

# news@UK

*The Newsletter of the UK UNIX Users Group*

Published electronically at <http://www.ukuug.org/newsletter/>

---

Volume 11, Number 3

ISSN 0965-9412

July 2002

---

## Contents

<b>Notes from the Secretariat</b>	<b>3</b>
<b>UKUUG Ltd – Annual General Meeting</b>	<b>3</b>
<b>Knoppix CD</b>	<b>4</b>
<b>Help wanted</b>	<b>4</b>
<b>UKUUG Open Source Award 2003</b>	<b>4</b>
<b>Call for participation - LISA/WINTER CONFERENCE - February 2003</b>	<b>5</b>
<b>UKUUG Linux Developers' Conference 2003 - Call for Papers</b>	<b>6</b>
<b>Report on UKUUG Linux 2002 Conference, 4th-7th July</b>	<b>6</b>
<b>Reports from Linux Developers' Conference 2002 (2)</b>	<b>8</b>
<b>Going SANE–2002</b>	<b>9</b>
<b>The Gimp</b>	<b>10</b>
<b>BSD Update</b>	<b>14</b>
<b>A Linux User Defects</b>	<b>15</b>
<b>Book review: HTML Pocket Reference, 2nd Edition</b>	<b>16</b>
<b>Book review: System Performance Tuning, 2nd Edition</b>	<b>17</b>
<b>Book review: XSLT</b>	<b>18</b>
<b>News from O'Reilly</b>	<b>20</b>
<b>Shell Competition Results - Prime Numbers</b>	<b>21</b>
<b>Next issue of the Newsletter</b>	<b>21</b>
<b>Contacts</b>	<b>22</b>



## Notes from the Secretariat

**Jane Morrison**

For those of you who didn't attend the recent Linux 2002 Developers' Conference in Bristol, I can confirm it was a great success. With over 200 delegates it was the largest event UKUUG has organised for some years.

The conference CD was given out at the event to all attendees and we have also posted the CDs to those members who did not attend. If you haven't received the CD please let me know.

Just out of interest I have looked back in the files and the first Linux conference we organised on 27th and 28th June 1998 in Manchester, attracted some 50 delegates. We had just 7 speakers (Richard Moore, Stephen Tweedie, Gareth Bult, Ian Jackson, Jeremy Chatfield, Alan Cox, Donal Fellows) compared to the 30 speakers this year.

Linux 2003 will be held in Edinburgh (dates and venue to be confirmed) and the Call for Papers appears in this issue Newsletter.

The next event in the UKUUG diary is the Annual General Meeting, to be held on Thursday 26th September 2002 at the Institute of Education, London at 6.00 pm. Members will automatically receive full details about the AGM during August.

The Call for Papers for the Winter conference in 2003 (probably February) also appears in this Newsletter. More details will be circulated to members once we can confirm the dates and venue.

A reminder that Newsletter submissions from members' are very welcome. If you have something you would like to appear in the future issues, you are encouraged to submit it to [newsletter@ukuug.org](mailto:newsletter@ukuug.org)

---

## UKUUG Ltd – Annual General Meeting

**Thursday 26th September 2002 at 6 pm**

Institute of Education  
20 Bedford Way  
London  
WC1H 0AL

The Agenda and paperwork will automatically be sent to all members in mid-August.

The AGM will be followed at 7.00 p.m. by a LUUG talk (speaker to be announced).

UKUUG Secretariat

PO Box 37

Buntingford

Herts SG9 9UQ

Tel: 01763 273475

Fax: 01763 273255

[office@ukuug.org](mailto:office@ukuug.org)

<http://www.ukuug.org>

## Knoppix CD

All members, whether they attended the Bristol Conference or not, should now have received the Conference CD and also a copy of the Knoppix CD. If you have not received these, please contact the secretariat.

Knoppix is a “live-filesystem” bootable Linux CD which is available from <http://www.knopper.net/knoppix/index-en.html>

To quote from the site: “KNOPPIX is a bootable CD with a collection of GNU/Linux software, automatic hardware detection, and support for many graphics cards, sound cards, SCSI devices, and other peripherals. KNOPPIX can be used as a Linux demo, educational CD, rescue system, or adapted and used as a platform for commercial software product demos. It is not necessary to install anything on a hard disk. Due to on-the-fly decompression, the CD can have up to 2 GB of executable software installed on it.”

The English version was customised for the CD distributed at the conference by Alasdair Ker-gon.

---

## Help wanted

**Charles Curran**

As you will see above, our AGM is on Thursday, 26 September. As ever, we are trying to get members to participate in UKUUG work, particularly in working- and focus-groups. Much of UKUUG’s work is currently handled by Council or individuals thereon but we, UKUUG, could be far more productive if we were to work together (in groups). If you are interested in co-operating thus, in, for example, work on Events, Newsletters, Membership, Commercial Liaison, Academic Liaison, Schools, Web-site and Documentation, Reginal Groups, Publicity, or any other item, please contact Jane Morrison at the UKUUG office.

---

## UKUUG Open Source Award 2003

**Charles Curran**

UKUUG has established an Open Source Award which is open to current students in UK Higher Education. The prize of £ 500 is to be awarded annually (provided submissions of sufficient merit are received) for a significant contribution to open source; this might be in the form of an article or paper, software product, or other contribution. The winner will be expected to deliver a talk at the annual UKUUG Linux conference, which for 2003 will be held in Edinburgh in the summer (the date will be finalised during August 2002).

The judging panel will include representatives from UKUUG, the Open Source community, and UK Computing Science departments.

The closing date for submissions is Friday, 4 April 2003. E-mail any enquiries to [office@ukuug.org](mailto:office@ukuug.org). Unfortunately, the award for this last year did not take off. This time, notice should be adequate so that students get to hear of it early, although we should be grateful if you would spread the news.

---

## Call for participation - LISA/WINTER CONFERENCE - February 2003

UKUUG will hold its next Winter Conference in February 2003. The UKUUG Winter Conference is historically an event where not only prominent topics are discussed within the Conference presentations but also where members and friends meet, learn, and enjoy lively debate on a host of subjects.

As always, the UKUUG wishes to encourage work-in-progress presentations, and student project posters; proposals for these should be submitted in the same way as for full papers.

The event will take the form of a series of presentations, each followed by a discussion on issues raised. To this end, papers are invited from interested parties on the general themes, and related topics.

