AUUGN

The Journal of AUUG Inc.

Volume 25 • Number 4
December 2004

Features:

News:

Regulars:

ISSN 1035-7521

Print post approved by Australia Post - PP2391500002

AUUGN

The journal of AUUG Inc.

Volume 25, Number 3
September 2004

Editorial

Frank Crawford <frank@auug.org.au>

Well, after many, many years of involvement with AUUGN, I've finally been roped into writing the editorial. In fact, AUUGN has a very long and distinguished history, providing important information to generations of Unix users. During that time, there have been a range of editors all of whom have guided it through ups and downs. Certainly you will know many of the recent ones, such as David Purdue (current AUUG President), Gunther Feuereisen and most recently Con Zymaris (Open Source Advocate). In fact, if I go back through the past AUUGNs, it is interesting that the editors were all well respected in the Unix community, spending a number of years undertaking duty for the community. These included such people as Jagoda Crawford (yes, my wife), John Carey and Peter Ivanov.

Each one of these editors has brought their own style and experiences to the role and the AUUGNs of each period reflect this. We move from the early periods where a lot of the content was on new developments within the Unix kernel, on to software development, and now to an era dominated by Open Source development.

If there is one thing constant throughout the entire time it is the difficulty in finding contributors. This has been tackled by different editors in their own ways. We have had republication of other user journals (e.g. USENIX's :login;) to publication of AUUG regional conferences, and on to reprinting of open source publications found around the Internet.

Well now we have a new era, Con has stepped down, and rather than putting the load on a single person, AUUGN will have an editorial committee, preparing each edition. Currently, this consists of Greg Lehey and myself, but we are keen to expand this by a few more, in an effort to spread the load. And as with previous changes, we have a "new" approach to finding contributions. AUUG has a huge body of work, from both the Annual Conference and regional meetings that should be seen more widely, especially by those who weren't able to attend these events.

Over and above this, we are always looking for new columns, book reviews and one off papers. In addition we will also be publishing items of interest from throughout the Open Source and Unix (yes there is some "proprietary" Unix) community. AUUG has a very wide and knowledgeable community and we want to see items of interest to all parts of it, and hope to see all those parts contributing.

Finally I'd like to thank all those who have contributed and continue to contribute, and that all the readers continue to find AUUGN an important part of their universe.

AUUG Membership and General Correspondence

The AUUG Secretariat

AUUG Inc
PO Box 7071
Baulkham Hills BC NSW 2153
Telephone: 02 8824 9511
or 1800 625 655 (Toll-Free)
Facsimile: 02 8824 9522
Email: auug@auug.org.au

AUUG Executive Director

Elizabeth Carroll
AUUG Inc
PO Box 7071
Baulkham Hills BC NSW 2153
liz@auug.org.au

AUUG Board of Directors
Email: auugexec@auug.org.au

President

David Purdue
Sun Microsystems
Level 6, 476 St Kilda Road
Melbourne, VIC 3004
Phone: +61 3 9869 6412,
Fax: +61 3 9869 6288
David.Purdue@auug.org.au

Vice-president

Steve Landers
Digital Smarties
PO Box 717 Willetton WA 6155
Business phone: +61 8 9313 6868
Business fax: +61 8 9313 6077
Steve.Landers@auug.org.au

Secretary

Jonathon Coombes
Cybersite Consulting Pty Ltd
34 Newcastle Road, Wallsend NSW 2287
Business Telephone: +61 2 4965 6989
Jonathon.Coombes@auug.org.au

Treasurer

Gordon Hubbard
Custom Technology Australia Pty Ltd
3 Spring Street,
Bus Tel: 02 9659 9590,
Bus Fax: 02 9659 9510
Gordon.Hubbard@auug.org.au

Ordinary board members

Grant Allen

Adrian Close
Cybersource Pty.Ltd.
4, 10 Queen Street Melbourne VIC 3000
Business Telephone: +61 3 9621 2377
Business Fax: +61 3 9621 2377
Mobile: +61 417 346 094
adrian@auug.org.au

Andrew Frederick Cowie
Operational Dynamics
GPO Box 4339
Sydney, NSW, 2001
Telephone: +61-2-9977-6866
Andrew.Cowie@auug.org.au

Enno Davids

Michael Still
Phone: +61 414 382 568
mikal@auug.org.au

Immediate past president:

Greg Lehey
PO Box 460
Echunga, SA, 5153
Bus. Tel (08) 8388 8286,
Mobile 0418 838 708,
Fax (08) 8388 8725
Greg.Lehey@auug.org.au

Returning officer:

Jason Ozolins

Assistant Returning officer:

David Baldwin

See page 61 for information about AUUGN.

AUUG Incorporated gratefully acknowledges the support of its corporate sponsor:

IBM®

President's Column

David Purdue <David.Purdue@auug.org.au>

> **"reshape: Function: transitive verb : to give a new form or orientation to : reorganise"**
>
> —*Merriam-Webster Dictionary*

As you should be aware, the AUUG Board are busy reshaping AUUG—in both what we do and how we do it—in order to improve member services and the value of an AUUG membership, to update the organisation and keep it relevant.

Within this change we are taking great care to keep AUUG's core values the same. One of our first tasks was to clearly state these values, as can be seen now on the AUUG home page:

"AUUG's role is building a community by connecting, supporting and promoting people with an interest in interoperable computing—which we see as a key to achieving fitness for purpose and value for money in Information and Communication Technology.

"Our members are professionals who care about

- Unix operating systems from proprietary vendors
- Linux, BSD and other Open Source operating systems
- Open standards, specifically those which enable the free and unrestricted interchange of information
- Open Source applications, regardless of the platform they run on
- other cool technical stuff"

One of the main areas of focus for us is events. Of course, our main event will still be our Annual Conference—AUUG 2005 will be held in Sydney on October 12—14, with tutorials to be held on October 9—11. Adrian Close <adrian@auug.org.au> has volunteered to act as Programme Committee Chair, and I'm sure he would like to hear from anyone with ideas for presentations or volunteers for the Programme Committee. We are just finalising the conference theme and invited speakers, so expect to see a call for papers real soon now.

We have also started the organisation of seven one-day symposia—smaller events that will be held in capital cities around the country. These will be:

- The Digital Pests Symposium—to be held in Melbourne on the 8th of February, this will feature presentations discussing how to handle spam, viruses, adware, spyware and other malicious programming. This is also being run by Adrian Close <adrian@auug.org.au>, who is looking for another one or two presentations, so drop him a line if you have any ideas.
- Canberra in April as a co-conference with linux.conf.au (LCA).
- The First Australian Unix Developer's Symposium—a conference for all those developing software in UNIX like environments. There will be two streams, one for newcomers to learn the basics of the UNIX programming philosophy and tools, and one for experienced programmers to discuss the latest and greatest development tools and techniques.
- The UNIX Systems Administration Symposium—focusing on the technical problems involved in maintaining UNIX systems and networks of UNIX systems.
- The AUUG Security Symposium—discussing all aspects of computer and network security, over the past few years this has been our most popular symposium.
- Open Source Userconf—many open source events are staged for the benefit of Open Source Developers. The Userconf will change the focus to those who use and rely on open source.
- AOSS7—The 7th Australian Open Source Symposium, a low-cost conference run by open source developers for open source developers.

In addition, we are looking at running another "Code-Con Goes Bush"—a weekend in the wilderness with just your laptop, a generator, a data projector and a gang of code-hacking geeks for company.

These events are all at different stages of organisation, but we will publish a full calendar in January 2005.

As ever—AUUG is a member based organisation, and the success of these events depends on finding volunteers to organise and run them (with lots of support from the board and AUUG's executive director, Liz Carroll). So if these events interest you, please consider putting your hand up to contribute—drop a note to auugexec@auug.org.au.

As has been previously noted, this is the last dead tree edition of AUUGN you will be receiving. However, AUUG will still distribute CD's of popular open source software, and AUUGN will appear on those—in some different bundles to facilitate reading online or printing your own paper copy. In 2005 we plan to move from distributing CDs to DVDs, so that we can send you even more software each quarter. Suggestions for content of these distributions should be sent to auugexec@auug.org.au.

By the way—many thanks to the team (including Greg Lehey and Frank Crawford) who are continuing to put AUUGN together while we search for a new editor. If you are at all interested in the role of AUUGN editor, helping out the editorial team or just contributing some content, please let us know.

In 2005 we will be assessing the effectiveness of several cost-cutting exercises that have been introduced in an attempt to lower the AUUG membership fee. We can't make any definitive statement regarding new fees at this stage, but rest assured that if membership fees are lowered, then existing memberships will be adjusted pro-rata, so you do not lose anything by renewing now.

So, the main message from the AUUG board is that the task of updating AUUG is well underway. As always, we need volunteers to make AUUG successful and to deliver services to members—volunteering to help AUUG is a good way to gain recognition in the community and to meet a lot of interesting people, so give it some thought, and then most importantly take a step forward and volunteer! You know where to find us!

AUUG is what ***you*** make it—so don't wait till tomorrow, get involved today!

My Home Network

Frank Crawford <frank@crawford.emu.id.au>

In this modern world, with everything getting more and more serious, more and more hectic, more and more full of rubbish, that Murphy is alive and well. Unfortunately, this week he decided to drop by my place for a visit, and still hasn't left.

