simon Waters

23 Falkirk

URL www.falkirk.lug.org.uk

24 Manchester

URL www.manlug.mcc.ac.uk
Contact John Heaton, Owen Le Blanc

25 Hertfordshire

URL www.herts.lug.org.uk
Contact Nicolas Pike

26 West Yorkshire

URL www.wylug.lug.org.uk
Contact Jim Jackson

27 Sheffield

URL www.sheflug.co.uk
Contact Richard Ibbotson

28 Staffordshire

URL www.staffslug.org.uk

29 North East

URL www.shofaruklinux.net/NELUG

30 London

URL www.lonix.org.uk

31 Thames Valley

URL www.sclug.org.uk

32 Liverpool OpenSource

URL http://linux.liv.ac.uk/_liv_linux_ug/
Contact Simon Hood

33 Deal Amiga Club

Email superhighwayman@hotmail.com
Contact John Worthington

34 Chesterfield

Email spirelug@yahoo.co.uk
Contact Robin Needham

35 South Derbyshire

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

36 Belfast (BLUG)

URL www.belfastlinux.cx
Contact Ken Guest

37 Wiltshire

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

38 South London

URL www.sl.lug.org.uk
Contact Ben@benguin.co.uk

39 Cheshire

URL www.sc.lug.org.uk
Contact Anthony Prime - enquiry@sc.lug.org.uk

40 North Wales

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

41 Midlands

URL www.midlandsLUG.cjb.net WARNING: Popup ads
Contact Pete Thompson

42 Cumbria

URL www.cumbria.lug.org.uk
Contact Jamie Dainton

43 Dorset

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

44 Shropshire

URL www.shropshire.lug.org.uk
Email shropshire@lug.org.uk

45 South West

URL www.southwestlug.uklinux.net
Email southwest@lug.org.uk

46 South Wales

URL www.sw.lug.org.uk
Contact Tim Bonnell

47 North London

URL <http://www.kemputing.net/lug/anlug-aims.html>

48 Malvern

URL www.malvern.lug.org.uk
Contact Greg Wright

49 Huddersfield

URL www.hud.lug.org.uk
Contact Adam Brookes

50 Nottingham

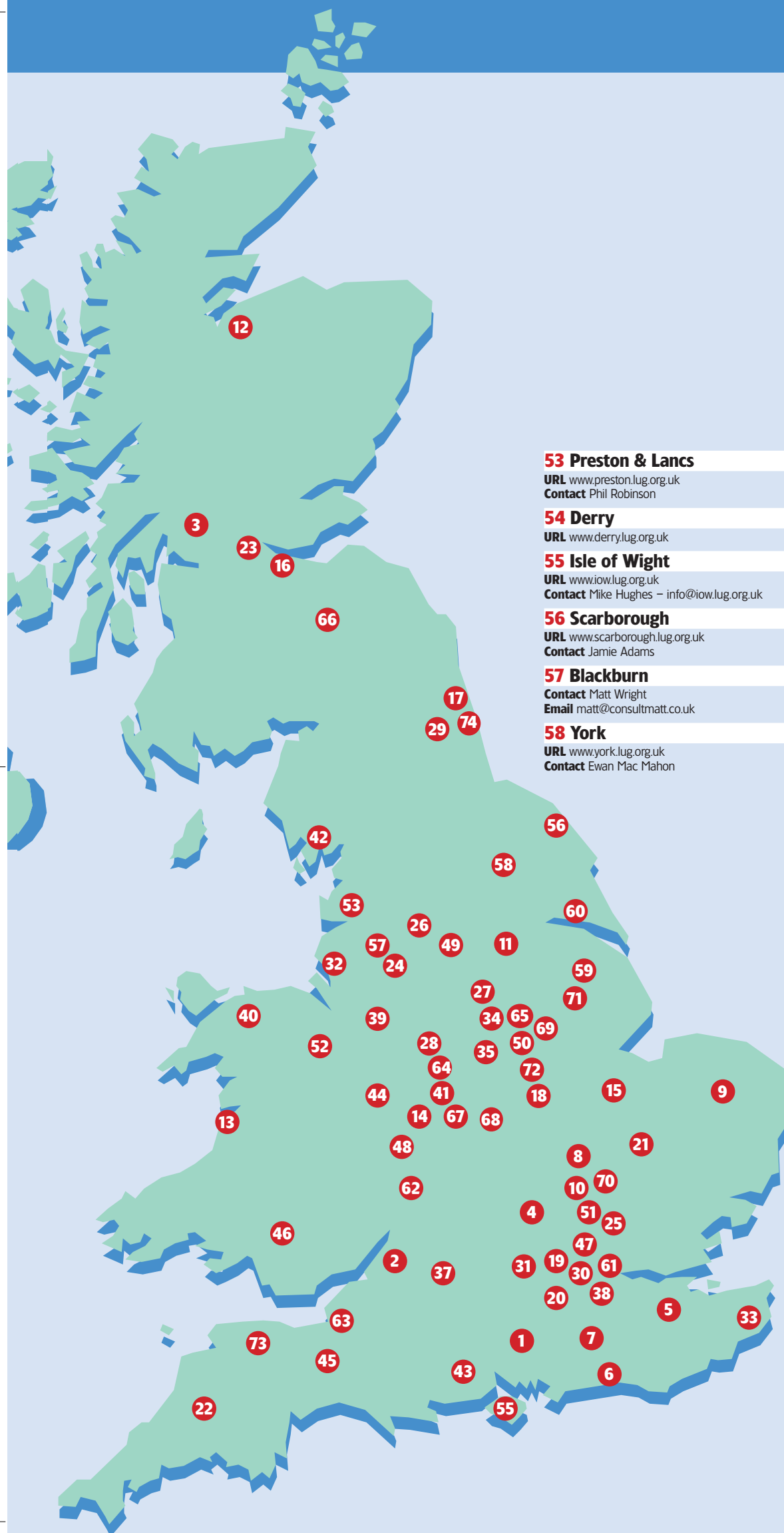
URL www.nottingham.lug.org.uk
Contact Godfrey Nix

51 St Albans & Luton

URL www.lust.lug.org.uk
Contact Michael Culverhouse - mike@easily.co.uk

52 Wrexham

Contact Paul Kersey-Smith
Email paul@pkls.fsnet.co.uk

**53 Preston & Lancs**

URL www.preston.lug.org.uk
Contact Phil Robinson

54 Derry

URL www.derry.lug.org.uk

55 Isle of Wight

URL www.iow.lug.org.uk
Contact Mike Hughes – info@iow.lug.org.uk

56 Scarborough

URL www.scarborough.lug.org.uk
Contact Jamie Adams

57 Blackburn

Contact Matt Wright
Email matt@consultmatt.co.uk

58 York

URL www.york.lug.org.uk
Contact Ewan Mac Mahon

59 Lincs

URL www.lincs.lug.org.uk
Contact Chris Lingard

60 Hull

URL www.hull.lug.org.uk
Contact Chrus Burton – chris@hull.lug.org.uk

61 East London

URL www.eastlondon.lug.org.uk
Contact Jonathan Spriggs

62 Gloucestershire & Cotswolds

URL www.gloucs.lug.org.uk
Contact Barrie Haycock

63 Yeovil College

URL www.ycplug.lug.org.uk
Contact Adam Parker

64 South Staffordshire

URL www.staffs.lug.org.uk
Contact Oliver Keenan

65 Mansfield

URL www.mansfield.lug.org.uk
Contact Brent Vardy

66 Borders

URL www.linux.bordnet.co.uk
Contact Welby McRoberts

67 South Birmingham

URL www.sb.lug.org.uk
Contact Tim Williams

68 Coventry

Contact Darren Austin
Email info@coventry.lug.org.uk

69 Newark & Lincoln

URL www.newlinc.lug.org.uk

70 Bedfordshire

URL www.beds.lug.org.uk
Contact Neil Darlow

71 Lincoln

URL www.lincoln.lug.org.uk
Contact Jon Shamash

72 Loughborough

URL www.loughborough.lug.org.uk
Contact Martin Hamilton

73 Exeter University

Contact Nicholas Murison
Email N.J.Murison@exeter.ac.uk

74 Sunderland

Contact Thomas Croucher
Email thomas.croucher@sunderland.ac.uk

**NEW
DETAILS**

LinuxUserGroups

LUG OF THE MONTH!