The UKUUG wishes to encourage discussion on all aspects of systems and their administration and is especially interested in papers covering theory and practice, high-availability, performance, network management, novel solutions to practical problems, integration, interoperability, and security (including the business, legal and moral issues).

You do not have to be a member of UKUUG to submit a paper. Submissions from speakers from outside of the UK are welcome.

Last year's Winter Conference included talks by:

Paul Anderson on Large-scale Linux Configuration with LCFG

Jim Davies on A Web-based Administration System

Andrew Findlay on LDAP and Security

Matt Holgate on The Arusha Project

Stuart McRobert on Applied Ethernet 10GB and an update on Sun SITE

Wayne Pascoe on A business case for FreeBSD

Peter Polkinghorne on Containing Windows

Alain Williams on Configuration: Making it Easy, Getting it Right

Mike Wyer on Lexis Exam Invigilation System

Simon Cozens also gave a tutorial on Perl

### Significant dates

Initial abstracts submitted: 26 September 2002

Closing date for abstracts: 11 November 2002

Authors notified: 18 November 2002

Programme published: end November 2002

Final papers due: 13 January 2003

### Method of Submission

Potential authors may request further information by sending e-mail to [winterconf@ukuug.org](mailto:winterconf@ukuug.org)

Initial abstracts should be sent as e-mail to the same address

Abstracts should be accompanied by a short biography, and, ideally, should be about 250-500 words long. Final papers should normally last 30-40 minutes. If you need more time for your presentation, please tell us when you submit your abstract.

Submissions are welcome from members and non-members of UKUUG and particularly submissions by students. Student sponsorship may be available on a discretionary basis.

## UKUUG Linux Developers' Conference 2003 - Call for Papers

### EDINBURGH 2003

Planning for the 2003 Linux Developers' Conference is already underway and again we are seeking ideas, speakers and sponsors.

We would like to invite speakers on all types of Linux development to contribute. The programme will cover a variety of subjects, including kernel and desktop development, tools, applications, and networking. Any topic likely to be of interest to Linux developers and enthusiasts will be considered.

The topics presented in recent years have included: ARM Linux, Clustering, CORBA, Debian Package Management, Enterprise Filesystems, Exim, GNOME, I20, Mail Servers, Memory Management, Performance Programming, Samba, Security, SMP, and VMware.

Further details of earlier conferences can be seen at  
<http://www.linux.ukuug.org/linux99/>  
<http://www.ukuug.org/events/linux2000/>  
<http://www.ukuug.org/events/linux2001/>

and of course at  
<http://www.ukuug.org/events/linux2002>

Initial abstracts should be submitted to the conference organisers electronically using the form on the Linux 2003 website at  
<http://www.ukuug.org/events/linux2003/CFP/>

Abstracts should be accompanied by a short biography, and, ideally, should be about 250-500 words long. Final papers should normally last about 40 minutes, including 10 minutes for questions and answers. If you need more time for your presentation, please tell us when you submit your abstract. We shall acknowledge all submissions.

#### **Significant dates:**

Closing date for abstracts: 16th March 2003

Accepted authors notified by: 7th April 2003

Final papers due by: 18th May 2003

Particular queries should be sent either to the UKUUG office, or to the Linux 2003 mailing list [linux2003@ukuug.org](mailto:linux2003@ukuug.org)

To keep the conference fees low, we are seeking sponsors and exhibitors. For further information about sponsoring, exhibiting, or attending the event please contact the UKUUG office: [office@ukuug.org](mailto:office@ukuug.org)

Telephone: 01763 273475

Information about the event will be updated on a regular basis on the website:  
<http://www.ukuug.org/events/linux2003/>

---

## Report on UKUUG Linux 2002 Conference, 4th-7th July

*Dominic Hargreaves*

### **TUTORIAL: Thin Clients in a GNU/Linux Environment with the Linux Terminal Server Project (LTSP)**

Jim McQuillan - LTSP Founder

This in-depth tutorial guided the audience through setting up an LTSP server and configuring workstations. The event was a good demonstration of real-life problems as Jim fought with

corrupted downloads and the hardware incompatibilities of an ancient PC donated for the day by the University, and tried not to destroy his EPROM programmer with the local voltage. Overall, his demonstrations were informative and enjoyable, and I'm sure that more people will be deploying LTSP as a result.

### **Buried alive in patches: Six months picking up the pieces of the Linux 2.5 kernel**

Dave Jones - SuSE Labs

This was, for someone who doesn't work directly on the kernel, a fascinating insight into the development process. The technical and political aspects of maintaining the 2.5 -dj tree (the primary purpose of which is to provide a mostly-working 2.5 during a time of rapid development) alongside Linus' tree, were discussed, as well as the dangers of offering to maintain archaic filesystems. Dave spoke and fielded audience questions with panache, and attracted a large and appreciative audience. Oh, and he isn't going to maintain 2.6.

### **Panel Discussion: The Linux Kernel**

Marcelo Tosatti (2.4 maintainer, Brazil) and others

The panel here consisted of Christoph Hellwig, Dave Jones, Marcelo Tosatti, although Marcus Brinkmann of GNU Hurd fame also put in a few words. The discussion touched on many areas such as the effects of Palladium on Linux and other Free software, code freezes for 2.6, commercially certified kernels, and many other aspects of kernel development. It stood out for many as the highlight of the conference.

### **Ten things to do with a Dead PC in Bristol: Using Linux in Undergraduate Teaching**

Craig Duffy - Bristol UWE

The opening event of Saturday's programme, Craig's talk discussed UWE's extensive use of Linux systems as a development and research tool for computer science undergraduates. It was light-hearted and took the initial form of a straight comparison between different teaching methodologies such as traditional textbooks and an up to the minute Linux-based locally-produced material, as well as between the use of proprietary and Linux-based systems. And of course, no talk by this name could be complete without the collected wild ideas that the conference delegates put to paper to answer the question of what one should actually do with a dead computer.

### **Bristol's Nomadic Network: Building a Secure, Seamless and Scaleable Wireless LAN**

Josh Howlett - University of Bristol

This was a highly-topical talk, especially for all the network junkies crowding around the Debian stand throughout the weekend, since the conference's connectivity was part of Josh's pilot scheme for a network that seemed rather bizarre at first sight. Designed for universal access wherever you are, on or off campus, a combination of PPPOE and PPTP access methods are starting to abstract the connectivity of Bristol network users for users from students to senior academics alike. And, of course, it all runs on Linux.