It started last Saturday, when I was finishing this column and I received this little email:

```
Subject: SMART error (health) detected \
    on  host: bits.crawford.emu.id.au

This email was generated by the smartd
daemon running on host:
bits.crawford.emu.id.au in the domain:
(none)

The following warning/error was logged
by the smartd daemon:

Device: /dev/hda, FAILED SMART
   self-check. BACK UP DATA NOW!

For details see the SYSLOG (default:
/var/log/messages) for host:
bits.crawford.emu.id.au

You can also use the smartctl utility
for further investigation.
No additional email messages about this
problem will be sent.
```

Of course I knew immediately that it meant problems. :-)

Now for those who don't quite know what it is about, modern disks, particularly ATA-disks have a facility called Self-Monitoring, Analysis and Reporting Technology (SMART). This is a system to monitor the reliability of of the hard drive and predict drive failures. It includes the ability to perform different types of self-tests.

Within the Open Source community, the package *smartmontools* (*http://smartmontools.sourceforge.net/*) is used for this process, and consists of two executables, *smartd*, a daemon to monitor the status of the disks, and *smartctl*, a utility to control and report on disks.

Fairly obviously, this indicates there is a problem with the disk, however, while SMART should be able to "predict" drive failures, in this case it was more an issue of just "reporting". The disk had already generated sufficient hard-errors for the kernel to detect. The kernel detected the error and immediately remounted the filesystem read-only. Unfortunately, while this does prevent further problems, it didn't do anything to correct the current problem.

The next step in correction is to have the disk remap the faulty sector, which it will do when that block is rewritten. Unfortunately, since the disk block is not readable, that data is still lost. In this particular case, there were multiple blocks, all of which were in */home* and in particular my home directory.

Now knowing there was an error was a good start, fixing it was the obvious next step, and the easiest way to do this was to reboot and run *fsck*. In fact *fsck* does understand I/O errors and attempts to rewrite the block, it just isn't able to reconstruct the missing data. By the time it finished, the disk was again usable, although there was some loss of data. At that point I was hoping that was the end of it, but as you can guess, it wasn't. The next night, back comes the same problem, just at a different set of blocks, and around we go again.

So at this point, I have a disk that is dying, which needs to be replaced (hopefully under warranty) and a long term plan to ensure it doesn't happen again (which almost certainly involves mirrored disks). I will certainly be talking about it in a later column.

Anyway, now on to the original plan for this column, spam killing. One of the fastest growth areas is email spam, which seems to be increasing at an exponential rate. It is now threatening to overwhelm all my other email despite some current standard packages. In addition, I decided to improve the efficiency of my mail by the introduction of *milter* technology (more about that below).

First off, some background. If you have read my previous columns, you will be aware that I've been using *spamassassin* to segregate spam and good mail. In addition, I have recently switched to 'clamav' as my anti-virus package.

The normal process for much of this was to run it through *procmail*, but as this involves passing through a pipeline, it would be far better if *sendmail* had direct knowledge of the process. To enable this a facility called Mail Filter or *milter* has now been added to sendmail. This involves a well defined API which receives data from sendmail, and returns a status. From this *sendmail* then processes the mail appropriately.

Now, the first two milters I installed were *clamav-milter* and *spamass-milter*, which as expected run the mail data through *clamav* and *spamassassin*. This just keeps the whole of the delivery process within *sendmail*'s control, and generally performs it for the all users. In most cases the milter process involves a separate daemon running continuously, and a stub process which links into sendmail and makes the call to the daemon.

The first milter invoked is *clamav-milter*, which comes as a standard part of the whole *clamav* package. Its sole purpose is to reject email containing suspect attachments. Installation is simple, in fact, just being the installation of an RPM, start the daemon and a slight modification to *sendmail.mc*. The full list of milters is shown below. It specifies the full range of macros that are passed to all milters in their API, not all of which are used by each milter, followed by the actual milter definitions, and finally the order that the milters are invoked in.

> *Note: in the following listing, lines are continued with \. This is to fit into AUUGN's format. The originals are single longer lines.*

```
dnl Milter definitions.
dnl These are the minimum needed, but
   some defaults include these plus more.
define(‘confMILTER_MACROS_CONNECT’, ‘j, \
 _, {daemon_name}, {if_name}, {if_addr}, \
 b’)dnl
define(‘confMILTER_MACROS_HELO’, \
  ‘{tls_version}, {cipher}, {cipher_bits}, \
  {cert_subject}, {cert_issuer}, \
  {verify}’)dnl
dnl define(‘confMILTER_MACROS_ENVFROM’, \
  ‘i, {auth_type}, {auth_authen}, \
  {auth_ssf}, {auth_author}, \
  {mail_mailer}, {mail_host}, \
  {mail_addr}’)dnl
dnl define(‘confMILTER_MACROS_ENVRCPT’, \
  ‘{rcpt_mailer}, {rcpt_host}, \
  {rcpt_addr}’)dnl
dnl
dnl Use GreyListing to cut down spam
INPUT_MAIL_FILTER(‘greylist’,  \
  ‘S=local:/var/milter-greylist/ \
  milter-greylist.sock’)dnl
dnl
dnl Use ClamAV for anti-virus scanning
INPUT_MAIL_FILTER(‘clmilter’,‘S=local:  \
  /var/clamav/clmilter.socket, \
  F=, T=S:4m;R:4m’)dnl
dnl
dnl Use Spamass-milter to link to  \
   Spamassassin.
INPUT_MAIL_FILTER(‘spamassassin’,  \
  ‘S=local:/var/run/spamass.sock, F=, \
  T=C:15m;S:4m;R:4m;E:10m’)dnl
dnl
dnl Now order them.
define(‘confINPUT_MAIL_FILTERS’,  \
‘greylist, clmilter, spamassassin’)dnl
```

Each one of the macros passed to the milter (e.g. *confMILTER_MACROS_CONNECT*) specifies variables that are defined elsewhere in *sendmail.mc*.

The next block is the definition of the milter itself (*INPUT_MAIL_FILTER*) which specifies the milter handle (e.g. "greylist"), a socket to communicate through (*/var/milter-greylist/milter-greylist.sock*), and optionally some other arguments. The final option, *confINPUT_MAIL_FILTERS*, specifies the order that the milters are applied in. In this case, *greylist* first, followed by *clmilter* and finally *spamassassin*. If this option is not given, then the order of definition is used.

The next milter I installed was *spamass-milter*, which runs all mail through *spamassassin*, to perform standard anti-spam checks. While this isn't too different to the previous invocation, there are a couple of differences for a site wide setup. The biggest one being how to handle multiple addressees. For a single addressee, it attempts to match it against a local user (note, this happens even when it may not be delivered locally), but if there are multiple addressees, then the defaults for a "standard local user" are used. This may mean that some specific settings may not take effect in these cases. There are a couple of different milters to invoke spamassassin, but the one I am using was from DAG's repository (*http://dag.wieers.com/packages/spamass-milter/*).

The final milter and the one that is most interesting is *milter-greylist*, which is something I heard about at a recent conference. It implements a new scheme to combat spam and viruses, by putting a delay in the acceptance of the mail. The assumption is that valid mail will be spooled and re-sent later, hence a delay is acceptable, while spam or virus engines will not retry and so it will be dropped. A full description of this can be found at *http://projects.puremagic.com/greylisting/*.

While this is a bit of a fudge it does seem to work well. The standard process used by the *greylisting* milter is the first time a new mail item attempts to be delivered, it collects the details of who sent it, who it was addressed to and what host it will come from, and then rejects it with a temporary error. After a configurable time (usually 30mins) if mail with the same parameters is received, it will accept it. This triplet is also remembered for some time after to allow immediate delivery, rather than being "greylisted". Of course it is also configurable to allow certain hosts or users to be automatically accepted (i.e. "whitelisted") or even to accept mail that is authenticated.

While there are a couple of these milters around, the one I'm using is from *http://hcpnet.free.fr/milter-greylist/* and fits in fine. I have notice a dramatic drop in the spam and virus mail I am receiving ever since the installation of greylisting, although it isn't perfect. One of the biggest drawbacks is from spam, etc, that is redirected from some mail relay host.

In particular this comes into play with site relay hosts that are not using the same facilities, that are specified as higher value MX records. This is because spam engines will connect to these sites, deliver mail for you and then disconnect. Following this, the relay sites will then keep trying to deliver the mail, and in most cases succeed. Even worse, some spam engines deliberately search for these relays and use them instead of the target host. This has meant that I've dropped all the MX's for my domain, because it was causing such issues. Even worse, redirections from some of my old email addresses also suffer from this problem and are now the main source of my spam.

So, with these updates to my sendmail configuration, I have dramatically reduced my spam and more importantly, viruses, although I haven't completely eliminated it. However, with the use of some of the new features of *evolution*, in particular the filtering of junk mail, it is almost gone.

Of course all these updates won't stop me receiving good mail, and I hope you will send me such items. I'd really like to hear what you are up to with your home network, or even with your mail filtering. Better yet, by the time I get it, I should have my dead disk fixed and will be back on track!

See you all next year, have a good Christmas and a Happy New Year.

A Convert to the Fold

Replacing Windows™ infrastructure with Linux™

David Newall <davidn@rebel.net.au>

This paper was originally presented at the AUUG 2004 Conference "Who are you", and is reprinted with the permission of the author.

Abstract

A good friend and colleague asked me to help move his infrastructure from Windows to Linux. Ten out of thirteen machines running Windows, on a network with ADSL, Cable and 802.11b wireless, were converted to Linux.