Greater London

The Greater London Linux User Group is simply the best user group in the country. This is down to the mix of members, which consists of sysadmins, developers and humble users. Based in the centre of London, it regularly has 80-100 members turn up for meetings held on Saturday afternoons.

We run a popular mailing list which is very friendly with a few

off-topic postings, but mainly technical help, and an IRC channel for instant online chat. The group members have wide ranging abilities from newbie beginners to the most experienced gurus. Lectures are held with an emphasis on technical solutions, and members are encouraged to bring along equipment for problem solving. After meetings most members



usually partake in robust discussions at one of the many fine local hostelrys.

There are no formalities to attending a GLUG meeting, no subscription, age restrictions or

entry fees, you can just turn up on the day. We welcome and encourage all Linux users, from the inexperienced to hardened IT professionals and gurus.

<http://gllug.linux.co.uk>

You can email your submission for "LUG of the Month" 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.

Worldwide Linux User Groups

Free Software users across the globe

Australia

ADELAIDE

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 <http://plug.linux.org.au/>

Europe

AUVERGNE

URL www.linux-arverne.org

Email Cyril.Hansen@wanadoo.fr

DENMARK

Alssund www.alslug.dk

Esbjerg www.eslug.dk

Fyns www.flug.dk

Midt-og Vestjylland www.mvjlug.dk

Nordjylland www.njlug.dk

Skåne Sjælland www.sslug.dk

Trekantsområdet www.tlug.dk

Vest-fyn www.haarby-net.dk/vflug

Århus www.aalug.dk

EIRE

URL www.linux.ie

Email root@linux.ie

URL www.dilu.org

Email glossary@dilu.org

GOTHENBURG

<http://nain.oso.chalmers.se/LUGG/>

UK

Don't forget the distribution-specific mailing lists:

URL www.lug.org.uk/maillist.html

India

URL www.linux-india.org

Email newsmaster@linux-india.org

TRIVANDRUM

URL www.river-valley.com/tux

Email anil@river-valley.com

Middle East

EGYPT

URL www.linux-egypt.org

Contact Hesham Bahram

North America

ALASKA

URL www.aklug.org

Email deem@wdm.com

BATON ROUGE

URL www.brlug.net

Email dpuryear@usa.net

BAY AREA

URL www.balug.org

Email aftyde@balug.org

CLARKSVILLE, TN

URL www.cllug.org

Email tux@cllug.org

DENVER

URL <http://clue.denver.co.us/>

Email resident@clue.denverco.us

FLORIDA

URL www.flux.org

LOS ANGELES

URL www.lalugs.org

Email dank@alumni.caltech.edu

NORTH COLORADO

Email nclug@nclug.org

Contact Mat Taggart

TAMPA

URL www.suncoastlug.org

Email president@suncoastlug.org

UHACC Normal, IL

URL www.uhacc.org

Email lug@uhacc.org

VIRGINIA TECH

URL www.vtluug.org

Email nega@vt.edu

South America

BUENOS AIRES

Email dcoletti@impost.com.ar

CHILE

URL www.linux-chile.org

MONTEVIDEO

URL www.linux.org.uy

PARAGUY/ ASUNCION

Email rolgiati@conexion.com.py

SAO PAULO

URL <http://gul.ime.usp.br/>

Email gul@ime.usp.br



Bristol install day

April 20th was not a day to be enjoying the sun – there was work to do installing and demonstrating Linux.

Bristol IT Co-op held an Install Day

as part of it's initiative to recycle old hardware into useful *nix boxes. Sean Kenny writes: "The Linux Install festival at Windmill Hill City Farm in Bristol seemed to be an enjoyable day for everybody involved. Business was brisk from early in the day, in fact several members of Bristol & Bath Linux Users Group turned up early and gave the organisers a much needed hand. There was a rush in the morning, a lull at lunchtime, and then another rush in the afternoon. People were still arriving up to us closing the doors at 4pm. We estimated around a hundred attendees throughout the day.

"We set up a Red Hat based LTSP server and were able to demonstrate on several machines that were brought in by attendees – including video and sound. Several participants set up a Smoothwall workshop to show cheap firewall technology.

"We had representatives from several largish organisations in the city, who are considering moving to Open Source models for their IT needs, they went away with most of their questions answered. Highlight of the day – the Zaurus PDA, and the in-car MP3 player. Most popular question – when and where is the next one happening?"

<http://www.bitcoop.co.uk/>

Linux User Group organisers

If you're not listed here, or we have your details wrong, please contact us at: **LUGS!, Linux Format, 30 Monmouth Street, Bath, BA1 2BW** or email your details to: linuxformat@futurenet.co.uk

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** Richard Smedley richard.smedley@futurenet.co.uk**Editorial Contributors**

Andy Channelle, Jono Bacon, David Cartwright, Hoyt Duff, Charlie Stross, Maurice Kelly, Paul Cavanagh, Chris Brown, Dave Cross, David Coulson, Brian Long.

COVER CD PRODUCTION**CD Editor** Neil Bothwick**Senior CD Editor** Kate Hadley**ART CONTRIBUTORS****Photography** Louise Parker, Amanda Thomas, Getty Images**Illustration** Paul Bateman, Shane Collinge**ADVERTISING SALES**

Matt Stanley 01225 442244

matt.stanley@futurenet.co.uk

Emma Mimmack 01225 442244

emma.mimmack@futurenet.co.uk**MARKETING AND PROMOTIONS****Marketing Manager** Fiona Tully**Subscriptions Executive** Nick Skardon**PRODUCTION****Production Co-ordinator** Diane Ross**Group Production Manager** Clare Tovey**Advertising Production Co-ordinator** Jo Crosby**MANAGEMENT****Publisher** Dave Taylor**Group Advertising Manager** Simon Moss**Group Publisher** John Weir**Circulation Director** Sue Hartley**Managing Director** Colin Morrison**DISTRIBUTION AND CIRCULATION****Circulation Manager** Jamie Malleyjamie.malley@futurenet.co.uk**Distributed by** Seymour Distribution, 86 Newman Street, London W1T 3EX**Tel** +44 (0)207 3968000**Overseas Distribution** by Future Publishing Ltd.**Tel** 01225 442244.**Overseas Licences**Simon Wear simon.wear@futurenet.co.uk**Tel** +44(0)1225 442244 **Fax** +44 (0)1225 732361**Contact Details**

Linux Format, 30 Monmouth Street, Bath BA1 2BW

Tel +44 (0) 1225 442244 **Email** linuxformat@futurenet.co.uk**Subscriptions and Mail Order****Phone** +44 (0)1458 271178. See page 96**Email** linuxformat.subs@futurenet.co.uk

Copyright No part of this publication may be reproduced without written permission from our publisher. We assume all letters sent – by email, fax or post – are for publication unless otherwise stated, and reserve the right to edit contributions. All contributions to Linux Format are submitted and accepted on the basis of

non-exclusive worldwide licence to publish or license others to do so unless otherwise agreed in advance in writing recognises all copyrights and trademarks. Where possible, we have acknowledged the copyright holder. Contact us if we haven't credited your copyright and we will always correct any oversight.

All CD-ROM demos and reader submissions are supplied to us on the assumption they can be incorporated into a future covermounted CD-ROM, unless expressly stated to the contrary. We cannot be held responsible for mistakes or misprints. Linux Format recognises all copyrights in this issue. Where possible we have acknowledged the copyright holder. Please contact us if we have failed to credit copyright.