### **The GNU Hurd**

Marcus Brinkmann (Germany)

Marcus gave a packed talk about the design philosophy of GNU Hurd (and micro-kernel OSes in general), its progress to date, and its future development. This talk appeared to be one of the most highly praised of the day, and it was a shame that time limitations prevented a complete exposition.

### **Attacking Linux Servers**

Nils Magnus (Germany)

This was a whistle-stop tour through the essential tools needed for probing Linux systems (though most of them could be used equally well against other systems).

### **The Easy Way to make Embedded Distributions: emdebsys**

Wookey - Aleph One Ltd

Wookey gave an overview of this emerging system for developing Linux-based embedded systems. As one of the most commercially interesting areas at the moment, he demonstrated that embedded systems development can be made extremely powerful by adding Debian's huge package repository.

---

## Reports from Linux Developers' Conference 2002 (2)

*Owen LeBlanc*

### **Linux Printing Using CUPS (workshop)**

Kurt Pfeifle

CUPS – the Common Unix Printing System – has grown out of a report from the Printer Working Group which attempted to define an Internet printing protocol to replace the existing muddle of incompatible, obsolete, and not-quite-adequate protocols and programs which have existed up to the present. The workshop presented a clear picture of how the CUPS system works and can be administered. Kurt's easygoing presentation actually covered a number of configuration utilities and files in amazing detail, touching on issues of security and flexibility. He also presented some of the future plans for CUPS, as well as developments in other related pieces of software, such as ghostscript. I found the whole workshop most valuable.

### **RAS (Reliability, Scalability, Serviceability) round up from the IBM Linux Technology Centre**

Richard Moore

This talk summarised a number of ways of investigating problems and failures, and in particular of possible changes in the kernel which would improve our ability to find out what went wrong while at the same time allowing the service to recover as soon as it could. The presentation was well thought out and clear; it included discussion of dynamic probing, kernel hooks for dynamically loading RAS tools, flexible dump utilities for post-mortem analysis and for taking snapshots, and security features.

### **Exim 4**

Philip Hazel

Why was it necessary to make fundamental changes in the way Exim works? Philip Hazel explained the need to reorganise Exim completely, and showed how the new strategy was far more flexible than the old. It wasn't possible to include many details in such a short presentation, but it gave us a good place to start if we plan to study the new version of Exim in greater detail.

### **Running Linux on AMD's x86-64 architecture**

Bo Thorsen

Last year Bo Thorsen presented a humorous account of porting the Linux kernel and development tools to this architecture; this followup was most informative and entertaining as well.

Dynamic Binary Translation Mark Probst

Mark is working on a project which does binary translation of code from one architecture on another. The translation is done in one small block at a time, which enables it to adapt dynamically by, for example, not wasting a lot of time generating condition codes which are not needed in later steps. The account was careful and intriguing.

### **Subversion: a version control system replacing CVS**

Sander Striker

This new version control system is built on existing components, such as apache and the Berkeley db library. It attempts to provide a portable tool that overcomes some of the weaknesses of

---

CVS. The alpha release should occur in July, and version 1.0 should be released before the end of the year. Although he had no slides, Sander's presentation was clear and direct. He explained a number of features that were and were not present in the first release, and suggested some of the future course of development of this package.

### **Experiences of Using PHP in Large Websites**

Aaron Crane

You might expect a PHP user and trainer like Aaron to be completely gung-ho about his chosen tool. But, while admitting that PHP was often useful for doing small, simple tasks, Aaron warned that there are a number of pitfalls in using it for long term, large, complex projects. This paper was fun and had some good content as well.

### **Grid Computing**

Ruediger Berlich

Ruediger explained how grid computing attempts to share serious computing resources across distant sites. He explained that the techniques are still in their earliest state of development, and that a number of questions must be addressed relating to security and other issues. In the long term, he proposes that a developed worldwide resource which can develop out of today's primitive grid projects will have a greater impact on IT than the world wide web has done in the past decade. This clear presentation was accurately timed to end at exactly the right point; if only all other speakers could manage this!

---

## **Going SANE-2002**

***Charles Curran***

NLUUG, the Dutch user group, ran their third, biennial SANE event: (for details, including some papers, see <http://www.sane.nl/>) at MECC (Maastricht's exhibition and conference centre) at the end of May. The aim of this event is to cover anything of interest to systems and networks administrators. NLUUG gave me a conference pass.

The twin-track conference was held on the Thursday and Friday, preceded by three days of five-track tutorials. There were 500 participants altogether, with about 250 visible at the conference; three-quarters came from the NL, six were from the UK.

Part of the event is a semi-formal exhibition in a large space that is also used for poster displays and the refreshment area. There were fewer exhibitors than in 2000, with only eight active stands. BladeLogic, with its European office at Farnham Common, was the only UK-based exhibitor.

The tutorials were split into four 90-minute chunks per day, with most lasting just a half-day. Topics covered were: IPSEC, DNSSEC, IPv6 deployment, DNS basics, FreeBSD Internals, FreeBSD f/s and Networking, Linux Kernel Internals, Samba, CUPS, Network Monitoring Tools, RRDtool, Network Security and Java, Performance Tuning... and Capacity Planning, Tuning applications, UFS and NFS CookBook Tuning, Packet Filtering and NATting, Firewalls, and the most popular, the sessions on Black Hats (i.e. intruders).

The conference opened with an interesting and amusingly delivered keynote by Bill Cheswick, now of Lumeta, on mapping the Internet. Details of Bill's work can be found at [lumeta.com](http://lumeta.com) or [www.cs.bell-labs.com/who/ches/map/](http://www.cs.bell-labs.com/who/ches/map/)

He talked about why and how he and others have been mapping the Internet.

Other conference talks were entitled or about:

---

Portable Dynamic Web-content  
A self-learning SPAM filter  
Black-box cryptography  
Distributed IDS  
Reusable System Eng. Techniques  
Forensic Issues  
CUPS  
Demilitarizing backups and restores  
Designing server software for zero down-time  
E-mail processing with Perl  
SNMP Tools  
Global distribution of free software  
Honeynet  
Honeypots and staying informed (of the latest vulnerabilities)  
ICANN Policies  
Internet Trauma 1994-2002  
Maintaining clusters using PXE and Rsync  
Nomadic Collaboration Management  
fsck in the background  
Scheduled Maintenance Windows  
Simulating Web workloads  
Sociological phenomena at Unix hacker conferences  
System admin as communication over a noisy channel  
User-friendly home control

The social evening was an enjoyable extravaganza in Maastricht's Festi-Village where we had plenty of opportunity to chat, eat, drink, and even dance to the jazz bands.