A firewall/router was installed, which in addition to improving security, provided traffic management policies which resulted in savings in cost of data. Over 150 email accounts across twelve virtual domains were transferred with minimal disruption. A pleasant yet unanticipated side-benefit was a freeing up of computing infrastructure, permitting functional expansion of his facilities.

Acknowledgement

This paper could never have been written without the full and frank assistance of my colleague, who has chosen to remain anonymous. I thank him for his trust and time, and for choosing to make the switch to Linux, which has to be a hard decision for anybody.

Introduction

A good friend and colleague asked me to help him move his infrastructure from Windows 2000 to Linux. He made this decision after Microsoft took six months to repair a serious security vulnerability in Windows (*#AD20040210*) (*#TA04-041A*) his machines suddenly started sending large amounts of data to Russia. He had thirteen machines, ran his own software development and computer consultancy, and hosted twelve virtual domains (including e-shops) for clients This was a major change in infrastructure, to put it mildly.

Over the years I had shown him unix a few times, and each time he was impressed by the incremental improvements in ease of use and aesthetics Eventually he agreed it was as easy to install as Windows. He had discovered Open Source and SourceForge.net *[SOURCEFORGE]* and had a realistic idea of the effort involved in changing platform. He also had a number of new ideas that he wanted to develop, and pursuing them using an open source framework had an obvious financial appeal.

Designing a new system

Starting with an almost clean slate, we sat down to design the new system. Goals to be achieved were:

- Get away from Windows;
- Open solutions;
- Security;
- Ease of administration;
- Traffic management on external internet links; and
- Minimise surprise for customers.

New Network

The old facility had two separate networks, one comprised of Windows machines on an ADSL connected network with IP addresses assigned by the ISP, and the other comprised of Windows machines connected by 802.11b wireless to a home-grade Cable modem. The wireless access point provided DHCP service and performed NAT over the cable, and was principally used for web browsing. The fixed-address machines were a mixture of hosting servers, local servers and personal workstations. One machine, my colleague's personal, development workstation, (unwisely) crossed both networks.

We designed a new network comprising a central routing firewall with four segments. One segment is for official servers and is maintained at high-security. The second segment is for non-servers and is a medium-security segment. The final two segments areq external internet links, one ADSL and one wireless link to cable; both are considered low-security.

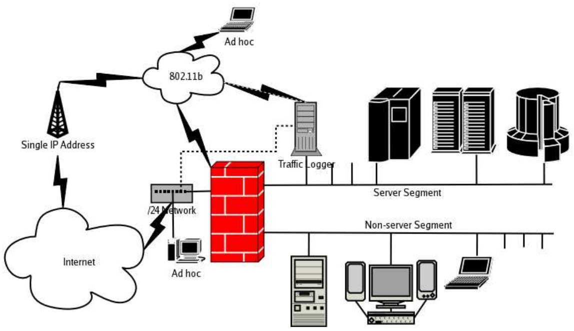


Figure 1: New Network Topology

External Links and Ad-hoc computers (Games)

The external links, almost by definition, have no security, and are suitable for ad-hoc networks for games, public access browsing and so on. Machines on the ADSL switch are given dynamic addresses from a small pool of public IP addresses, and route all of their packets via the ADSL modem. Machines on the wireless network are given dynamic addresses from a pool of non-routable addresses, and all of their packets are NATed via the cable modem.

Server Segment

The Server segment is the most restrictive environment. Access is granted according to specific type of services on specific machines. Machines are given fixed, public IP addresses. New network connections are preferentially routed via the ADSL link, but fall back to a NATed connection via the cable modem, if necessary.

Development Segment

The development segment has less restrictive protection. As a general rule, incoming connections are not permitted, but no restriction is made on connections to external machines. Most machines on the development network are dynamically assigned addresses from a pool of non-routable addresses. These machines have some limited trust relation with machines on the Server Segment, in that the server machines can be trust that the data really did come from a local machine.

Routing

Requests from the server segment are considered important, and so all connections are routed via the ADSL link if it's up, and otherwise fall back to a NATed connection via the cable modem. The development segment is considered less important, and for cost and load balancing purposes, it's web traffic is routed via the cable using a single NATed address. Other types of data are routed via ADSL, either using the machine's assigned public IP address, or by NATing the non-routable address using the router's own public address.

Devil-Linux

Prior to my arrival, my colleague spent time researching different Linux distributions, and found Devil-Linux *[DEVIL]* What he liked about it was that it runs directly from CD so there's no chance of programs being modified by intruders. (This concern might seem paranoid until you consider *why* he was replacing his infrastructure.) Devil stores configuration details on a floppy or USB disk and so when a machine dies, as they all eventually will, restoring service can be as simple as inserting the CD and configuration disks into a new machine and turning on power.

I particularly like Devil's configuration data being stored on a separate disk. It permits use of RCS to store different versions, and use of diff to examine changes prior to committing them. Bearing in mind that my colleague was somewhat new to unix, my being able to easily see his changes using one diff provided me with a sense of control and security that I might otherwise not have enjoyed.

I was concerned that the system would be too slow running off CD, but that hasn't been the case. The system is slower to initially load a program, but after first use it remains in cache and subsequent invocations are "immediate". I wouldn't recommend running unix from CD for a general workstation, but it does seem to work well for a dedicated purpose appliance.

fwbuilder

My colleague chose Fwbuilder *[FWBUILDER]* to maintain his firewall rules, and using a simple awk script, he also uses it to generate his DHCP and DNS tables. Fwbuilder is a simple GUI tool which you use to describe your network, with the ultimate aim of automatically generating a script for your firewall. In addition to Linux's *iptables*, it can emit rules for *Cisco PIX*, *ipf*, *ipfilter*, and *Berkeley packet filter*. Fwbuilder saves your data in an XML format, and this made it easy to write a script to extract MAC and IP addresses and generate DHCP and DNS tables for the complete network. As a nice touch, when you select "Install", fwbuilder calls our installation script which automatically builds new iptable rules, Bind DNS and ISC DHCP tables, transfers them to the firewall using SSH, and re-loads the appropriate services.

Network Logger

We built a logger for the two external links using a PC with extra network cards. Packets are logged using Ethereal *[ETHEREAL]* with log files rotated hourly. This is discussed in the interview, at the end of the paper.

Migrate Services

Email

The business provides email-account hosting for a number of domains, and we felt it was important to choose the email server wisely. Sendmail *[SENDMAIL]* has been the mainstay of mail servers for more than twenty years. It is powerful, flexible, and popular. Although the numbers are a year old, *[CREDENTIA03]* reported that Sendmail is still the leading mail transport, with more than twice as many servers as the second place getter, although it is losing market share *[BERNSTEIN]*. Sendmail used to be my choice, and I have no doubt that it could have easily met my colleague's needs, but I advised against it. A 30 second glimpse at a Sendmail configuration file might help explain why. *[See figure 0 on page 13]*.

For someone switching from Windows, I think I can sum it up with one, simple statement: *Sendmail is complex. It's true that most people configure Sendmail using the .mc files, and that it can and does do everything you're likely ever to need a mail system to do, but it is also takes a lot of learning. There have also been a fair number of security incidents during it's long history. We looked at alternative mail servers and felt that Courier Mail Server [COURIER] (#COURIER) would be better.*

Courier

My colleague's needs were modest.

His clients use Microsoft Outlook and Outlook Express, but don't use non-email functions, such as calendering. Therefore it was sufficient to provide SMTP, and POP3 or IMAP.

Courier is well documented and supports an extensive array of features. In addition to SMTP, IMAP and POP3 services, Courier includes Webmail, a minimal but functional web interface for sending and reading mail, which is a nice extra. Courier stores mail in Maildir format [BERNSTEIN2] (*#BERNSTEIN2*), which is much faster than traditional unix mailboxes (mbox). It supports virtual email addresses, quotas, LDAP, comes with a mailing list software, a rich filtering facility and can send and receive faxes. Users can be authenticated using a wide variety of mechanisms, and it's easy to write new mechanisms.

Web

My colleague originally had a number of virtual domains which he hosted, using a mixture of ASP and static pages. The plan was to move the static pages to Apache, and indeed Apache was loaded onto the new server, and to install a new IIS machine for the pages which couldn't be moved to Apache. Due to a late change in direction, my colleague instead chose to retire some applications and move others to third party servers. Apache is still running but is waiting for pages to serve!

Post Project Interview

Four months after completing the project I interviewed my colleague to hear his thoughts about the change. Here is an edited transcript of that interview.

Overall Impression

Q. What's the best thing about moving to unix?

A. The best thing? I think the thing that strikes me most in my mind, is the one to know that there isn't any code in any system that I'm running that hasn't been looked at by a number of pairs of eyes, which means there aren't secret spies in the code; and if there was one found it would be so devastating for the people who tried to get away with it and the companies who were associated with them, and anybody associated with it. The code is there so you can see it.

So it's the number of eyes, I'm really happy about the number of eyes that have looked at this thing before me so that I don't have to be as concerned.

Q. What's the worst thing about moving to unix?

A. I suppose it's the unknown. Like you know your backyard but there's always that little corner that you don't really know. Well, in linux, the corners are a lot bigger than in Windows. I know it works, I'm happy to install the upgrades, but I have to dedicate months of my life to some subcomponent to really get to know it.

Probably the only other thing that would give me a bit of angst is: I've reset my clock once. When does the same amount of threat that Microsoft is enduring now, when does the same sort of threat raise it's ugly head at Linux? Has Linux got five years before it suffers the same threat as Microsoft or has it got 20 years?