Disclaimer All tips in this magazine are used at your own risk. We accept no liability for any loss of data or damage to your computer, peripherals or software through the use of any tips or advice.

Printed in the UK by Midway Clark (Holt) and © Future Publishing Ltd 2002

LINUX is a trademark of Linus Torvalds, GNU/Linux is abbreviated to Linux throughout for brevity.

All other trademarks are the property of their respective owners

Future Publishing Ltd. is part of The Future Network plc.

The Future Network produces carefully targeted specialist magazines and websites for groups of people who share a passion. We aim to satisfy their passion by creating titles that offer superb value for money, trustworthy information, multiple ways to save time and money, and are a pleasure to read or visit. Today we publish more than 80 magazines and over 30 magazine websites and networks from offices in five countries. The company also licenses 32 of its titles resulting in over 60 local editions in a further 23 countries. The Future Network plc is a public company quoted on the London Stock Exchange (symbol: FNET).

Non-executive Chairman: Roger Parry**Chief Executive:** Greg Ingham**Chief Operating Officer & Managing Director, UK:** Colin Morrison**Group Finance Director:** John Bowman**Tel** +44 (0)1225 442244www.thefuturenetwork.plc.ukBATH • LONDON • MILAN •
NEW YORK • PARIS •
SAN FRANCISCO • WROCLAW

NEXT MONTH

Issue 29 on sale Friday 21 June

GETTING PATCHED UP

Patches? We don't need no steenkin patches. Or maybe we do. Virtually every distro comes with a pre-patched kernel. We'll tell you what patches are out there, why you want them and how to apply them to your kernel.

The future of gaming?

A look at the WineX 2.0 release and the technologies behind it, featuring an exclusive interview with it's creators, Transgaming.

PLUS

Reviews of *SuSE 8.0*, *Mozilla 1.0*, a tutorial on setting up *Tripwire*, the lowdown on IPv6, file permissions explained, and lots more.

Including LINUXPRO

Our new mini magazine will be back, with a look at more real-world IT solutions, trends and interviews.



DON'T MISS YOUR COPY OF LINUX FORMAT!

Get it delivered to your door every month – subscribe on page 96

The exact contents of future issues are, sadly, subject to change

LINUXPRO

From the makers of LINUX Format

June 2002

Thinking big

IBM's £2m Linux server – serious hardware, serious application

PLUS:

Georg Greve – Why Free software is good for business

Being Prepared – The real and present danger of Linux viruses

WELCOME

Contacts

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

Richard Smedley
richard.smedley@futurenet.co.uk

Photography

Kath Lane-Sims,
 James Cheadle

Publisher

Dave Taylor

ADVERTISING

Group advertising manager

Simon Moss

Advertising manager

Matt Stanley
matt.stanley@futurenet.co.uk

Deputy advertising manager

Emma Mimmack
emmamimmack@futurenet.co.uk

LinuxPro is a part of



magazine

Published by



Media with passion

30 Monmouth Street,
 Bath BA1 2BW
 Tel +44(0) 1225 442244

Welcome

Twenty-four pages of real-world Linux for IT professionals

A warm welcome to our very first issue of *Linux Pro*, a new mini magazine produced by the team behind the UK's longest running and best-selling Linux magazine, *Linux Format*.

While we're sure that IT Professionals find the content of *Linux Format* as helpful and as practical as everyone else, we thought this mini-magazine would be an ideal place to discuss real-world computing on a larger scale, and deal with the issues, trends and technologies currently shaping the IT world.

One of the first things we wanted to include was a piece on IBM's mainframe Linux. IBM made the headlines by investing over \$1 billion into Linux technologies, but their strategy has paid off, through increased sales of S/390 systems running Linux.

In an exclusive interview with Georg Greve of FSF-E, we find out his thoughts on why free software is good for business, and patents are bad.



Finally, we take a look at that perennial favourite – viruses. Yet to make an impact on Linux, but are we ignoring the threat at our peril?

Finally I'd just like to say we'd appreciate any feedback on what you have seen here, – just email me !

Nick Veitch Editor
nick.veitch@futurenet.co.uk

Contents

IBM: Mainframe Linux in action at Warwick **p3**

GEORG GREVE: The FSFE president on business and free software **p14**

Viruses Are you prepared? **p14**



THINKING BIG:

Mainframe Linux

Richard Drummond meets IBM's Doug Nielson and sees a £2million Linux machine in the form of the S/390 at Warwick University's Computing Science Department.

S/390
Parallel Enterprise Server
Generation 5



INTERVIEW | WARWICK UNIVERSITY/IBM

« **G**iven the chance to see IBM's big iron in action, and learn all about it from Doug Nielson, of IBM's Enterprise Systems Group, *Linux Pro* grabbed a tape recorder and headed up the motorway to Warwick University.

LINUX PRO: How did the Linux port to the S/390 come about?

DOUG NIELSON: IBM's been working with Linux for I guess in total about three years now. We were particularly intrigued by the idea of putting it on our mainframes. The guys in our German labs did the port. They just felt it might be an interesting proposition. The thinking was not that we were going to anything dramatically different, but that because of some of the technologies in the mainframe, particularly the ability to create virtual machines, the *VM Hypervisor*, that it

involved with providing 390 distributions.

We dipped our toes into this three years ago. The guys who did it – clever programmers – did it because it was cool [laughs]. It's a good reason for doing things with Linux, the only business case you need. It's cool, it looks like fun. We did it, started to talk about it on the web and people started to get interested. We're now at a point where it's a major driver in the mainframe market. The mainframe market is booming at the moment. Of course, there are several reasons for that. Some of it is because traditional workloads are going out on the web. But also, for the first time in twenty years, we're seeing new customers coming to the mainframe, and that's mostly to run Linux, mostly to deploy these virtual servers.

LXP: What does Linux bring to the mainframe platform?

DN: What Linux brings is the open-source applications – and a combination of enthusiasm, youth and fun. We've tried lots of things to make the mainframe appeal over the years. We've tried. But it's this combination that's catching the imagination of our customers around the world.

Linux gives us a new combination of fun, speed and youth, that is associated with Linux, with the industrial strength reliability associated with the mainframe. We felt this could be win-win in the market – that the mainframe would gain a lot in image from supporting Linux, as well as gaining a lot of applications – because the whole portfolio of Linux applications would become available. We also felt that Linux itself would gain credibility and respectability in the commercial market place by doing this. And that is indeed how it has panned out over the last two years.

Since then things have been moving fairly substantially. IBM's got about 50 or 60 customers worldwide who have installed mainframes to run Linux. That's all happened in the last 18 months. I'm aware of about ten big, household-name customers here in the UK who are running Linux on the mainframe. Unfortunately I cannot share any of their names with you, because none of

“We also did it as an academic exercise – to prove that Linux was truly platform independent.”

could be interesting. I guess we also did it as an academic exercise – to prove that Linux was truly platform independent. I don't know what the current count of architectures that Linux runs on is – fifteen, twenty, it's of that order – the S/390 was just another one.

So that was done. We started talking to our customers about it in late 1999 and we were surprised at how many were interested to have a look to see what it could do. To begin with, because IBM had done the work to port Linux to the 390, the code was first available for download from IBM. Actually we made it available from a US college called Marist College. Since then, there has been a lot of change in the market place, and we now have mainstream distributors: Red Hat, SuSE, TurboLinux – and there are some specialist ones, but those three big boys are fully

The mainframe Or a bluffer's guide to the IBM S/390 and zSeries



Inside the belly of the monster: buried deep inside the S/390 is a unique and specialised microprocessor.

The S/390 mainframe at Warwick is a direct evolution from the IBM S/360 range introduced in the 60s. The 360 referred to 360 degrees, a full circle: the S/360 was the first general-purpose computer. The term 'mainframe' derived from the fact these machines were installed in racks or frames, and the processor and memory was in the main frame.