My main purpose at the event was to meet with those from other user groups and determine just how we might co-operate. This was relatively useful, although not as organised as it could have been.

The fees for the SANE tutorials and conference are not cheap but one does get an abundance of high-quality speakers. Besides, as with UKUUG's own events, attendance and participation is of benefit to most enterprises, bringing better, more efficient practices, quicker response, and, ultimately, lower costs. Yes!?

---

## The Gimp

***Ellie Baskerville***

Hi, I'm Ellie and I'm a gimpoholic. Interesting fact numero tres, I hate writing introductions. So, I'll keep this brief. Here's how I started with the Gimp. One fine summers evening Rob came home from work and said he'd downloaded some sort of graphics program with a weird name that I might like to have a look at. Now you would have thought that at my great age, I could have come up with something better that would ruin my eyesight and strain my wrists, but 4 years on I'm still hooked. Still the happy little Gimper. When you like something, you kinda want to share the joy and, I guess, this series of articles is my attempt to do just that. I thought I'd start off with a look at a bit of Gimp's background, how to get a copy and how to install it. Then, plunge head first into a tutorial that will hopefully inspire you to go forth and gimp on your own.

## Roots

Gimp started out as an extracurricular activity of two good looking boys from Berkeley - Spencer Kimball and Peter Mattis, in 1995. Legend has it they were frustrated with their lack of results from their compiler class project in LISP. Inspired by Kimball's brief encounter with Photoshop - he had caught some deviant attempting to remove Cindy Crawford's bikini using a clone tool, they decided to write an image manipulation program in C. The result quickly grew into a full-fledged image editing program, but after working on it for two years, the boys graduated and moved onto "real" jobs. People slowly began to pick up the project in their wake, gradually forming a Gimp development community, who originally focused on stability, but has now blossomed into providing new features for the program called plug-ins.

What's in it for me?

It's free. Yep, you read it right, this robust, flexible, powerful program for painting, manipulating, animating and processing images will cost you nada, nil, zilch, and if, like myself, you've an eye for a bargain, that's excuse enough. But, if you know someone that's just spent 500 quid plus on Photoshop, why not twist that knife in just a little bit further?

Gimp comes stuffed with features. Many form part of the core package, but many more can be downloaded as plug-ins. You get to chose which features you really want, and guess what? Most of them are absolutely free too!

Gimp's plug-ins offer more parameters, which allows you more choices, which in turn allows you more artistic freedom - which is always to the good. Gimp is focused on web design graphics. It's the bees knees for tools designed for creating images on the web. Photoshop, erm..., isn't.

Gimp has a smart, efficient menu system and for relatively small images, it's Speedy Gonzales, on wheels.

Once, you're up and running with Gimp, you can create your own plug-ins too, in no less than four scripting languages, Perl and three others (guess which one I use). Historically this has been very important as it's allowed almost anyone with more than a couple of brain cells to rub together the chance to contribute to the Gimp and get a feeling of ownership.

And the downside?

When they originally developed the program, Kimball and Mattis didn't know anything about pre-press (issues to do with printing out your images). As a result Gimp isn't as intuitive as Photoshop in the pre-press area. It doesn't support CMYK for example (if this means diddly to you, don't worry, all will be explained later). Also Photoshop is faster when it comes to handling large images.

OK, sold. Where do I get it?

Installing Gimp is relatively easy. Chances are you already have it on your system. Gimp is included in all the major Linux distributions. Just type in `gimp` at the command prompt and press Enter. If you've never used Gimp before, the Installation dialog box will pop up, telling you what files it's going to install. After, it's finished chugging away, you'll see a splash screen. Version 1.2.3 has a balloon, older versions, a cat in a room and the really ancient versions just show Wilbur, the Gimp mascot. I don't like Wilbur. I can't work out if he's supposed to be a cat or a dog. He's the perineum of the "cartoon-character-as-logo" world. I think they should have chosen a cow, but then, I've got a thing about cows, as you're about to discover.

If you need to install, then check your system's up to scratch before you download. Realistically, for a casual user, interested in creating web graphics then to work comfortably you're going to need

A Unix-like OS with X11 (although see below)

A 16 bit display, 32 meg of RAM and oodles of virtual memory

---

Lots of free disk space to write the swap file to  
At least 30 meg for a binary-only install, around 200 for a source install  
A floating point processor

You'll also need GTK+ to compile Gimp, because all the Gimp's GUI and functions are built on top of it. Did I mention GIMP stands for GNU Image Manipulation Program? I didn't? Oh well, now you know. That's probably why you need to make sure you have GNU Ghostscript and GNU wget installed, as well.

If you're looking to do pre-press work or edit large images then you'll want at least 64 meg of ram, at least a p-200 and maybe 500 meg of hard drive.

At the time of writing (early May 2002), the latest stable version of Gimp is v 1.2.3, the latest ever-so-wobbly developers version is 1.3.5. The developers chose a naming scheme similar to the one used for Linux kernels. Even minor numbers mean relatively stable versions, while odd numbers denote the "live a little" development versions. Both are released under the General Public Licence of the Free Software Foundation (GPL) and as such are open source.

Go to <http://www.gimp.org> or <ftp://ftp.gimp.org/pub/gimp/v1.2.3> and whilst you're at it, nip into [/pub/gimp/fonts](ftp://ftp.gimp.org/pub/gimp/fonts) and cadge some nice fonts as well.

Gimp for Windows really isn't a sick UNIX joke. It's available to download at the [gimp.org](http://gimp.org) site, although I must come clean and say that if you want to know more, I'm not the person to ask. I went from DR DOS to Apple to Unix to Linux and managed to bypass all things Bill in the process. I'm sure Bill's gutted about it.

As I said, installing Gimp is easy peasy lemon squeezy. Well, OK, it was for me, I just asked Rob. For those who don't enjoy intimate relations with their sys admin though, he's kindly agreed (as in "been cajoled") to pass on a few tips (see box).

A comprehensive set of installation instructions can be found on the [gimp.org](http://gimp.org) site. I'll write more about downloading plug-ins and fonts at a later date, but for now, let's crack on.

OK, with any luck, you should be staring at a funny little box with 25 icons and a couple of coloured squares in it. That's the Toolbox, the Gimp Nerve Centre, the cerebral cortex of the image manipulation world. Clicking on an icon will perform a certain task to an image, but first you've got to get an image and to do that we need a work space - a clean sheet of digital paper and that ladies and gentleman (cue dramatic music) brings us to our first tutorial.