Q. What was the hardest thing about moving to unix?

A. The short answer is that there's a lot there. It's the whole back yard thing.

Current Use

Q. What machines are you running now?

A. Well right now the only ones I've got running are my new firewall, my web and email server, my samba server, and my windows game machine. And a notebook, well I've got three notebooks, and my windows development machine. There's another eight machines I don't have any use for at the moment. I've got some ideas for them like backup servers.

Q. How many Windows servers do you still run?

A. None. I've changed my entire focus. Now I'm focusing on developing a whole lot of ideas I've had over the years.

With Windows letting me down so harshly, I had an awakening of whether or not I really wanted to provide services to people. It's hard enough for Telstra and major corporations to do these things well, when you get flaky this and flake that that it becomes very, very painful, so I pretty much made the active decision to leave serving for others. The only things I'll be serving in the future are my own things. I'll serve my own web sites and my own email and my own this and my own that, so the only person who can be damaged by any sort of failure will be myself and maybe immediate family, no external company. What that new focus effectively means is that a lot of the stuff I was doing is no longer going to be necessary because it all came from the fact that I was putting together custom serving environments for customers.

Q. How central has unix been to your decision to change focus?

A. Well the ideas I've got are going to build on top of a lot of tools, and all of those tools are available in unix for free. I couldn't have afforded all those tools on a Windows system.

Q. What services did you shut down?

A. Web serving. There were thirteen domains, nine were burned to CD and handed back, and the other four have been moved somewhere else. In fact at the time I was going to put in a Windows server in to run the Windows domains, and the others, I could have done them on Apache. I'd still consider putting in an IIS server, but just make sure it was very, very tight.

Samba

The big one for me was Samba, because otherwise it's too hard to move stuff around. It's got to be easy to move stuff around otherwise it's too hard to play with. Primarily Samba lets all of my machines, based on who's using them and where they are, share data. If it wasn't for Samba I'd be running two networks: A windows network and a unix network and never the two shall meet. It's the glue that binds them together (other than whatever application is on them.)

Q. How do you share printers?

I don't print directly from linux at this stage because the bulk of my printing comes from windows.

Network Use and Security

Q. How has your traffic use changed?

A. I use about 1.2 to 1.8G a month. It turns out that before, about half of the traffic was the outside world trying to break in. The firewall keeps that out. My usage has about halved. I still do the same email, my usage there hasn't changed much. Maybe it's gone up a little as they put on a new staff member but it hasn't changed much.

Q. Have there been any security incidents since you converted, and if so,

have you been affected by them?

A. There have been some. I've got my firewall turned down so tight, even to the point of having special permission to contact the Windows patch server. Nothing goes on now unless I allow it.

What I do find is that when something stops working I was far more used to pulling out a windows machine and checking its network or monitor cable. Now there are different levels of routers and firewalls, and so finding a fault, there's a lot more things to look at to get the final verdict.

Q. Are you happy to have a level of control that detailed?

A. No, I wish I could do my job without it.

Q. Tell me about your network logger.

Over the last couple of months I've had a few times when an email wasn't getting through for one of my clients, so for a couple of hours I captured every packet and I narrowed it down to the particular IP address and and no, I'm perfectly fine, it's their end that's got the problem. Maybe seven or eight cases like that where the only way I could really truly know was by looking at the traffic.

A couple of weeks ago there was another Microsoft thing when all of a sudden the traffic crept up a bit on my only three Windows machines and it was just the people out there trying it on, but the machines behind the linux firewall were unaffected. I also looked at packets coming to the firewall that were trying to break in to windows machines which I keep on the outside of the firewall and don't have the benefit of the security.

No doubt there's many tools like it, but the graphic interface, where you can colourise it, you can look through huge amounts of stuff and sort it. It's very, very powerful. I used to have my notebook doing nothing but sniffing packets so that I could take it to clients. A thousand dollar notebook and Linux used for nothing else.

The beautiful part of it is my logger just copies hour long chunks of data, and even that alone, you can run your eye down it and see what's going on, and if you're really keen you can find out exactly what's going on. Up to this point in my life I've never had everything before me. You used to feel like you could look and touch and maybe think you knew what was going on but you never really did know what was going on. The way that Ethereal breaks the packets down and understands the packets, it's not just saying here's a packet and it's from this address to that address. There's just huge amounts of data which it gives.

Half the battle, of course, is security. With my early Windows environments, I could get in, but have each machine unto itself and I'd only do what was required on each machine. The problem was that the patches and the vulnerabilities for windows were coming out so quickly, and trying to keep ten machines unbelievably, perfectly aligned with no gaps. Somehow I got a gap in there and someone got in and re-jigged me so that it was no longer a single virus but it was my security had been compromised and once you've got that you might as well reformat and start again. But why do it with windows?

Final Thoughts

Q. Is there anything you'd have done differently?

A. No. The only thing that might have changed was because Sourceforge had a certain set of software available at that particular time. At another time it might have been a different choice, or perhaps something might not have been available, but apart from that I'd have done it the same.

Q. What advice would you give someone contemplating making the same switch?

A. Go to Sourceforge.

Spend about a month looking at what's there, and what you need, because everything's there and you've got multiple choices for everything. It doesn't really matter which one you choose, so just pick one and go with it.

References

[AD20040210-2]
eEye Digital Security Advisory AD20040210-2, *Microsoft ASN.1 Library Bit String Heap Corruption*, (*http://www.eeye.com/html/Research/Advisories/AD20040210-2.html*)

[AD20040210]
eEye Digital Security Advisory AD20040210, *Microsoft ASN.1 Library Length Overflow Heap Corruption*, (*http://www.eeye.com/html/Research/Advisories/AD20040210.html*)

[BERNSTEIN2]
D. J. Bernstein, *Using maildir format*, (*http://cr.yp.to/proto/maildir.html*)

[BERNSTEIN]
D. J. Bernstein, *Internet host SMTP server survey*, (*http://cr.yp.to/surveys/smtpsoftware6.txt*)

[COURIER]
Double Precision, Inc, *Courier Mail Server*, (*http://www.courier-mta.org*)

[CREDENTIA02]
Credentia, *E-Mail Server Survey Results*, (*http://www.credentia.cc/surveys/smtp/200212/*)

[CREDENTIA03]
Credentia, *E-Mail Server Survey Results for*

April 2003 (*http://www.credentia.cc/surveys/smtp/200304/*)

[DEVIL]
Devil-Linux, (*http://www.devil-linux.org*)

[ETHEREAL]
Ethereal, (*http://www.ethereal.com*)

[FEDORA]
Fedora™ Project, *Fedora Core 1*, (*http://fedora.redhat.com*)

[FWBUILDER]
Firewall Builder, (*http://www.fwbuilder.org*)

[MS04-007]
Microsoft Security Bulletin MS04-007, *ASN.1 Vulnerability Could Allow Code Execution (828028)*, (*http://microsoft.com/technet/security/bulletin/MS04-007.asp*)

[SENDMAIL]
The Sendmail Consortium, *Sendmail*, (*http://sendmail.org*)

[SOURCEFORGE]
SourceForge.net, (*http://sourceforge.net*)

[TA04-041A]
US-CERT Technical Cyber Security Alert, TA04-041A, *Multiple Vulnerabilities in Microsoft ASN.1 Library*, (*http://www.us-cert.gov/cas/techalerts/TA04-041A.html*)

About David Newall

David is a local unix luminary, and a software expert of almost 30 years experience. His first programming language was APL, and then Pascal, BASIC and FORTRAN in quick succession. He has worked on a wide variety of machines and environments, including the venerable IBM 1130, Apple][, Cyber 173, Vax and, of course, the ubiquitous "Wintel."

David discovered UNIX in the early 1980's, and his first commercial use of it was in 1986, using UNIX System V on a Motorola 68000 made by Stride Micro. He is very vocal within his local linux community and has been a member of AUUG for over ten years.

As an independent, private consultant, David specialises in multi-tiered, client/server systems, or "making computers talk to computers."

Windows™ and Microsoft™®; apparently are registered trademark thingos belonging to Microsoft Corporation. Linux™ is not; it belongs to Linus Torvalds. Other non-attributed marks may appear in this paper, and they belong to their respective owners.