The S/360 design was updated with the S/370 in the 70s and S/390 in the early nineties, the latter machine marking a fundamental technology shift. Up until then, the mainframe's processor had used bipolar semiconductor technology – which

was fast, but expensive and hot (the original machines were water-cooled). By the 90s, CMOS technology – which had grown up with the desktop PC – was on a faster growth curve than bipolar, so IBM switched the mainframe to CMOS and designed a unique microprocessor for the S/390. It's a very complex instruction set chip, with built-in error detection and correction. Since IBM need only 10,000 units a year, they can afford to fabricate it with an advanced, low-yield process.

Six generations of the S/390 – G1 to G6 – were shipped throughout the 90s (the mainframe at Warwick is a G5), and in 2000 a new machine was

brought out, called the zSeries (in keeping with IBM's re-badging of its server families), featuring a new 64-bit architecture (the S/390 was 31-bit). The z900 was launched in 2000, and a 'stripped down' version, the z800 in 2001. Linux runs on both the S/390 and the zSeries.

Although, you can run Linux directly on the mainframe, the usual method is to use IBM's virtual server technology, called VM (Virtual Machine). Both Linux and IBM's z/OS (*née* OS/390) can exist as guests on VM, and VM allows the creation of hundreds, even thousands, of separate, co-existing virtual servers on the mainframe.

INTERVIEW | WARWICK UNIVERSITY/IBM



Dr Roger Packwood (seated) and Doug Nielson.



them are yet ready to go public. But they're all names that you would recognize. One of the most recent who we could get to go public was Ford Europe – they're actually based in Germany, but they do deploy services here in the UK. Ford in Germany are running Linux on their mainframes.

The kind of projects that most customers are using this for are what you might call IT infrastructure: web serving, file and print serving, domain name serving. There's also a move towards more mainstream commercial applications as large-scale commercial

packages become available on the mainframe version of Linux.

LXP: What prompted the donation of hardware to Warwick University?

DN: After the port was done, we were very keen to promote that, keen to encourage the platform to grow. One opportunity we saw – we knew universities were important because, to be frank, today's IT graduate is tomorrow's IT director. If you were to go around most universities today and ask about mainframes, you'd get a very blank

response: "Never heard of it," or, at best: "Heard of it, but you know, it's sort of a dusty, fusty technology."

Machines like the one installed here are very much part of a worldwide program – there are mainframes in several universities in the US. The aim is to have graduates who can say, not that they are mainframe systems programmers, but graduates who can say, "Mainframes. I know what they're good for. I know what they're not good for. I've got an opinion. I've heard about it. I've used it"

We wanted to locate a UK university who might be interested in exploring this.

As far as I know, there were no mainframes in any UK universities at that point. There had been – in the 60s, 70s, early 80s – but they'd petered out. This time last year the 390 was installed here, and the *VM Hypervisor* was put on. IBM initially did the installation and put on SuSE. Since then it's been under Roger and his colleagues' control, and they've done their own things with Red Hat and others.

Working with Warwick was very useful for us – it gave us an early start into a new market place. It gave the university a chance to experience and form their own opinions – good, bad and otherwise – about what the technology is, what it's good at, what it's not good at. We knew some of that. We tried to be very up-front with Roger and the department. We said, it's not a super-computer. It's not for intensive numerical calculation. It's good at doing lots of things at the same time. It's not particularly good at doing one thing. You put it up against a big Intel PC, and it may look rather poor in terms of raw performance. But you put 20 or 40 simultaneous workloads on it, you create large numbers of virtual servers, and this is where this architecture starts to score.

And it's the architecture that's being used, of course, in all the large commercial organisations around the world, it's fair to say, although most are running its own native operating system, z/OS (which used to be called OS/390).

LXP: Are there applications that you can run on Linux on the mainframe that you

can't run under z/OS? Or is it just that nobody had bothered porting these sorts of applications?

DN: That's right. The answer is that in most cases that nobody had seen the business case to do the port. To port an Intel – Unix or Windows – application to z/OS is hard. In fact very few people have done it. It's not a port, it's a re-write. But to port Unix, NT, Windows applications to Linux can be quite straightforward.

What kind of applications can we run? Well, everything that comes with a standard distribution: *Samba*, *Apache* and so on. z/OS has analogues of all of those – functionally not dissimilar. There are ways of doing file and print serving, ways of doing web serving. But these are using IBM-developed products that are not open-source, not low-cost. We think that there's a place for both.

If I were a big bank today and I were going to support large numbers of my customers for a secure, continuous banking system, I'd probably be using z/OS for that. However, if I wanted to do something quick and simple like static web serving, I'd use Linux.

What the mainframe has traditionally always

“It's the architecture that's being used in all the large commercial organisations around the world...”

been good at was things like staying up, being constantly available, highly-performing, being secure, being manageable. And, interestingly, this whole transition to doing business on the web is playing back into the mainframe's strengths. If you've got a large scale e-commerce server, then you want it to be available, secure, highly-performing, manageable – all the things that the mainframe was always good at.

Now, interestingly, many of the customers who've grown up on the web and have deployed solutions have come at it differently.



INTERVIEW | WARWICK UNIVERSITY/IBM



They come at it from a PC or a Unix server mind-set. And they've installed a PC. And they've installed another one. Soon, they've got 20, 40, 60 servers. That's fine. You can do it that way. But sometimes the service you can deliver that way is not optimum.

Let me tell you a little story. You know the year 2000 problem? Well, a good thing came out of that. Customers went 'round saying, "I'd better check all my servers are year 2000 ready!" And, indeed, IBM engaged a lot of services, did a lot of work in that area. A customer would come to us and say, "Just check my 400 servers are year 2000 ready, would you?" And we'd come back and say, "Yes, your servers are ready, but you don't have 400, you have 2000!" Most large corporations had totally lost control of their servers. They don't know how many they've got. They don't know where they are. They don't know what they're doing. Very often they have lists of network addresses, but they don't know whether the machine is there or not. The only way to find out who's using a server is to turn it off and see who complains. And very often no-one complains.

This is all about cost, hidden cost. You get people who are meant to be providing a service, who are meant to be customer-facing,

can pick up large workloads. The reason we can do this is that, in a big rack of small Intel servers, some of them are very busy, a lot of them aren't. The average utilisation – we've done a lot of studies on this – the average utilisation of an Intel server in a commercial organisation is just 10%. Part of this is because there are a lot of machines – back-up machines, spare machines, recovery machines, development machines – all lying idle. Of course, PCs don't like being very busy, anyway. They either fall over or don't provide a very good service. But you can take a mainframe with VM and transfer the load of 50 Intel servers to 50 virtual servers on the machine. Because the peaks don't coincide in those workloads, we can absorb all of that on a machine which physically doesn't appear to be much more powerful than the Intel servers it replaced. It can consolidate those workloads.

Believe it or not, the cost of the mainframe is generally dramatically lower than the cost of the racks of servers it replaces. Customers are finding that it's not about hardware or even software, it's about people. Most commercial customers find that they can support about 20 servers per person. Of course, they're will be counter examples, but usually that means if you have 200 servers, you need 10 people; 2000 servers, you need 100. In the mainframe world, you can scale up the mainframe – more memory, more processors, more capacity – and you don't need more people. So the cost proposition for large systems is very attractive. We've seen customers – telcos and ISPs – start to do this.

We have a customer, a telco in Sweden called Telia. One advantage they've found – amongst all the others I've already mentioned – is one we never anticipated. They took out a lot of Sun Unix servers and replaced them with a mainframe. They run a hosting business. In the old days, when they got a new client, they had to get a new server, put it in the rack, cable it, customise it, test it – it took two days. Now they do it on the phone. Press a key and a new virtual server is created – in fact they have a queue of empty virtual servers waiting in the machine. A few more key strokes and it's customised. They can get the server up and in

“The average utilisation of an Intel server in a commercial organisation is just 10%”

who are actually tinkering with PCs, or are worrying about LAN backup or not worrying about LAN backup (I don't know which is worse). We've got all of these hidden costs in the industry.