Perhaps there's a couple of things I should mention before we start. I'm using the latest stable version of Gimp v 1.2.3 and have enough of an ego to assume everyone else in the universe is using it too. I use the abbreviations Rclick and Lclick for right and left mouse clicks and courier for describing the path to Gimp menu commands, file names and code. I've got RSI, so I try and limit the number of keystrokes I use, or it's straight back to the wacky world of Ibuprofen. Since, it seems, I've only got enough room in me bonce to learn a single set of keystrokes anyway, and I've been vi-ing a lot longer than I've been gimping, well it's no great personal loss. However, if keystrokes are your thing you can find a summary of all the best ones at <http://gimp-savvy.com/BOOK/index.html>. Finally, if you're just starting out then consider keeping a Gimp Diary to jot down all the functions and parameters you use on the way to achieving the finished result. It really is very easy to lose track.

OK, enough of the blurb, I've got clicky fingers. Let's go make some funky cow patterns.

Let's Moo!

I love cows. I love cows almost as much as I love chocolate and I really, \*really\* love chocolate. The following is based around the first Gimp tutorial I ever attempted (excuse me as I go all teary eyed here), which was written by Zack Beane. His was based on somebody else's. Us Gimpers we love to recycle.

OK, see File on the top left hand corner of the toolbox, click on it to bring up a list of options. If you select New that in turn will bring up the File New dialog box. Take a good look, but for now just accept the default setting by clicking OK on the bottom left hand side of the box. You should have a new 'canvas' area to work on. So far, so goody, lets fill it with something.

Move the cursor so it's over the window. Right click, select Filter and then Noise and then noisify from each subsequent drop down menu. Whap the slide bar up to 80 for each colour and click OK.

We want a black and white image, so lets ditch the colour. Again, make sure the cursor's over the window, right click, this time select Image – Colours – Desaturate. You will now have a rough grey stippled texture.

Next, Rclick – Filters – Noise – Spread, accept the defaults and watch the random noise begin to shift, forming areas with differing concentrations of dots. Repeat four times more to intensify the effect. You can re-do a command in a number of ways - Alt+f or Rclick – Filters – Repeat Last, whatever floats your boat.

Let's try blurring those edges just a bit by Rclick – Filters – Blur and then selecting either of the two Gaussian Blurs listed. Nobody's come up with a decent explanation as to which one to use in a particular situation, so feel free to chose your own path.

If you Rclick – Image – Colours – Auto – Stretch Contrast and then Rclick – Image – Colours – Threshold, ta da! You should now be seeing cow spots!

Go back to the toolbox and find the icon that's supposed to look like a knife (9th button along). This is the crop tool, it lets us chose which bit we want from an image and ditches the rest. Activate it by left clicking on it once, move the cursor back to the image window and then click and drag to select an area roughly 64 x 64 pixels in a part of your image that contains the most interesting spots.

Blur the spots again using Rclick – Filters – Blur – Gaussian Blur but this time set the radius to 2.0 to reduce the effect.

Better save the image before disaster strikes. Rclick – File – Save. As should do the trick. If you haven't created a directory for your Gimp work, you can do so through the Save As dialog box. The native Gimp format is .xcf . A good rule of thumb is to save any work in progress in this format, as it contains all the Gimp- specific information and then convert the final image to a file type of your choice once you've stopped fiddling with it. Let's save this file as moo .xcf .

Okey Dokey, time to add a dash of colour. The problem with all this artistic freedom is it can be a bit overwhelming when you first start out. For example, we can colour the cow spots in many different ways using a variety of tools and achieve hundreds of effects. Here's some simple ideas to play around with first.

Route 1. Create a new file, again just accept the default options, and then in the new window Rclick – Filters – Render – Clouds – Plasma and then Rclick – Filters – Map – Bump Map and play around with the parameters till you get an effect you like.

Route 2. Rclick – Filters – Distorts – Emboss doesn't add any colour, but looks classy (in a grey sort of way.)

Route 3. Rclick – Filters – Colours – Map – Alien Map 2, followed by Rclick – Filters – Colours – Map To Gradient and then choose a groovy sounding gradient from the drop down list.

Route 4. As route 3 and then create a new file and fill using Rclick – Filters – Render – Pattern – Qbist. Select a pattern that catches your eye, click OK and then it's back to moo .xcf to Rclick – Filters – Colours – Map – Sample Colourize.

Leave the destination side of the dialog box alone, but select your untitled qbist pattern in the sample drop down menu, click Get Sample Colours to preview the final effect and once you're happy, click Apply.

Just two more things to do before we finish. The first is to distort the cow pattern so it will tile seamlessly. The easiest way of achieving this is to use Rclick – Filter – Map – Make Seamless. Finally let's put moo to some practical use. Why don't we save it as a pattern so we can use it to fill in things like backgrounds or web images? Select Save As again, but this time name the file moo.pat and save it in the .gimp/patterns

directory. Rclick – Dialogs – Patterns to view the pattern selection dialog. Press refresh and voila, your newly created funky cow pattern is ready for your image filling pleasure.

If you've got this far, well done! Hopefully you've got a feel for where Gimp is coming from and created some cool patterns using some pretty complex tools in the process. Even if you're still confused, as long as you remember that Gimp is a graphics tool and it's very easy to use, then my work here is done.

---

## BSD Update

**Sam Smith**

Here's what happened in the last 2 months:

There is now a UKUUG BSD mailing list, for members to discuss BSD questions, issues with a vaguely UKUUG feel:

```
$ echo "subscribe bsd" | mail majordomo@ukuug.org
```

Remote exploits have been found in OpenSSH 2.3 to 3.3, other bugs in Apache httpd and the BSD resolver. FreeBSD 4.6.1 has been released as a point update to fix these major security issues. OpenBSD, NetBSD and Mac OS X have patches out – update to the latest stable/patch release.

```
http://www.FreeBSD.org
http://www.NetBSD.org/
http://www.OpenBSD.org/
http://www.apple.com/
```

OpenSSH now ships with Privelege Separation enabled by default. PrivSep is a feature where all of the authentication and authorization is done by an unpriviledged client – making another remote root exploit significantly harder.

```
http://www.citi.umich.edu/u/provos/ssh/privsep.html
```