This document was written using *vi*. The diagrams were composed using *dia*.

```
#
# Copyright (c) 1998-2003 Sendmail, Inc. and its suppliers.
#       All rights reserved.
# Copyright (c) 1983, 1995 Eric P. Allman.  All rights reserved.
# Copyright (c) 1988, 1993
#       The Regents of the University of California.  All rights reserved.
#
# By using this file, you agree to the terms and conditions set
# forth in the LICENSE file which can be found at the top level of
# the sendmail distribution.
#
#
######################################################################
######################################################################
#####
#####           SENDMAIL CONFIGURATION FILE
#####
##### built by bhcompile@bugs.devel.redhat.com on Tue Oct 28 16:06:03 EST 2003
##### in /usr/src/build/320829-i386/BUILD/sendmail-8.12.10/cf/cf
##### using ../ as configuration include directory
#####
######################################################################
#####
#####   DO NOT EDIT THIS FILE!  Only edit the source .mc file.
#####
######################################################################
######################################################################
#####  $Id: cfhead.m4,v 8.108.2.3 2003/04/03 17:51:51 ca Exp $  #####
#####  $Id: cf.m4,v 8.32 1999/02/07 07:26:14 gshapiro Exp $  #####
#####  setup for Red Hat Linux  #####
#####  $Id: linux.m4,v 8.13 2000/09/17 17:30:00 gshapiro Exp $  #####
#####  $Id: local_procmail.m4,v 8.21.42.1 2002/11/17 04:25:07 ca Exp $  #####
#####  $Id: no_default_msa.m4,v 8.2 2001/02/14 05:03:22 gshapiro Exp $  #####
[...]
#####  $Id: proto.m4,v 8.649.2.24 2003/08/04 21:14:26 ca Exp $  #####
# level 10 config file format
V10/Berkeley
# override file safeties - setting this option compromises system security,
# addressing the actual file configuration problem is preferred
# need to set this before any file actions are encountered in the cf file
#O DontBlameSendmail=safe
# default LDAP map specification
# need to set this now before any LDAP maps are defined
#O LDAPDefaultSpec=-h localhost
##################
#   local info   #
##################
# my LDAP cluster
# need to set this before any LDAP lookups are done (including classes)
#D{sendmailMTACluster}$m
Cwlocalhost
# file containing names of hosts for which we receive email
Fw/etc/mail/local-host-names
[...]
###############
#   Options   #
###############
# strip message body to 7 bits on input?
O SevenBitInput=False
# 8-bit data handling
#O EightBitMode=pass8
# wait for alias file rebuild (default units: minutes)
O AliasWait=10
# location of alias file
O AliasFile=/etc/aliases
# minimum number of free blocks on filesystem
O MinFreeBlocks=100
[...]
###########################
#   Message precedences   #
###########################
Pfirst-class=0
Pspecial-delivery=100
Plist=-30
Pbulk=-60
Pjunk=-100
#####################
#   Trusted users   #
#####################
# this is equivalent to setting class "t"
Ft/etc/mail/trusted-users
Troot
Tdaemon
Tuucp
#########################
#   Format of headers   #
```

```
#########################
H?P?Return-Path: <$g>
HReceived: $?sfrom $s $.$?_($?s$|from $.$_)
$.$?{auth_type}(authenticated$?{auth_ssf} bits=${auth_ssf}$.)
$.by $j ($v/$Z)$?r with $r$. id $i$?{tls_version}
(version=${tls_version} cipher=${cipher} bits=${cipher_bits} verify=${verify})$.$?u
for $u; $|;
$.$b
H?D?Resent-Date: $a
H?D?Date: $a
H?F?Resent-From: $?x$x <$g>$|$g$.
H?F?From: $?x$x <$g>$|$g$.
H?x?Full-Name: $x
# HPosted-Date: $a
# H?l?Received-Date: $b
H?M?Resent-Message-Id: <$t.$i@$j>
H?M?Message-Id: <$t.$i@$j>
#
######################################################################
######################################################################
#####
#####                   REWRITING RULES
#####
######################################################################
######################################################################
############################################
###  Ruleset 3 -- Name Canonicalization  ###
############################################
Scanonify=3
# handle null input (translate to <@> special case)
R$@                     $@ <@>
# strip group: syntax (not inside angle brackets!) and trailing semicolon
R$*                     $: $1 <@>                       mark addresses
R$* < $* > $* <@>       $: $1 < $2 > $3                 unmark <addr>
R@ $* <@>               $: @ $1                         unmark @host:...
R$* [ IPv6 : $+ ] <@>   $: $1 [ IPv6 : $2 ]             unmark IPv6 addr
R$* :: $* <@>           $: $1 :: $2                     unmark node::addr
R:include: $* <@>       $: :include: $1                 unmark :include:...
R$* : $* [ $* ]         $: $1 : $2 [ $3 ] <@>           remark if leading colon
[...]
# handle virtual users
R$+                     $: <!> $1               Mark for lookup
R<!> $+ < @ $={VirtHost} . >    $: < $(virtuser $1 @ $2 $@ $1 $: @ $) > $1 < @ $2 . >
R<!> $+ < @ $=w . >     $: < $(virtuser $1 @ $2 $@ $1 $: @ $) > $1 < @ $2 . >
R<@> $+ + $+ < @ $* . >
$: < $(virtuser $1 + + @ $3 $@ $1 $@ $2 $@ +$2 $: @ $) > $1 + $2 < @ $3 . >
R<@> $+ + $* < @ $* . >
$: < $(virtuser $1 + * @ $3 $@ $1 $@ $2 $@ +$2 $: @ $) > $1 + $2 < @ $3 . >
R<@> $+ + $* < @ $* . >
[...]
# handle local:user@host syntax -- ignore host part
R< $+ @ $+ > $* < @ $* >        $: < $1 > $3 < @ $4 >
# handle local:user syntax
R< $+ > $* <@ $* > $*           $#local $@ $2@$3 $: $1
R< $+ > $*                      $#local $@ $2    $: $1
###################################################################
###  Ruleset 93 -- convert header names to masqueraded form     ###
###################################################################
SMasqHdr=93
# do not masquerade anything in class N
R$* < @ $* $=N . >      $@ $1 < @ $2 $3 . >
R$* < @ *LOCAL* >       $@ $1 < @ $j . >
[...]
# check sender address: user@address, user@, address
R<$+> $+ < @ $* >       $: @<$1> <$2 < @ $3 >> $| <F:$2@$3> <U:$2@> <D:$3>
R<$+> $+                $: @<$1> <$2> $| <U:$2@>
R@ <$+> <$*> $| <$+>    $: <@> <$1> <$2> $| $>SearchList <+ From> $| <$3> <>
R<@> <$+> <$*> $| <$*>  $: <$3> <$1> <$2>               reverse result
# retransform for further use
R<?> <$+> <$*>          $: <$1> $2      no match
R<$+> <$+> <$*>         $: <$1> $3      relevant result, keep it
[...]
##################################################
###   Local and Program Mailer specification   ###
##################################################
#####  $Id: local.m4,v 8.58 2000/10/26 01:58:29 ca Exp $  #####
#
#  Envelope sender rewriting
#
SEnvFromL
R<@>                    $n                      errors to mailer-daemon
R@ <@ $*>               $n                      temporarily bypass Sun bogosity
R$+                     $: $>AddDomain $1       add local domain if needed
R$*                     $: $>MasqEnv $1         do masquerading
#
#  Envelope recipient rewriting
```

```
#
SEnvToL
R$+ < @ $* >            $: $1                   strip host part
#
#  Header sender rewriting
#
SHdrFromL
R<@>                    $n                      errors to mailer-daemon
R@ <@ $*>               $n                      temporarily bypass Sun bogosity
R$+                     $: $>AddDomain $1       add local domain if needed
R$*                     $: $>MasqHdr $1         do masquerading
#
#  Header recipient rewriting
#
SHdrToL
R$+                     $: $>AddDomain $1       add local domain if needed
R$* < @ *LOCAL* > $*    $: $1 < @ $j . > $2
#
#  Common code to add local domain name (only if always-add-domain)
#
SAddDomain
R$* < @ $* > $*         $@ $1 < @ $2 > $3       already fully qualified
R$+                     $@ $1 < @ *LOCAL* >     add local qualification
Mlocal,         P=/usr/bin/procmail, F=lsDFMAw5:/|@qSPfhn9, S=EnvFromL/HdrFromL, R=EnvToL/HdrToL,
T=DNS/RFC822/X-Unix,
A=procmail -t -Y -a $h -d $u
Mprog,         P=/usr/sbin/smrsh, F=lsDFMoqeu9, S=EnvFromL/HdrFromL, R=EnvToL/HdrToL, D=$z:/,
T=X-Unix/X-Unix/X-Unix,
A=smrsh -c $u
```

Figure 2: Extract from *sendmail.cf* file

A commentary on the Sixth Edition UNIX Operating System, part 2

John Lions

This is the second article in a series of reprints of the infamous “Lions Book”, written by John Lions around 1977.

The copyright notice below is the original. It is now of historical importance only. The current version is reproduced with the kind permission of Peter Salus, who has full rights to reproduce the documents.

The format of this installment has been adapted to more closely resemble the original. We’re not trying to reproduce the original *nroff* output: instead we’re aiming for a reasonable approximation of what it might have looked like had it been set in *troff*.

Chapter 2

Fundamentals

UNIX runs on the larger models of the PDP11 series of computers manufactured by Digital Equipment Corporation. This chapter provides a brief summary of certain selected features of these computers with particular reference to the PDP11/40.

If the reader has not previously made the acquaintance of the PDP11 series then he is directed forthwith to the “PDP11 Processor Handbook”, published by DEC.

A PDP11 computer consists of a processor (also called a CPU connected to one or more memory storage units and peripheral controllers via a bidirectional parallel communication line called the “Unibus”.

The Processor

The processor, which is designed around a sixteen bit word length for instructions, data and program addresses, incorporates a number of high speed registers.

Processor Status Word

This sixteen bit register has subfields which are interpreted as follows:

bits	*description*
14,15	current mode (00 = kernel;)
12,13	previous mode (11 = user;)
5,6,7	processor priority (range 0..7)
4	trap bit
3	N, set if the previous result was negative
2	Z, set if the previous result was zero
1	V, set if the previous result gave an overflow
0	C, set if the previous operation gave a carry

The processor can operate in two different modes: kernel and user. Kernel mode is the more privileged of the two and is reserved by the operating system for its own use. The choice of mode determines:

- The set of memory management segmentation registers which is used to translate program virtual addresses to physical addresses;
- The actual register used as *r6*, the “stack pointer”;
- Whether certain instructions such as “halt” will be obeyed.