We are seeing this trend now towards re-consolidation. It's not a mainframe statement – it's true of all architectures: many Intel servers are being consolidated into fewer SMP Intel servers, for example. Mainframes tend to act as the ultimate consolidator, because they



**Warwick University
has lots of little Linux
boxes, too.**

business before they can process the guy's credit card. That's their business; that's a commercial advantage over their competitors.

To get back to the question, why not do it on z/OS? Sometimes it's cost, sometimes it's speed to market. Of course, there are big commercial customers who say that Linux isn't ready yet for prime-time, bet-your-business commercial applications. A bank is not going to put its customer database on Linux – yet. It's getting there. But they'll start putting

peripheral web-serving, file-serving activities on it, and they're going to start looking at applications they can port, and looking at server consolidation.

Of course, most of the big commercial customers have a mixture of their own programming and packages. They have a massive investment in their own bespoke programming. They wrote their own applications over the years – perhaps 20 or 30 years – mostly in COBOL, but increasingly



INTERVIEW | WARWICK UNIVERSITY/IBM



now in C and Java. If you've got a big commercial COBOL application, that looks like a mountain to climb to get it onto Linux.

LXP: Why do you think Linux is not ready yet? Is it a reliability issue?

DN: The message we want to encourage is that it is ready. Our customers are not ready yet to start betting the business on it. Our perception is that Linux is certainly more reliable than Windows. And arguably as reliable as many of the mainstream Unix systems. It wouldn't demonstrate the same continuous availability as z/OS, it is true.

With z/OS, we've not only eliminated failure as much as we can, we've also tried to eliminate the reasons for scheduled outage. With z/OS, you can maintain it, tune it, monitor it and make changes to it non-stop. You don't restart it. Similarly with the hardware – these machines are upgradeable on the fly – they don't come down for an upgrade. 90% of a customer's down-time is scheduled down-time – whether it's upgrading the hardware,

don't get hacked, don't get viruses.

Another issue is SMP scalability. The industry view tends to be that a single instance of Linux is fine on a two-way or four-way, but that eight-way begins to become problematic. The mainframe gets around that problem. First, we run virtual servers – you can have 50 Linux images on one machine with one processor; or you can have a 16-way mainframe running hundreds of copies of Linux. We're stepping around that scalability issue with virtualisation.

LXP: Would IBM be prepared to invest in such a security audit of Linux?

DN: I can't say. Well, there are some markets you can't sell to without these formal statements of security – mainly military markets – but that's a specialist area. Or if you don't have the statements, you have to go to incredible lengths of physical separation. So, I don't know. It's an interesting question.

Some people feel that Linux is inherently insecure because it's open source, that because you and I can look at the source, we can somehow find a way through it. We have successfully argued that because it is open source, it is manifestly more secure. Access to being able to read the source, doesn't give you access to the operating system. And because there is so much peer review and because there are so many clever guys around the world constantly poring through that source, the security is dramatically higher.

There may come a time when IBM, in order to win business in particular areas, might find it necessary to go forward with that – but, then, we don't own the source, so I'm not sure how that would work.

LXP: What are IBM's future plans for Linux?

DN: We've invested \$1 billion in bringing Linux to all our servers and encouraging application vendors to bring best-of-breed packages to Linux. We've also contributed a lot of new code to the Linux kernel – filesystems and so on, sometimes taking code from our traditional operating systems. Interestingly enough, although we've made that investment of over

“Linux is the next big disruptive technology, a game-changer. Like the PC, like the Internet...”

upgrading the OS, or reorganising the database – it's not failure. We've tried to engineer all that out in z/OS. Linux can't do that yet. If you make a big hardware change, you're going to have to reboot. With virtual servers, of course, you can bring up a new server and transfer the workload.

So, it's not reliability. It's more consistent management, continuous operation.

Some customers would also be concerned about auditable security. z/OS has a formal, military definition of its security – certain expensive, formal tests have been gone through to prove and classify its security level. Military level security is strong enough for banks to bet their business on. Mainframes

a \$1 billion, our VP of Servers at Linux World back in January said that most of that investment has already been recouped. So it turns out not only to be good vision, it turns out to be good business too.

We're going to make more investment and we expect to recoup that, too. Going forward we have a very strong plan to extend, develop and encourage Linux.

Our strategy for Linux is to be the provider of choice to the large commercial enterprises who want Linux solutions. If a customer wants to run a commercial Linux application, we hope that customer will turn to IBM. All our servers run it, we've brought a lot of our middleware, our databases, our transaction-processing environments, and we've also built up the skills and the knowledge around the world of how to do this.

We're not doing this because it's a poke in the eye for Bill Gates. The way I put this when talking to customers is that we think Linux is the next big disruptive technology. Customers don't like the word disruptive, but what I mean is that it's a game-changer. Like the PC, like the Internet, like Java. Now, IBM had the PC and lost it. Java, we had nothing to with in its early days, it's a Sun creation. The Internet, we were late to understand its importance. With Linux we think we see the opportunity to be in the lead with what we believe is the next major disruptive technology in the industry.

It's IBM strategy going forward to offer Linux on all our servers from the smallest PC to our biggest supercomputer – in addition to the traditional IBM native operating system, both at the same time. The reason for this is that we don't know where this market is going to go, but we are giving the customer the option to run Linux.

Five years from now, if Linux goes the way I think it will, then great – everyone's well-positioned. If it doesn't fly – and who knows – well that's okay too because we haven't cut anyone off from the traditional applications and operating systems.

We let them run both by using various Hypervisor or partitioning techniques across the whole family – the mainframes run the



Dr Stephen Jarvis puts the S/390's ability to simulate a vast network of machines to good use in his research, developing scheduling algorithms for supercomputing Grids.

VM system that provides 10s, 100s or even 1000s of virtual servers. (The benchmark record is 97,000, but that's a benchmark though, and meaningless for real workloads) Our mid-range machines – the pSeries (which is the RS/6000 technology) – our classic IBM Unix boxes are now also partitionable and can be split into multiple system images running either our version of Unix, called AIX, or Linux. We have hardware partitioning on our Intel servers, and we've very recently announced an arrangement with VMware, so we're starting to bundle VMware with some of our Intel servers to provide that partitioning to run Windows and Linux alongside. That's our dual-track policy going forward. ■

Acknowledgements

LinuxPro would like to thank Doug Nielson of IBM, and Dr Roger Packwood and Dr Stephen Jarvis, both of Warwick University, for taking the time to show us around the computing facility at Warwick and answering our questions.

LinuxPro interview

GEORG GREVE

Richard Smedley caught up with the President of the Free Software Foundation Europe as he addressed the Linux Seminar in Sheffield.

As Georg Greve had just given an address outlining the benefits of freedom (in software) to businesses, it was the ideal time to quiz him further upon the ideas and business links of the Free Software Foundation Europe (FSF-E).

LINUXPRO: Let's start with the founding of the European Free Software Foundation – and why it's so necessary to have a separate organisation for Europe?

GEORG GREVE: Well, it's legally independent, but not separate. We're sister organisations and we're working together very closely. I did not think about the reasons why we needed the FSF-Europe until I started thinking about formalising it, because it was just plain obvious to me that we did. Europe is not only the region of the world where Free Software is strongest, it's also a region that's very vibrant with different cultures and sometimes the United States is just a little bit too far away.

People – journalists, politicians – would normally prefer to talk to someone who is “in the same culture zone.” You don't want to have to worry about whether this person is asleep or not when you pick up the ‘phone.

Having someone who could actually be there was a good idea. We have seen that the FSF in North America is not large enough to cover all the questions that arise globally anymore – because Free Software has started out as a small part of a an insignificant market and now it's a very significant part of a large market. So the FSF needed to think about what to do to still be able to do its job.

What we've done is to start setting up FSFs that would act as a network, instead of just a single organisation trying to do everything in the world and have a hard time doing it.