The new FreeBSD team have been elected, including Mark Murray who is based in the UK.

```
http://election.uk.freebsd.org/
```

The Linux Developer's Conference gave away 250 FreeBSD 4.4 DVD/CD packs, which were very well received. Thanks to FreeBSD Services Limited for the donation:

```
http://www.freebsd-services.com/
http://www.ukuug.org/events/linux2002/
http://photos.codemonkey.org.uk/photo.php?id=134907
```

FreeBSD on the IBM S/390 prompts for a root device in the emulator

```
http://www.freebsd.org/cgi/query-pr.cgi?pr=bin/40355/
```

A NetBSD Live! CD has been produced which boots into KDE automatically, and includes major applications such as Mozilla, Gip, AbiWord, Xemacs, xpdf and KOffice, and is able to mount samba shares. A very useful tool for when you walk up to a machine for a demonstration: <ftp://ftp.netbsd.org/pub/NetBSD/iso/1.5.2/README.i386live>

Darniel Hartmeier gave a paper comparing iptables, ipFilter and OpenBSD's pf at USENIX, showing the relative differences, speed and features between the three

<http://www.usenix.org/events/usenix02/tech/freenix/hartmeier.html>

NetBSD 1.5.3 has been released as an update to their stable branch. NetBSD 1.6 should be released in early August.

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

BSD Meetings: semi regular meetings for BSD users are held in Manchester and London.

<http://www.bsdgroups.org.uk/>

---

## A Linux User Defects

**Simon Cozens**

FreeBSD? GNU Hurd? Windows? No, the operating system that's my tip for the OS of the future is nothing other than Apple's Mac OS X.

I first came across OS X at last year's FreeBSD conference, where Apple had provided a terminal room full of dual-processor PowerMacs running it. I suppose the first impression I got was that it provided a well-polished user experience; from the swish Dock animations, the sliding drawers and the "lickable" widgets to the seamless internationalisation and perfect Unicode support. (Very important to someone who handles a lot of East Asian text.) It seemed that Apple had finally found a stable platform on which to indulge their famous proclivities for "smooth" design.

And then when I opened up the terminal application and found a perfectly ordinary BSD system under the hood, my joy was complete - I realised that the dream of Unix on the desktop had been realised. It is indeed possible to have the best of both worlds.

I began saving for a Titanium PowerBook that day.

When it arrived, I really got my chance to explore the system. Putting NeXTStep on top of BSD on top of Mach might make for a precarious arrangement, but it has some huge advantages: the NeXT approach of separating libraries off into their own subdirectories and separating out library versions makes for a much tidier filesystem arrangement than simply bundling everything in `/usr/lib`; the use of loopback disk images (`.dmg`) for software distribution is particularly neat; the integrated CORBA-like "Services" where applications offer to perform tasks for each other. And then there are the little things: automatic text ligatures to make a font geek smile, while the antialiasing just looks beautiful; transparent windows (if you know how to get them); iTunes; and the wonder of the NeXT-ish `open(1)` command.

By this time, I've ordered a Mac for work and am using OS X pretty much everywhere. I'm also noticing that more and more of my hacker friends are appearing at conferences with iBooks and TiBooks. If you too go out and get an OS X system tomorrow, there are two pieces of software which supplement the operating system beautifully. First, OroborosX and the XDarwin X server. <http://julia.et.ic.ac.uk/adrian/software/oroborosx>

This gives you a rootless X server and Aqua-like window manager that, to a reasonable extent, makes your X applications appear just like other native Mac applications.

Second, "fink" <http://fink.sf.net> is essentially Debian on OS X. It gives you "apt" plus a repository of standard Debianised packages - if you're more used to the GNU "fileutils" than the BSD ones, you can install your favourites with one command. While it's not quite as big as the real Debian, 1191 packages ought to be enough to be going on with.

Of course, there are a few warts. It took me a long time to make the Terminal's ANSI capabilities behave. Packaging is in a mess - between Fink's packages, OS X's native packages and those applications which don't bother putting themselves in packages at all, things get a little confused. HFS+'s case-insensitive-but-case-preserving nature can cause a few nasty surprises. (Watch

with horror as LWP's "HEAD" utility for querying web servers overwrites your ordinary "head" command.) And just the other day I attempted to build a Darwin kernel, which made me decide I much preferred to look at Apple's software purely from the outside.

But all in all, I'd go so far as to say that OS X does precisely what I want from an operating system - it's flashy and easy to use, but it's powerful and gives you a full Unix underneath. What more could you ask for?

And, incidentally, no, I don't find it a problem having only one mouse button...

---

## **HTML Pocket Reference, 2nd Edition**

**Jennifer Niederst**

**O'Reilly and Associates**

**ISBN 0-596-00296-3**

**104 pp.**

**£ 8.06**

**reviewed by Andrew Cormack**

In the last issue Lyndsay Marshall asked for a small but comprehensive reference to HTML. For many years I kept a single A4 cribsheet to hand which provided all the information I needed to write HTML 3.2. However HTML 4 is a much bigger language and O'Reilly's 95 page pocket reference may be as concise as it is possible to get.

The reference starts with a list of tags grouped by type: block, inline, form, etc. Example templates are given for structures such as lists, tables and frames where tags are commonly used together. As a nudge towards best practice there is a note of the different ways to reference style sheets, though CSS is not covered in this volume. After this orientation section the bulk of the book consists of an alphabetical list of tags with a description of the function of each one and the attributes it can take. Finally there are tables of the HTML 2.0 and 4.01 character entities and a hexadecimal conversion table.

Sadly, HTML is no longer a single language and the guide has to cope with the variants implemented by Netscape version 6, Internet Explorer 6, Opera 5 and WebTV in addition to the official HTML 4.01 standard. For each tag, and some non-standard attributes, there is a list of the browsers and versions that support it.

So how well does the reference work in practice? The combination of alphabetical list and function index allows quick access to information for either a known tag or a required function. My own preference would be to group the descriptions by function and index these alphabetically - this allows browsing of a particular area of the language - but this is a question of personal taste and familiarity. The descriptions and attribute lists provide almost all the information an author is likely to need: my only failure was when searching for the conventional target frames such as top. The character entity tables have saved me a lot of trial and error.

Of course all this information is available on the web, but for those like me whose screen space is already overflowing, or who simply find printed, bound, paper an easier medium for quick reference, this pocket reference is well worth the shelf space. Even better, prop it between your screen and speaker so it is immediately available when needed.