General Registers

The processor incorporates a number of sixteen bit registers of which eight are accessible at any time as “general registers”. These are known as

r0, *r1*, *r2*, *r3*, *r4*, *r5*, *r6* and *r7*.

The first six of the general registers are available for use as accumulators, address pointers or index registers. The convention in UNIX for the use of these registers is as follows:

r0, *r1* are used as temporary accumulators during expression evaluation, to return results from a procedure, and in some cases to communicate actual parameters during a procedure call;

r2, *r3*, *r4* are used for local variables during procedure execution. Their values are almost always stored upon procedure entry, and restored upon procedure exit;

r5 is used as the head pointer to a “dynamic chain” of procedure activation records stored in the current stack. It is referred to as the “environment pointer”.

The last two of the “general registers” do have a special significance and are to all intents, “special purpose”:

r6 (also known as “sp”) is used as the stack pointer. The PDP11/40 processor incorporates two separate registers which may be used as “sp”, depending on whether the processor is in kernel or user mode. No other one of the general registers is duplicated in this way;

r7 (also known as “pc”) is used as the program instruction address register.

Instruction Set

The PDP11 instruction set includes double, single and zero operand instructions. Instruction length is usually one word, with some instructions being extended to two or three words with additional addressing information.

With single operand instructions, the operand is usually called the “destination”; with double operand instructions, the two operands are called the “source” and “destination”. The various modes of addressing are described later.

The following instructions have been used in the file “m40.s” i.e. the file of assembly language support routines for use with the 11/40 processor. Note that N, Z, V and C are the condition codes i.e. bits in the processor status word (“ps”), and that these are set as side effects of many instructions besides just “bit”, “cmp” and “tst” (whose stated function is to set the condition codes).

adc Add the contents of the C bit to the destination;

add Add the source to the destination;

ash Shift the contents of the defined register left the number of times specified by the shift count. (A negative value implies a right shift.);

ashc Similar to “ash” except that two registers are involved;

asl Shift all bits one place to the left. Bit 0 becomes 0 and bit 15 is loaded into C;

asr Shift all bits one place to the right. Bit 15 is replicated and bit 0 is loaded into C;

beq Branch if eaual, i.e. if Z = l;

bge Branch if greater than or equal to, i.e. if N = V;

bhi Branch if higher, i.e if C = 0 and Z = 0;

bhis Branch if higher or the same, i.e. if C = 0;

bic Clear each bit to zero in the destination that corresponds to a non-zero bit in the source;

bis Perform an “inclusive or” of source and destination and store the result in the destination;

bit Perform a logical “and” of the source and destination to set the condition codes;

ble Branch if greater than or equal to, i.e if Z = 1 or N = V;

blo Branch if lower (than zero), if C = l;

bne Branch if not equal (to zero), i.e. if Z = 0;

br Branch to a location within the range (.˜-128, .˜+127) where “.” is the current location;

clc Clear C;

clr Clear destination to zero;

cmp Compare the source and destination to set the condition codes. N is set if the source value is less than the destination value;

dec Subtract one from the contents of the destination;

div The 32 bit two’s complement integer stored in rn and r(n+l) (where n is even) is divided by the source operand. The quotient is left in rn, and the remainder in r(n+l);

inc Add one to the contents of the destination;

jmp Jump to the destination;

jsr Jump to subroutine. Register values are shuffled as follows: pc, rn, --(sp) = dest., pc, rn

mfpi Push onto the current stack the value of the designated word in the “previous” address space;

mov Copy the source value to the destination;

mtpi Pop the current stack and store the value in the designated word in the “previous” address space;

mul Multiply the contents of rn and the source. If n is even, the product is left in rn and r(n+l);

reset Set the INIT line on the Unibus for 10 milliseconds. This will have the effect of reinitialising all the device controllers;

ror Rotate all bits of the destination one place to the right. Bit 0 is loaded into C, and the previous value of C is loaded into bit 15;

rts Return from subroutine. Reload pc from rn, and reload rn from the stack;

rtt Return from interrupt or trap. Reload both pc and ps from the stack;

sbc Subtract the carry bit from the destination;

sob Subtract one from the designated register. If the result is not zero, branch back “offset” words;

sub Subtract the source from the destination;

swab Exchange the high and low order bytes in the destination;

tst Set the condition codes, N and Z, according to the contents of the destination;

wait Idle the processor and release the Unibus until a hardware interrupt occurs.

The “byte” version of the following instructions are used in the file “m40.s”, as well as the “word” versions described above:

bis
clr
cmp
inc
mov
tst

Addressing Modes

Much of the novelty and complexity of the PDP11 instruction set lies in the variety of addressing modes which may be used for defining the source and destination operands.

The addressing modes which are used in “m40.s” are described below.

Register Mode. The operand resides in one of the general registers, e.g.

```
clr  r0
mov  rl,r0
add  r4,r2
```

In the following modes, the designated register contains an address value which is used to locate the operand. *Register Deferred Mode.* The register contains the address of the operand, e.g.

```
inc  (rl)
asr  (sp)
add  (r2),rl
```

Autoincrement Mode. The register contains the address of the operand. As a side effect, the register is incremented after the operation, e.g.

```
clr  (rl)+
mfpi (r0)+
mov  (r1)+,r0
mov  r2,(r0)+
cmp  (sp)+,(sp)+
```

Autodecrement Mode. The register is decremented and then operand, e.g.

```
inc  -(r0)
mov  -(r1),r2
mov  (r0)+,-(sp)
clr  -(sp)
```

Index Mode. The register contains a value which is added to a sixteen bit word following the instruction to form the operand address, e.g.

```
clr  2(r0)
movb 6(sp),(sp)
movb _reloc(r0),r0
mov  -10(r2),(rl)
```

Depending on your viewpoint, in this mode the register is either an index register or a base register. The latter case actually predominates in “m40.s”. The third example above is actually one of the few uses of a register as an index register. (Note that “_reloc” is an acceptable variable name.) There are two addressing modes whose use is limited to the following two examples:

```
jsr  pc,(r0)+
jmp  *0f(r0)
```

The first example involves the use of the “*autoincrement deferred*” mode. (This occurs in the routine “call” on lines 0785, 0799.) The address of a routine intended for execution is to be found in the word addressed by r0, i.e. two levels of indirection are involved. The fact that r0 is incremented as a side effect is not relevant in this usage.

The second example (which occurs on lines 1055,

1066) is an instance of the "*index deferred*" mode. The destination of the "jump" is the content of the word whose address is labelled by "0f" *plus* the value of r0 (a small positive integer). This is a standard way to implement a multi-way switch.

The following two modes use the program counter as the designated register to achieve certain special effects. *Immediate Mode.* This is the pc autoincrement mode. The operand is thus extracted from the program string, i.e. it becomes an immediate operand, e.g.

```
add  $2,r0
add  $2,(rl)
bic  $17,r0
mov  $KISA0,r0
mov  $77406,(rl)+
```

Relative Mode. This is the pc index mode. The address relative to the current program counter value is extracted from the program string and added to the pc value to form the absolute address of the operand, e.g.

```
bic  $340,PS
bit  $l,SSR0
inc  SSR0
mov  (sp),KISA6
```

It may be noted that each of the modes "index", "index deferred", "immediate" and "relative" extends the instruction size by one word.

The existence of the "autoincrement" and "autodecrement" modes, together with the special attributes of r6, make it conveniently possible to store many operands in a stack, or LIFO list, which grows downwards in memory. There are a number of advantages which flow from this: code string lengths are shorter and it is easier to write position independent code.

Unix Assembler

The UNIX assembler is a two pass assembler without macro facilities. A full description may be found in the "UNIX Assembler Reference Manual" which is contained in the "UNIX Documents" The following brief notes should be of some assistance:

(a) a string of digits may define a constant number. This is assumed to be an octal number unless the string is terminated by a period ("."), when it is interpreted as a decimal number.

(b) The character "/" is used to signify that the rest of the line is a comment;

(c) If two or more statements occur on the same line, they must be separated by semicolons;

(d) The character "." is used to denote the current location;

(e) UNIX assembler uses the characters $ and "*" where the DEC assemblers use "respectively.

(f) An identifier consists of a set of alphanumeric characters (including the underscore). Only the first eight characters are significant and the first may not be numeric;

(g) Names which occur in "C" programs for variables which are to be known globally, are modified by the addition of a prefix consisting of a single underscore. Thus for example the variable "_regloc" which occurs on line 1025 in the assembly language file, "m40.s", refers to the same variable as "regloc" at line 2677 of the file, "trap.c";

(h) There are two kinds of statement labels: name labels and numeric labels. The latter consist of a single digit followed by a colon, and need not be unique. A reference to "nf" where "n" is a digit, refers to the first occurrence of the label "n:" found by searching forward. A reference to "nb" is similar except that the search is conducted in the backwards direction;

(i) An assignment statement of the form

```
identifier = expression
```

associates a value *and* type with the identifier. In the example

```
. = 60+^+.
```

the operator ’**+**^+’ delivers the value of the first operand and the type of the second operand (in this case, "location");

(j) The string quote symbols are "<" and ">".

(k) Statements of the form

```
        .globl   x, y, z
```

serve to make the names "x", "y" and "z" external;

(l) The names "_edata" and "_end" are loader pseudo variables which the define the size of the data segment, and the data segment plus the bss segment respectively.