LXP: The FSF-Europe is very much centred on having local chapters and affiliated groups. Despite the single market/EU, every country approaches various areas of legislation separately; and culture, and the education system.


GG: People are still very different in Europe and I do not think that's a bad thing, but it's something of course you have to think about when you set up such an organisation. The vision was to have a single European association, unfortunately there are no European laws for associations, so we needed to come up with some sort of legal hack. This legal hack is our chapter-structure. We have one association that all of us are members in, the so-called Hub, which – by happenstance – turned out to be located in Germany. This is where we take the decisions together, where we do most of the work.

Attached to this, in terms of formalising legal presence in other countries – and being charitable in those, too – we have the chapters in the local countries.

LXP: What are the key differences from the US in the threats to Free Software?

GG: Europe is widely different in that it has a totally different copyright tradition and copyright is, of course, extremely important for software. So we have the *droit d'auteur*

GEORG GREVE | INTERVIEW

A portrait of Georg Greve, a young man with short brown hair, smiling and looking towards the camera. He is wearing a dark blue jacket over a maroon button-down shirt. The background is a textured blue wall. The image is framed by a dark blue header and a white border with crop marks.

“With Free Software you have the advantage of being independent and determining for yourself what you want.”

« tradition – the right of the author – as opposed to the right-of-the-industry tradition which the UK and the US employ.

LXP: You find the UK closer to the US ...

GG: Yes. In terms of copyright issues it is. Europe has a lot of countries that follow the *droit d'auteur* position, so it is important to look at the different traditions, to make sure that Free Software works with the *droit d'auteur* tradition as well as possible – it works

“The GPL, written with the Berne convention in mind, does a good job of applying to every country.”

already but we always have to ask ourselves “Is there anything we can do to improve the situation? Does anything require change?”

Some European countries are beginning to introduce laws on information – it's important to ensure these laws represent Free Software.

LXP: They're coming under pressure to do the equivalent of the DMCA?

GG: That is one of the most severe threats, right now. The DMCA is a really big problem. The European countries have the European Copyright Directive that is essentially the same thing as the DMCA. So it's not illegal anymore to do something forbidden – it's illegal to *think about* something forbidden. And that is just incredibly harmful. You know it's really amazing that no-one said anything about this before. It seems that the general population is not aware of the importance of copyright issues, nowadays. So it's a field where the big digital right management organisations could completely do whatever they wanted. They were free to push through whatever they felt they could push through. And politicians were apparently easily swayed. So it's important that we raise our voice now and make sure that this does not go any further. That we do not have DMCA-like legislation in Europe.

LXP: So how successful has FSF-E been at getting changes to proposed legislation? You mentioned a German law on donations and giving away...

GG: I would actually say that we've been fairly successful. Together with the IFROSS [Institut für Rechtsfragen der Freien und Open Source Software], which is a German institute for legal issues of Free Software, we've been able to get wind of this upcoming legislation; it's very hard for the average citizen to know what's going on and what's going to happen – that alone is not an easy task. They informed us that something was coming up and created a proposal that was reviewed by us and then filed. They know the intricacies of the German copyright system in terms of legislation and were able to push it through unmodified. It has been adopted in that form on January 25th, which of course is a big win for Free Software.

LXP: We're seeing here stuff that coders cannot help with – you need copyright lawyers. Has it been a problem getting people to give time – or funding them?

GG: Well, right now we do have some donors who realise how important this work is. We do not have as many as we would like to, of course, since the FSF Europe is still rather young. A lot of coders and community people have not yet realised just how important these issues are. The freedom to program might actually be taken away from them one day, on a legal level that they could not easily circumvent. I would say the gravest threats to Free Software currently arise in legal issues and areas, so the Free Software Foundation has been very active in those. Of course we're trying to bring awareness about these issues to people. We've been successful, but not as successful as we would have liked to be. However we have existed for less than a year now, and this process takes time.

In terms of companies – naturally companies are the ones who should support us more. Because our work not only enlarges their markets, it also makes sure that their business models remain safe. And for companies who only use Free Software, our work helps securing that it is here to stay.



The big companies in Germany that work in and around Free Software, like SuSE, they were not aware of the upcoming copyright law revision I told you about! They didn't even seem to notice. In a way the work of the IFROSS and the FSF-Europe saved their hide. You would think – or you would wish – that companies would understand that what we do is very important to their sustained business in the end, and that helping the FSF-Europe is an investment in their own future.

LXP: You've been in London recently for a consultation with our government?

GG: The Commission on Intellectual Property Rights, where I was invited as an expert witness: I was one out of ten or twelve people invited to give input on copyright law and recommendations for developing countries.

LXP: That's interesting – Britain has been way behind in understanding *anything* about Free Software. How are you getting on in other countries? Are you doing much to promote Free Software – or too busy fighting legislation, and governments are realising on their own?

GG: Very often these things happen behind closed doors. Politicians will call you and ask you some questions, or invite you to some meeting to talk. This is what's been happening.

I'm actually pretty sure that you'll find that in the majority of pro-Free Software legislation in Europe we were somehow in touch with the process. Someone in the FSF Europe will have participated at some point, and very often it's not even newsworthy.

We're always willing to talk to people – even if they came from Microsoft. We'll explain Free Software to them, and try to convince them that what they're doing is wrong, mistaken, and stupid in the long term. Of course if we did that we would not be endorsing them – we do not want to endorse companies that are not there yet. If we endorse anybody, it's people that have gone quite a way towards Free Software.

LXP: You're very much working behind the scenes with your lobbying...



INTERVIEW | GEORG GREVE



« **GG:** ...a lot, yeah. But the FSF (America and Europe) have *always* been extremely active behind the scenes, talking to people and building understanding, support and awareness. A lot of the things that have been happening have been influenced by the FSF, without too many people being involved in it.

LXP: Why do you think the UK is so far behind on waking up to Free Software?

GG: That is a good question. I'm probably not in a good position to *judge* the UK system in

any way. But the problem with the UK seems to be that sometimes they appear to be quite eager to listen to what the United States tell them – but the USA is behind in terms of Free Software, so essentially the UK is listening to someone who really doesn't know what he's talking about – which is a bad idea.

LXP: I believe GPL is being re-examined to try and accomodate European law?

GG: The GPL, since it was written having the Berne convention in mind, does a very, very good job at applying to every country on this planet. But some parts of it may be improved further. For instance, in Germany there is customer protection law that does not allow to limit liability completely; you cannot exclude liability 100%. So you do have limited liability due to the GPL, but you could have even less if you phrase it slightly differently for the German system. Of course, these are issues that are extremely difficult to handle, since the GPL is extremely important. Probably more than 50% of Free Software is under the GPL. If we made a mistake with the GPL it would be pretty harmful. Therefore we're always moving with extreme care and caution whenever the GPL is being revisited. Whatever we plan to do with the GPL will always be reviewed by a lot of people who have a long standing in such issues. Also we give it some time to be reviewed publicly before we really change anything. It's just too important to act in haste.

LXP: Any other areas of the FSF-Europe's work that you'd like to mention?

GG: There are some projects that are coming up that are going to be rather interesting. There will be an AGNULA project – "A GNU/Linux Audio Distribution". Also we'll take care of securing the "legal maintainability" of the project to ensure the lastingness of this project for the Free Software community. We've been asked by Centro Tempo Reale – the main project participant – to join, so that we can move it in the right direction.

Although we haven't even existed for a year yet, we already have our first official European Union project, which I think is actually not a bad sign [grins]. Things are going quite well.

We've been at many trade shows, talking to many people, and we see an increasing number of people wanting to work with us. There are still people who don't know that we exist, but the situation is improving.

We also see initiatives to spread the FSF-Europe, and the concept of Free Software in other countries in a more formalised way, too. The FSF-Europe has been seminal in setting up other Free Software associations – for instance ANSOL in Portugal, FFS in Austria, or AFSF in the UK – the FSF-Europe has helped those come along, by encouraging the local communities to build something like them.