---

## **System Performance Tuning, 2nd Edition**

**Gian-Paolo D Musumeci and Mike Loukides**

**O'Reilly and Associates**

**ISBN 0-596-00284-X**

**336 pp.**

**£ 28.60**

**reviewed by Andrew Cormack**

I read the first edition of Mike Loukides' *System Performance Tuning* when it was new and afterwards felt, for the first time, that I really understood how a unix system worked. That book concentrated on the operating system and its relationship to the underlying hardware: how to find out what the system was doing, how to tell when it was not performing to its full capability and what to do to resolve the situation. Twelve years on the book has been almost entirely rewritten and a rather different manual has emerged.

There is a much greater range of general purpose computing hardware available nowadays, and the rewrite spends much more time discussing how to select the right platform for a particular application. This concentration on hardware inevitably makes the book more vendor-specific than its predecessor. Most of the information here relates to Sun hardware and the Solaris operating system, indeed at times it comes close to being a detailed catalogue of Sun CPUs and buses and their different characteristics.

There are good sections on memory management and disk arrays. The former includes both physical and virtual memory and the sometimes unexpected interactions between them. The latter has a clear description of the various RAID levels, their aims and the trade-offs that each of them represents.

Although most of the book is specific to Solaris, there is also mention of the performance monitoring tools available for Linux and some of the options for tuning that operating system. However, throughout the book, almost all of the examples show systems in a good state of health. It would be more useful to an administrator trying to diagnose a slow computer to show these alongside listings from systems in distress and then suggest changes that might improve the situation.

In the introduction to the book, the author asks that it be read straight through as a whole, as effective performance tuning is likely to require an understanding of the whole system as well as individual components. I found this hard going as the chapters are not sufficiently well-structured to be read quickly as an overview. Important points are buried in the detail and it is hard to tell which paragraphs can be skimmed on a first reading. Unfortunately I suspect that most people will just concentrate on the sections that most interest them and will miss the broader picture.

If the first edition of the book was aimed at the administrator trying to understand how to get the best out of an existing system, this one concentrates instead on characterising a demanding application and specifying a new system to run it. For those involved in procuring Sun systems for demanding applications its detail will be useful, however it is also likely to date much faster than its predecessor. I doubt that readers will still be learning from this second edition in twelve years time.

## XSLT

**Doug Tidwell**

**O'Reilly and Associates**

**ISBN 0-596-00053-7**

**473 pp.**

**£ 28.50**

**reviewed by Sebastian Rahtz**

XSLT is one of the success stories of the XML saga. When the visionary Jon Bosak and Tim Bray brought off the short XML specification in 1997, they envisioned a fairly rapid follow-up of many companion specifications, including one for stylesheets. Bosak had been active in promoting the ISO DSSSL standard, which had itself only recently been completed, and it was initially suggested that a subset of DSSSL for XML could be quickly agreed upon. However, the committee which was set up decided that it was vital that the new language (XSL: Extensible Stylesheet Language) be expressed in XML, and that it should include both the transformation and formatting parts of DSSSL. A rapid period of development during 1998/1999 then split the original proposal into three parts: first, the XPath specification, providing a language for addressing parts of an XML tree structure; second, the XSLT language, for describing transformations of XML documents; and third, the poor relation, XSL Formatting Objects, which is an XML vocabulary for describing formatted pages. The XSLT and XPath specifications were finally completed in November 1999, and are described in a remarkably short and lucid pair of W3C Recommendations. For this, we have to thank James Clark, the 'only begetter', who had previously brought DSSSL to fruition.

Since 1999, the XML community has been very well served by XSLT. It has been fully (and well) implemented in Java, C, C++, and Python. It is delivered with Gnome in the Linux world, and Internet Explorer in the Microsoft world. It is easy to learn, easy to teach, and seems to fill the needs of the hoards of semi-programmers who want to do something to an XML document, but do not want the burden of a full-scale programming language. It is not, however, without its problems. The verbosity is off-putting; not many people find this

```
<xsl:call-template name="linkListContents">
  <xsl:with-param name="style">
    <xsl:text>frametoc</xsl:text>
  </xsl:with-param>
</xsl:call-template>
```

a succinct way of calling a function, and people brought up on imperative programming find it hard to accept that XSLT is a genuine functional language - no side-effects or changing of variables here.

Which brings us to Doug Tidwell's O'Reilly book about XSLT. Doug was (is?) an IBM evangelist for XML and XSLT, who used to do excellent introductory sessions at XML conferences, and it is plain that he was asked to develop his tutorial work into a book. From the preamble above, it should be obvious that such a book has two problems: a) it must explain possibly unfamiliar functional programming, and b) it must demonstrate a language with a very short history. We do not have long experience in XSLT, nor have we developed the canon of use cases, nifty tricks, programming patterns etc. Can Tidwell explain the language clearly, and can he inspire us with a programming 'style'?

The main body of the book is not very substantial; a chapter on XPath, a chapter on 'branching and control' (does this betray Tidwell's imperative past?), a section on creating links, a section on sorting, and a section on combining XML documents. The second half of the volume

provides reference sections on XSLT, XPath, then a separate function reference, and finally a FAQ-like set of questions and answers (ie ‘How do I implement an if-else statement?’).

Where is this going? What is the plan of the book? After hardly any time into the book, we are going down sidelines and dead ends. We jump around the language, and don’t get introduced to things in order. My annotations over the first 100 pages are all along the lines of ‘but he has not mentioned that feature yet’ and ‘why does he mention that now when it isn’t needed’. By p.89, we are writing a stylesheet which generates a stylesheet which emulates a ‘for loop’. Why? When you start reading about recursion, you are presented (p. 83) with an explanation in an undocumented and unexplained pseudocode notation. Why?

I could live with these problems, but not unless I trust the book technically. Which, sadly, I cannot. To take a simple example, on p.88 there is a long case statement which contains the line

```
<xsl:when test="starts-width($operator,'<')">
```

which plainly breaks the fundamental rules of XML (that ‘<’ must be expressed as ‘&lt;’). So the examples in the book have not been tested, or the typesetting has screwed up; this is not good enough. You can see the same mistake again in the reference section on p.323. Worse is the mistake of which there is an example of p.50, where Tidwell says that

```
/x/text()
```

will return all the text nodes of the child nodes of

```
<x>
```

which is simply wrong.