Memory Management

Programs running on the PDP11 may address directly up to 64K bytes (32K words) of storage. This is consistent with an address size of sixteen bits. Since it is economical and not unreasonable to do so the larger PDP11 models may be equipped with larger amounts of memory (up to 256K bytes for the PDP11/40) plus a mechanism for converting sixteen bit *virtual* (program) addresses into *physical* addresses of eighteen bits or more. The mechanism, which is known as the memory management unit, is simpler on the PDP11/40 than on the 11/45 or the 11/70.

On the PDP11/40 the memory management unit consists of two sets of registers for mapping virtual addresses to physical addresses. These are known as "active page registers" or "segmentation registers". One set is used when the processor is in user mode and the other set, in kernel mode. Changing the contents of these registers changes the details of these mappings. The ability to make these changes is a privilege that the operating system keeps firmly to itself.

Segmentation Registers.

Each set of segmentation registers is composed of eight pairs, each consisting of a "*page address register*" (PAR) and a "*description register*" (PDR).

Each pair of registers controls the mapping of one *page* i.e. one eighth part of the virtual address space which 8K bytes (4K words).

Each page may be regarded as an aggregate of 128 blocks, each of 64 bytes (32 words). This latter size is the "grain size" for the memory mapping function, and as a practical consequence, it is also the "grain size" for memory allocation.

Any virtual address belongs to one page or other. The corresponding physical address is generated by adding the relative address within the page to the contents of the corresponding PAR to form an extended address (18 bits on the PDP11/40 and 11/45; 22 bits on the 11/70).

Thus each page address register acts as a relocation register for one page.

Each page can be divided on a 32 word boundary into two parts, an upper part and lower part. Each such part has a size which is a multiple of 32 words. In particular one part may be null, in which case the other part coincides with the whole page.

One of the two parts is deemed to contain valid virtual addresses. Addresses in the remaining part are declared invalid. Any attempt to reference an invalid address will be trapped by the hardware. The advantage of this scheme is that space in the physical memory need only be allocated for the valid r)art of a page.

Page Description Register

The page description register defines:

(a) the size of the lower part of the page. (The number stored is actually the number of 32 word blocks less one);

(b) a bit which is set when the upper part is the valid part. (Also known as the "expansion direction" bit);

(c) access mode bits defining "no access" or "read only access" or "read/write access".

Note that if the valid part is null, this fact must be shown by setting the access bits to "no access".

Memory Allocation

The hardware does not dictate the way areas in physical memory which correspond to the valid parts of pages should be allocated (except to the extent that they must begin and end on a 32 word boundary). These areas may be allocated in any order and may overlap to any extent.

In practice the allocation of areas of physical memory is much more disciplined as we shall see in Chapter Seven. Areas for pages which are related are most often allocated contiguously and in the order of their page numbers, so that all the segment areas associated with a single program are contained within one or at most two large areas of physical memory.

Memory Management Status Registers

In addition to the segmentation registers, on the PDP11/40 there are two memory management status registers:

SR0 contains abort error flags and other essential information for the operating system. In particular memory management is enabled when bit 0 of SR0 is on;

SR2 is loaded with the 16 bit virtual address at the beginning of each instruction fetch.

"i" and "d" Spaces

In the PDP11/45 and 11/70 systems, there are additional sets of segmentation registers. Addresses created using the pc register (r7) are said to belong to "i" space, and are translated by a different set of segmentation registers from those used for the remaining addresses which are said to belong to "d" space.

The advantage of this arrangement is that both "i" and "d" spaces may occupy up to 32K words, thus allowing the maximum space which can be allocated to a program to be increased to twice the space available on the PDP11/40.

Initial Conditions

When the system is first started after all the devices on the Unibus have been reinitialised, the memory management unit is disabled and the processor is in kernel mode.

Under these circumstances, virtual (byte) addresses in the range 0 to 56K are mapped into identically valued physical addresses. However the highest page of the virtual address space is mapped into the highest page of the physical address space, i.e. on the PDP11/40 or 11/45, addresses in the range

```
0160000 to 0177777
```

are mapped into the range

```
0760000 to 0777777
```

Special Device Registers

The high page of physical memory is reserved for various special registers associated with the processor and the peripheral devices. By sacrificing one page of memory space in this way, the PDP11 designers have been able to make the various device registers accessible without the need to provide special instruction types.

The method of assignment of addresses to registers in this page is a black art: the values are hallowed by tradition and are not to be questioned.

Chapter three

Reading "C" Programs

Learning to read programs written in the "C" language is one of the hurdles that must be overcome before you will be able to study the source code of UNIX effectively.

As with natural languages, reading is an easier skill to acquire than writing. Even so you will need to be careful lest some of the more subtle points pass you by.

There are two of the "UNIX Documents" which relate directly to the "C" language:

"C Reference Manual", by Dennis Ritchie

"Programming in C -- A Tutorial",
by Brian Kernighan

You should read them now, as far as you can, and return to reread them from time to time with increasing comprehension.

Learning to write "C" programs is not required. However if you have the opportunity, you should attempt to write at least a few small programs. This does represent the accepted way to learn a programming language, and your understanding of the proper use of such items as:

semicolons;
"=" and "=="
"" and ""
"++" and "--"
declarations;
register variables;
"if" and "for" statements
etc.

will be quickly reinforced.

You will find that “C” is a very convenient language for accessing and manipulating data structures and character strings, which is what a large part of operating systems is about. As befits a terminal oriented language, which requires concise, compact expression, “C” uses a large character set and makes many symbols such as “*” and “&” work hard. In this respect it invites comparison with APL.

There many features of “C” which are reminiscent of PL/1, but it goes well beyond the latter in the range of facilities provided for structured programming.

Some Selected Examples

The examples which follow are taken directly from the source code.

Example 1

The simplest possible procedure, which does nothing, occurs twice(!) in the source code as “nullsys” (2864) and “nulldev” (6577), sic.

```
6577 nulldev ()
     {
     }
```

While there are no parameters, the parentheses, “(” and “)”, are still required. The brackets “” and “” delimit the procedure body, which is empty.

Example 2

The next example is a little less trivial:

```
6566 nodev ()
     {
      u.u_error = ENODEV;
     }
```

The additional statement is an assignment statement. It is terminated by a semicolon which is part of the statement, not a statement separator as in Algol-like languages.

“ENODEV” is a defined symbol, i.e. a symbol which is replaced by an associated character string by the compiler preprocessor before actual compilation. “ENODEV” is defined on line 0484 as 19. The UNIX convention is that defined symbols are written in upper case, and all other symbols in lower case.

“=” is the assignment operator, and “u.u_error” is an element of the structure “u”. (See line 0419.) Note the use of “.” as the operator which selects an element of a structure. The element name is “u_error” which may be taken as a paradigm for the way names of structure elements are constructed in the UNIX source code: a distinguishing letter is followed by an underscore followed by a name.

Example 3

```
6585 bcopy (from, to, count)
     int *from, *to;
     {
      register *a, *b, c;
      a = from;
      b = to;
      c = count;
      do
        *b++ = *a++;
      while (--cc);
     }
```

The function of this procedure is very simple: it copies a specified number of words from one set of consecutive locations to another set.

There are three parameters. The second line

```
int *from, *to;
```

specifies that the first two variables are pointers to integers. Since no specification is supplied for the third parameter, it is assumed to be an integer by default.

The three local variables, a, b, and c, have been assigned to registers, because registers are more accessible and the object code to reference them is shorter. “a” and “b” are pointers to integers and “c” is an integer. The register declaration could have been written more pedantically as

```
register int *a, *b, c;
```

to emphasise the connection with integers.

The three lines beginning with “do” should be studied carefully. If “b” is a “pointer to integer” type, then

```
*b
```

denotes the integer pointed to. Thus to copy the value pointed to by “a” to the location designated by “b”, we could write

```
*b = *a;
```

If we wrote instead

```
b = a;
```

this would make the value of “b” the same as the value of “a”, i.e. “b” and “a” would point to the same place. Here at least, that is not what is required.

Having copied the first word from source to destination, we need to increase the values of “b” and “a” so that the point to the next words of their respective sets. This can be done by writing

```
b = b+1; a = a+1;
```

but “C” provides a shorter notation (which is more useful when the variable names are longer) viz.

```
b++; a++;
```

or alternatively

```
++b; ++b;
```

Now there is no difference between the statements “b++;” and “++b;” here.

However “b++” and “++b” may be used as terms in an expression, in which case they are different. In both cases the effect of incrementing “b” is retained, but the value which enters the expression is the initial value for “b++” and the final value for “++b”.

The “--” operator obeys the same rules as the “++” operator, except that it decrements by one. Thus “--c” enters an expression as the value after decrementation.

The “++” and “--” operators are very useful, and are used throughout UNIX. Occasionally you will have to go back to first principles to work out exactly what their use implies. Note also there is a difference between

```
*b++ and (*b)++.
```

These operators are applicable to pointers to structures as well as to simple data types. When a pointer which has been declared with reference to a particular type of structure is incremented, the actual value of the pointer is incremented by the size of the structure.

We can now see the meaning of the line

```
*b++ = *a++;
```

The word is copied and the pointers are incremented, all in one hit.

The line

```
while (--c);
```

delimits the end of the set of statements which began after the “do”. The expression in parentheses “--c”, is evaluated and tested (the value tested is the value after decrementation). If the value is non-zero, the loop is repeated, else it is terminated.