LXP: Long overdue...

GG: It seems like the potential was there all along, but it just needed an initial spark.

LXP: People get sidetracked with "Linux" – running it "because it works," running proprietary software on it – there wasn't the energy and focus on Free Software...

GG: That is the big problem that I've been seeing with Linux User Groups, too: losing focus. The issues are way above and beyond the Linux kernel, there's much more to it. There will be many more Free Software operating systems; there will be more kernels that are equally important in the end.

LXP: Certainly, the Hurd is looking good.

GG: The Hurd is *amazing*. Very many people have claimed that the architecture wouldn't work, but it does [smiles], it's working. It just does not now have the hardware support that you would like it to have. When those flaws are worked out – once they get to do these things, the Hurd will be really interesting.

LXP: It's going to go on by leaps and bounds; as they get a critical mass of developers it will keep growing..

GG: Yeah. I think the Hurd is going to be a very interesting kernel to use with the GNU system. I'm looking forward to having that running and to see it really take off.

LXP: Tell me about the "We speak about Free Software" campaign.

GG: Many of the Free Software companies have had a lot of trouble with the term "Open Source" because their market would be watered down by companies that would not do Free Software – they'd just release some of the source code and then claim to be "Open Source." There was no distinguishing term any more. They could not really tell their customers why they were different from those people. Whereas they are in fact very much different.

So we started the "We speak about Free Software" campaign, where we gave some of the major arguments for business people why they should actually prefer the term Free Software over Open Source, and make sure to speak of Free Software since it will give them additional value. We had several companies signing up for that right away. A few days after launching, funnily enough, we got an email from Bruce Perens, asking us to put him on there as well. Bruce Perens was one of the seminal people behind Open Source. – so that was an interesting thing to see. We even made an exception for him [as an individual, rather than a business] and put Bruce Perens on the list.

“Very often these things happen behind closed doors. Politicians call and ask you some questions.”

LXP: I think he feels that "Open Source" seemed like a good idea at the time but it didn't work out that way.

GG: The problem is that it has taken off in a totally different direction from what was planned. Open Source was meant to be a marketing campaign by the people who did it. We've viewed it critically right from the beginning because we felt that the movement would be diluted that way, but it was created out of the wish to push Free Software forward.



INTERVIEW | GEORG GREVE



« **LXP:** I don't think that anyone expected Microsoft to market Windows CE as "Open Source"....

GG: That is sort of scary. The problem is that since the Open Source movement failed to explain the importance of freedom to people, they do not see the difference between what Microsoft does and what the Open Source movement – in terms of Open Source Definition – does. It is not a clear term anymore, so I think that it's time to stop thinking about that and to start using [the term] Free Software again.

LXP: Small tech (particularly Linux) companies have been quick to see some of the advantages of Free Software. IBM and others who make their money selling boxes can see the advantages of the scalability of Free Software. In the middle are the vast majority of businesses, who by tradition have embraced ideas of proprietary software and so-called "intellectual property" – for them, what advantages has Free Software got?

GG: Well, regular businesses are usually software customers, which means that software is not their main point of interest: it's just something that should work so that they can do their job. For them software is not a value by itself, as soon as it's not maintained any more, it quickly becomes utterly worthless. In Free Software there is value because it's ultimately cost-efficient and ultimately future-proof. It allows them to use their IT department as long as they would like to use it. To run whatever software they want to run on whatever hardware they want to use. They are not depending on software companies pushing new releases to earn more money, which is essentially what the proprietary software industry have been doing. They have been pushing new release without really any sense behind it: customers were forced to go along with it because the old version would not be supported any more.

LXP: When it comes to try and market these ideas to companies, what's hardest about putting these ideas across?

GG: They have been told, for years and years, that software was some sort of “thing” that one could possess. And they’ve been told that when they buy a licence that they would possess the software – which is totally misleading as they do not: they have no rights over the software; proprietary software gives you no rights at all. You are essentially chained to whoever created the software, which means your business success, your whole business model, whether you go out of or stay in business – everything is related to what your proprietary software vendor does.

Most don’t even realise this dependency until their proprietary software package goes out of distribution, at which point of course it’s too late; because then they have to scrap their whole IT setup and find a new solution.

With Free Software they can go somewhere and ask for certain features. They can say “I want my IT thing to do this.” And their vendor who (hopefully) knows what he’s doing will inform them about existing solutions that may be modified for the current problem. This investigation costs time, so does the modification of the software. That is why the migration costs are probably the same – but in the middle to long term you make sure that your business success is sustainable. You make sure that you do not depend upon a proprietary software vendor. Freedom is incredibly important for companies – most of them do not realise that.

LXP: German companies have a tradition of being able to think in the medium and the long term; in Britain and America it just doesn’t happen. It’s another barrier?

GG: Possibly. Of course I cannot tell how far UK companies are planning ahead.

But with Free Software you do have the

“The problem is that the Open Source movement failed to explain the importance of freedom.”

major advantage of being independent and determining for yourself what you want. I mean we do have a weird one-size-fits-all paradigm for software; whereas every company needs a slightly different solution. So what do companies like the consulting companies do? They modify your business to fit the software better! [laughs] instead of the other way round. Which of course makes sure that every company works in the same inefficient way – loses money the same inefficient way – and they all have the same weaknesses.

LXP: Put like that it is madness, but one trouble with business culture is that at management level everybody’s looking over their shoulder and no-one wants to do anything different, or be the first.

GG: Yes, but think about it: what is the capital of a company? Normally it’s the idea behind a company, and what makes a company truly successful is being able actually to implement that idea.

OK the idea may be bad or good – I don’t know – but with Free Software you actually have a *chance* to try your potentially good idea instead of being forced to use a known bad idea. That is the big difference.

LXP: Georg, thank you very much. ■

Local freedom Campaigning for your rights to your software

The FSF-Europe can be found at <http://www.fsf-europe.org> along with links to local chapters. The newly founded Association For

Free Software (AFFS) in the UK is at <http://www.affs.org.uk> Other local organisations are linked from <http://www.fsf.org/links/links>.

<http://www.fsf.org/philosophy/free-sw.html> More information on Free Software can be found at <http://www.fsf.org/philosophy/free-sw.html>



We are not immune

Nick Veitch examines the myths and realities of the virus threat to Linux.

Since the issue of Linux viruses was raised in Linux Format magazine over a year ago, a handful of new viruses have been discovered that specifically target the Linux OS, and some which extend themselves to cover other flavours of Unix as well. Compared to the number of new viruses detected on the Windows platforms in the same period, this is a pretty insignificant number, but the threat does still remain. If you aren't taking the threat

of Linux viruses seriously, you may lose more than your ignorance.

The real cost of viruses

When people consider the consequence of a virus infection, they usually think of the unpleasantness of corrupted data and lost work. These are real and unpleasant dangers, but for IT professionals, there are more wideranging ramifications to a virus attack.

Containment may be the first issue. If one

system in your organisation gets the virus, is everything else going to catch it too? If a virus manifests itself behind your firewall do you have any defence against it?

In a well run IT department, disinfecting one machine may not be such a chore, but disinfecting ALL the machines is a different matter – do you really want to spend several days going from desktop to desktop?

The impact on resources is often underestimated too. even half a dozen machines trying to portscan everything they can find on the network is going to swamp your LAN with packets. As victims of the Melissa virus for Outlook found, the biggest problem was often that their mail gateway was swamped with mail packets being sent out – the virus itself didn't damage files, but it still brought companies to their knees as their email gateways (often hosting other important services) had to be taken offline.

And of course, there is always the embarrassment factor. Any of your customers are probably going to remember you less for your customer service, and more for the fact you sent them a virus which blitzed their system too.

Virus myths

Due to the relatively low numbers of viruses discovered on the Linux (or any other Unix based OSs), there are a number of myths about the invulnerability of the platform to virus attack. These range from the ill-considered to the downright dangerous.

Myth 1. If I'm not running as root, I won't come to any harm.