It’s a personal thing, but I do not like technical books which are over-joky, or over-American. One can ignore the spelling of ‘color’, but a section entitled ‘Using Recursion to Do Most Anything’ (p. 81)? No, it grates, as does ‘This elements creates a new variable named x, whose value is an empty string. (Please hold your applause until the end of the section.)’ (p. 79). Nor do I like books which lazily embed page after page of uncommented code, which is a constant feature of Tidwell’s XSLT opus. He commonly gives code listings which span 2 or 3 pages, and does not seem to have considered any formatting or presentation ideas to make them more readable.

I do not like O’Reilly books these days; they seem to have lost their charm. Those animals on the cover seem boring now; the choice of the Baskerville font does not suit relatively cheap production (it looks weedy and pale unless well-printed), and the basic page design never was nice. Who wants large bold italic unnumbered headings in a technical reference? I did not especially enjoy handling this XSLT book.

There are now 10 or more XSLT books in the shops, as well as numerous online guides (I like the ones at <http://www.zvon.org/> and most XML ‘bibles’ have a section on it. Unfortunately for most of them, the very first XSLT book, Mike Kay’s ‘XSLT Programmers Reference’ (Wrox Press 2000) was truly excellent, being definitive, well-written, and well-organised. It remains the volume of choice on my bookshelf, together with Jeni Tennison’s ‘XSLT and XPath On The Edge’ (M&T Books, 2001), which contains her insights as one of best-known XSLT programmers (curious, incidentally, that the UK has generated such a wealth of XSLT talent). I will not be keeping Doug Tidwell’s book around much, I am afraid. It is not a bad book, and some will appreciate it’s light style, but it is not a classic, and there are much better books to get hold of.

## News from O'Reilly

*Josette Garcia*

### **A batch job to add new user IDs**

Arnold Robbins shows a script to automate adding many user IDs at once.

<http://linux.oreillynet.com/pub/a/linux/2002/05/09/uid.html>

### **O'Reilly Mac OS X Conference**

The first O'Reilly Mac OS X Conference, September 30-October 3, 2002, in Santa Clara, CA, explores how Apple's completely rebuilt operating system is creating fertile ground for Mac users, \*nix programmers, and Java developers alike. Registration opens June 2002.

### **A bad, sad Hollywood ending?**

Tim O'Reilly comments on a recent "Business Week" article that points out the dangers to open source from the Hollings bill and the Broadcast Protection Discussion Group.

<http://www.oreillynet.com/cs/weblog/view/wlg/1428>

For more on these topics, don't miss Lawrence Lessig's keynote on "Freeing Culture" at the upcoming O'Reilly Open Source Convention.

<http://conferences.oreillynet.com/pub/w/15/keynotes.html>

### **The Sharp Zaurus: a lovely little computer**

Simson Garfinkel reviews Sharp's new Zaurus SL-5500, a Linux-based palmtop computer.

<http://linux.oreillynet.com/pub/a/linux/2002/05/23/zaurus.html>

### **Ship in a bottle**

David HM Spector shows us how to get the most out of one computer by running multiple operating systems on it. He covers programs such as WINE, DOSEMU, Bochs, and User-Mode Linux.

<http://linux.oreillynet.com/pub/a/linux/2002/05/30/enterprise.html>

### **An interview with Guido van Rossum**

The creator of Python discusses Python's latest releases and future plans in the "State of the Python Union" at OSCON. In a recent O'Reilly interview, Guido talks about the benefits of open source development.

<http://www.onlamp.com/pub/a/python/2002/06/04/guido.html>

For information on Guido's "State of the Python Union" session, see:

[http://conferences.oreillynet.com/cs/os2002/view/e\\_sess/2672](http://conferences.oreillynet.com/cs/os2002/view/e_sess/2672)

### **The strange case of the disappearing Open Source vendors**

Tim O'Reilly explains why open source is good for businesses even if it isn't always good for software vendors. "Customer lock-in is the real enemy of business, not the GPL," Tim says.

<http://www.oreillynet.com/pub/a/network/2002/06/28/vendor.html>

### **Wearable computers**

Remember when the Dick Tracy communicator watch seemed like something out of science fiction? Well here we are. Now what? Tim O'Reilly and Jon Orwant write about the fashion potential of wearable computers in the latest Ask Tim.

[http://www.oreilly.com/ask\\_tim/](http://www.oreilly.com/ask_tim/)

### **Seven common SSL pitfalls**

If you are deploying SSL-enabled applications with OpenSSL here are seven pitfalls you'll want to avoid, by the authors of "Network Security with OpenSSL."

<http://www.onlamp.com/pub/a/onlamp/2002/06/27/openssl.html>

## Shell Competition Results - Prime Numbers

*James Youngman*

The entries to the March shell competition were especially good, though no further entries were received following the previous issue of the newsletter (which lacked the shell puzzle column).

The winning entry is by Roger Andrews and is 53 bytes long; however, although this works on many Unix systems, it doesn't work on Linux because the Linux version of "factor" uses a different output format. Roger's entry is 86 bytes long. The first prize is therefore jointly awarded to Dick Porter's 86 byte entry.

Roger's program:

```
yes ''|pr -tn|sed 350q|factor|awk '$1>m{print;m=$1}'
```

Dick's program:

```
#!/bin/sh
dc -e [q]sa[lNp]sc[lDlNlD%0=a1-sDldx]sd[lNdd1=a1-sDldx1=c1-sNlex]se350sNlex
```

Dick and Roger both win their choice of book published by O'Reilly (up to a value of 30 pounds).

---

## Next issue of the Newsletter

The next issue of the newsletter will be published in October: all contributions from members and others are very welcome.

We would be particularly interested in short series of articles on, for instance, particular favourite tools, languages or techniques.

---

## **Contacts**

Charles Curran  
Council Chairman; Events; Newsletter  
Oxford  
Tel: 07973 231 870  
charles.curran@ukuug.org

James Youngman  
UKUUG Treasurer  
Manchester  
james.youngman@ukuug.org

David Hallowell  
Website  
Tyne and Wear  
david.hallowell@ukuug.org

Alasdair Kergon  
Events  
Reading  
alasdair.kergon@ukuug.org

Dr A V LeBlanc  
Newsletter  
Manchester  
owen.leblanc@ukuug.org

Roger Whittaker  
Schools; Newsletter  
Borehamwood  
roger.whittaker@ukuug.org

Jane Morrison  
UKUUG Secretariat  
PO Box 37  
Buntingford  
Herts  
SG9 9UQ  
Tel: 01763 273 475  
Fax: 01763 273 255  
office@ukuug.org