There is a small element of truth here. Because everything on a Linux system works on file permissions and privileges, a virus contracted by an ordinary user, or an ordinary process for that matter, is going to have limited scope for causing havoc, if it is of the classic executable manipulating type.

The virus itself won't have write permissions for any of the standard system commands. Only users who have executables stored in their own directories will be effected. Hopefully the problem will contain itself.

However, this theory neglects the fact that

processes themselves may have to run as root for certain tasks. On a well secured system with chroot jails, this shouldn't cause too much of a problem. The other caveat of course is that permissions are rigorously policed. A combined attack, perhaps using an exploit to

“The impact of viruses on system resources is often wildly underestimated too”

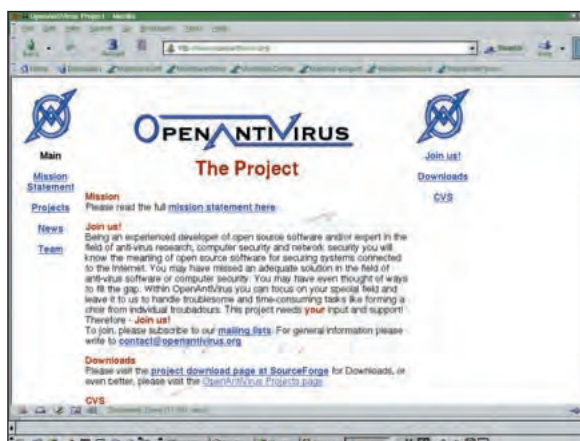
‘root’ the box, could also let a virus in. In fact, if you take a look at the virus profiles at the end, you'll see that this is a popular method of attack.

Networked storage can also be a problem here. Many users having access to the same filesystem is like everyone using the same toothbrush. Again, sensible security policy can often be helpful – is there any need for the shared filesystem to have exec permissions for example?

Myth 2. Linux is so diverse, no virus would ever work reliably enough to cause a problem.

Again, there is some element of truth here, but not a lot. While it's

The OpenAntivirus group are responsible for a number of different projects.





true that the most successful Windows viruses tend to attack particular files and weak points on that system, this is just a refinement of their behaviour. A virus which infects ELF executables for example, doesn't particularly care where it finds them. If it directory walks your disks it doesn't matter whether all your executables are in /bin,/usr/bin or anywhere

“The diversity of the OS protects against exploits to an extent, but not against brute force”

else. The diverse nature of the OS protects to a certain extent against exploits, but not against brute force.

Myth 3. There aren't any real viruses for Linux.

The current Linux viruses may not be in the same league in terms of destructive power or native cunning as, for example CODE RED, Nimda, SirCam or any number of other Windows viruses, but so far that's possibly more to do with the nature of the virus writers. So far, most Linux viruses have been content with benignly infecting files and passing on,

almost as an exercise. But the capability for destruction is there.

Myth 4. Linux is immune from email viruses.

Perhaps in the user space this is currently true (but only because nobody has implemented a virus which does it, not necessarily because it's impossible). However, if 99% of your users are running Windows desktops and accessing a Linux mailserver, then you are vulnerable to email viruses. Most of the damage, as we have said before, is done by stealing resources from the server.

Virus solutions

The more cynical among us may suspect that a lot of virus hype is stirred up by the people who make anti-virus software. This may or may not be true, but in any case, some form of virus protection is better than none. But what anti-virus software is available for Linux, that will run on a native Linux system?

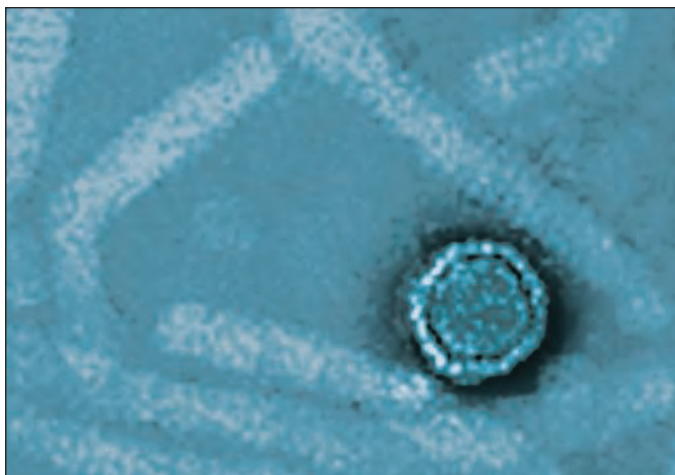
There are a few, and the bonus is that most of them can be used to scan for Windows viruses as well.

As far as we know, the first Linux native anti-virus software was created by Kaspersky (www.kaspersky.com). There are two versions of the software - one for personal use, and one for server use, which also integrates with sendmail, postfix and qmail to scan for Windows viruses transmitted as attachments.

Unusually in this field, the Kaspersky software is also supplied as open source code, which means that you can easily integrate the client into other applications (e.g. filtering file uploads to a server, or integrating with other mail clients).

Sophos, the UK based Antivirus specialists (www.sophos.com), also produce a version for Linux and other Unix systems to complement their Windows products. It performs the usual types of virus scanning, and also detects PC viruses. It can also run on-access scanning of networked Windows desktops. There are no specific modules for mail scanning, but uses an interface to allow third-party software to integrate with the system.

There are a number of Open Source anti virus projects also worth investigating. Possibly



Linux virus who's who

A list of some of the current known Linux viruses

Name	Risk	Type	Date discovered	Notes
Alfa	Low	file infector	15/3/2002	Not encountered in the wild. Doesn't spread recursively
Fork	Medium	trojan	20/7/2000	Trojan attempts to send network packets to other hosts, and ties up resources
Lion	Medium	worm	23/3/2001	Worm originating from China. Uses known vulnerability in BIND and portscans targets.
Ramen	Medium	worm	17/1/2001	Worm which targets web servers, uses known exploits.
Rootkit	Low	trojan	4/4/2001	Uses wu-ftp exploit to gain root access and replace system files
Secldpd	Low	trojan	14/3/2002	Uses LPRng exploit on x86. Not found in wild
Adore	Low	worm	9/4/2001	Uses binaries and scripts to search for vulnerable services (Bind, lpd, ftp)
Amdcrash	Low	trojan	28/2/2002	Amd buffer overflow exploit. Not found in the wild.
statdx	Low	trojan	14/3/2002	Not found in wild. Exploit which targets rpc.statd on Rad Hat
Woot	Low	trojan	28/2/2002	Not found in wild. Portscanning exploit detector.
Lindose	Low	file infector	27/3/2001	Not found in wild. Infects Windows/Linux files accessed by Linux system.
Rcpmountd	Low	trojan	28/2/2002	Not found in wild. Red Hat specific rpc exploit.
Snoopy.b	Low	companion	22/12/2000	Renames binaries and scripts and copies itself as original.
Cheese	Low	worm	22/5/2001	Attacks systems already infected by Lion worm.
Bliss	Low	file infector	29/9/19967	Possibly the oldest Linux virus? Must be run as root.

the most active of which at the moment is the collection of software maintained by the OpenAntivirus group (www.openantivirus.org). Most of the client software runs in Java for deployment on the widest range of platforms, and the current line-up includes file scanners, on-access scanning and mail scanners.

If you are considering virus protection for an email gateway, some commercial products have their own solutions. GL Mail from Gordano (www.gordano.com) for example includes the option of an anti-virus plugin which will scan for those troublesome attachments.

Of course, anti-virus software is never a complete solution. For the virus to be detected it has to be known, and to be known, it has to have already infected somebody – which could be you. A good security policy should help minimise the risks:

- Keep up to date with lists of exploits and update regularly.
- Control root access and police permissions.



- Construct chroot jails for vulnerable services.
- For critical systems, consider filesystem monitoring software such as Tripwire, which can detect modified files. ■

Kaspersky and others also host virus databases, which will keep you up to date.