Obviously if the initial value for “count” were negative, the loop would not terminate properly. If this were a serious possibility then the routine would have to be modified.

Example 4

```
6619 getf (f)
     {
       register *fp, rf;
       rf = f;
       if (rf < 0 || rf >= NOFILE)
         goto bad;
       fp = u.u_ofile[rf];
       if (fp != NULL)
       return (fp);
     bad:
       u.u_error = EBADF;
       return (NULL);
     }
```

The parameter “f” is a presumed integer, and is copied directly into the register variable “rf”. (This pattern will become so familiar that we will now cease to remark upon it.)

The three simple relational expressions

```
rf < 0    rf >=NOFILE    fp != NULL
```

are each accorded the value one if true, and the value zero if false. The first tests if the value of “rf” is less than zero, the second, if “rf” is greater than the value defined by “NOFILE” and the third, if the value of “fp” is not equal to “NULL” (which is defined to be zero).

The conditions tested by the “if” statements are the arithmetic expressions contained within parentheses.

If the expression is greater than zero the test is successful and the following statement is executed. Thus if for instance, “fp” had the value 001375, then

```
fp != NULL
```

is true, and as a term in an arithmetic expression, is accorded the value one. This value is greater than zero, and hence the statement

```
return(fp);
```

would be executed, to terminate further execution of “getf”, and to return the value of “fp” to the calling procedure as the result of “getf”.

The expression

```
rf < 0 || rf >= NOFILE
```

is the logical disjunction (“or”) of the two simple relational expressions.

An example of a “goto” statement and associated label will be noted.

“fp” is assigned a value, which is an *address*, from the “rf”-th element of the array of integers “u_ofile”, which is embedded in the structure “u”.

The procedure “getf” returns a value to its calling procedure. This is either the vale of “fp” (i.e. an address) or “NULL”.

Example 5

```
2113 wakeup (chan)
     {
       register struct proc *p;
       register c, i;
       c= chan;
       p=&proc[0];
       i= NPROC;
       do {
           if (p->p_wchan == c) {
             setrun(p);
           }
           p++;
       } while (--i);
     }
```

There are a number of similarities between this example and the previous one. We have a new concept however, an array of structures. To be just a little confusing, in this example it turns out that both the array and the structure are called “proc” (yes, “C” allows this). They are declared on Sheet 03 in the following form:

```
0358 struct proc
     {
       char p_stat;
       ..........
       int p_wchan;
       ..........
     } proc[NPROC];
```

“p” is a register variable of type pointer to a structure of type “proc”.

```
p =&proc[0];
```

assigns to “p” the address of the first element of the array “proc”. The operator “&” in this context means “the address of”.

Note that if an array has n elements, the elements have subscripts 0, 1, .., (n-1). Also it is permissible to write the above statement more simply as

```
p = proc;
```

There are two statements in between the “do” and the “while”.

The first of these could be rewritten more simply as

```
if (p->p wchan == c) setrun (p);
```

i.e. the brackets are superfluous in this case, and since “C” is a free form language, the arrangement of text between lines is not significant.

The statement

```
setrun (p);
```

invokes the procedure “setrun” passing the value of “p” as a parameter (All parameters are passed by value.).

The relation

```
p->p_wchan == c
```

tests the equality of the value of “c” and the value of the element “p_wchan” of the structure pointed to by “p”. Note that it would have been wrong to have written

```
p.p_wchan == c
```

because “p” is not the *name* of a structure.

The second statement, which cannot be combined with the first, increments “p” by the size of the “proc” structure, whatever that is. (The compiler can figure it out.)

In order to do this calculation correctly, the compiler needs to know the kind of structure pointed at. When this is not a consideration, you will notice that often in similar situations, “p” will be declared simply as

```
register *p;
```

because it was easier for the programmer, and the compiler does not insist.

The latter part of this procedure could have been written equivalently but less efficiently as

```
...........
i = 0;
```

```
do
  if (proc[i].p_wchan == c)
    setrun (&proc[i]);
while (++i < NPROC);
```

Example 6

```
5336 geterror (abp)
     struct buf *abp;
     {
       register struct buf bp;
       bp = abp;
       if (bp->b flags&B_ERROR)
         if ((u.u_error=bp->b_error)==0)
           u.u_error = EIO;
     }
```

This procedure simply checks if there has been an error, and if the error indicator “u.u_error” has not been set, sets it to a general error indication

“B_ERROR” has the value 04 (see line 4575) so that, with only one bit set, it can be used as mask to isolate bit number 2. The operator “&” as used in

```
bp->b_flags&B_ERROR
```

is the bitwise logical conjunction (“and”) applied to arithmetic values.

The above expression is greater than one if bit 2 of the element “b_flags” of the “buf” structure pointed to by “bp”, is set.

Thus if there has been an error, the expression

```
(u.u_error) = bp->b_error)
```

is evaluated and compared with zero. Now this expression includes an assignment operator “=”. The value of the expression is the value of “u.u_error” *after* the value of “bp->b_flags” has been assigned to it.

This use of an assignment as part of an expression is useful and quite common.

Example 7

```
3428 stime ()
     {
       if (suser()) {
         time[0] = u.u_ar0[R0];
         time[1] = u.u_ar0[R1];
         wakeup (tout);
       }
     }
```

In this example, you should note that the procedure “suser” returns a value which is used for the “if” test. The three statements whose execution depends on this value are enclosed in the brackets “” and “”.

Note that a call on a procedure with no parameters must still be written-with a set of empty parentheses, sic.

```
suser ()
```

Example 8

“C” provides a conditional expression. Thus if “a” and “b” are integer variables,

```
(a > b ? a : b)
```

is an expression whose value is that of the larger of “a” and “b”.

However this does not work if “a” and “b” are to be regarded as unsigned integers. Hence there is a use for the procedure

```
6326 max (a, b)
     char *a, *b;
     {
       if (a > b)
         return(a);
       return(b);
     }
```

The trick here is that “a” and “b”, having been declared as pointers to characters are treated for comparison purposes as unsigned integers.

The body of the procedure could have been written as

```
max (a, b)
char *a, *b;
{
  if (a > b)
    return(a);
  else
    return(b);
}
```

but the nature of “return” is such that the “else” is not needed here!

Example 9

Here are two quickies which introduce some different and exotic looking expressions. First:

```
7679 schar()
     {
       return *u.u_dirp++&0377);
     }
```

where the declaration

```
char *u_dirp;
```

is part of the declaration of the structure “u”.

“u.u_dirp” is a character pointer. Therefore the value of “*u.u_dirp++” is a character. (Incrementation of the pointer occurs as a side effect.)

When a character is loaded into a sixteen bit register, sign extension may occur. By “and”ing the word with 0377 any extraneous high order bits are eliminated. Thus the result returned is simply a character.

Note that any integer which begins with a zero (e.g. 0377) is interpreted as an octal integer.

The second example is:

```
1771 nseg(n)
     {
       return ((n+127)>>7);
     }
```

The value returned is n divided by 128 and rounded up to the next highest “integer”.

Note the use of the right shift operator “>>” in preference to the division operator “/”.

Example 10

Many of the points which have been introduced above are collected in the following procedure:

```
2134 setrun (p)
     {
       register struct proc *rp;
       rp = p;
       rp->p_wchan = 0;
       rp->p_stat = SRUN;
       if (rp->p_pri < curpri)
         runrun++;
       if (runout != 0&&
           (rp->p_flag&SLOAD) == 0) {
          runout = 0;
          wakeup (&runout);
       }
     }
```

Check your understanding of “C” by figuring out what this one does.

There are two additional features you may need to know about:

“&&” is the logical conjunction (“and”) for relational expressions. (Cf. “+||+” introduced earlier.)

The last statement contains the expression

```
&runout
```

which is syntactically an address variable but semantically just a unique bit pattern.

This is an example of a device which is used throughout UNIX. The programmer needed a unique bit pattern for a particular purpose. The exact value did not matter as long as it was unique. An adequate solution to the problem was to use the address of a suitable global variable.

Example 11

```
4856 bawrite (bp)
     struct buf *bp;
     {
       register struct buf *rbp;
       rbp = bp;
       rbp->b_flags =| B_ASYNC;
       bwrite (rbp);
     }
```

The second last statement is interesting because it could have been written as

```
rbp->b_flags = rbp->b_flags | B_ASYNC;
```

In this statement the bit mask “B ASYNC” is “or”ed into “rbp->b_flags”. The symbol “|” is the logical disjunction for arithmetic values.

This is an example of a very useful construction in UNIX, which can save the programmer much labour. If “⊕” is any binary operator, then

```
x = x ⊕ a;
```

where “a” is an expression, can be rewritten more succinctly as

```
x =⊕ a;
```

A programmer using this construction has to be careful about the placement of blank characters, since

```
x =+ 1;
```

is different from

```
x = +1;
```

What is to be the meaning of

```
x =+1;        ?
```

Example 12

```
6824 ufalloc ()
     {
       register i;
       for (i=0; i<NOFILE; i++)
         if (u.u_ofile[i]==NULL) {
           u.u_ar0[R0] = i;
           return (i);
         }
       u.u_error = EMFILE;
       return (-1);
     }
```

This example introduces the "for" statement, which has a very general syntax making it both powerful and compact.

The structure of the "for" statement is adequately described on page 10 of the "C Tutorial", and that description is not repeated here.

The Algol equivalent of the above "for" statement would be

```
for i:=1 step 1 until NOFILE-1 do
```

The power of the "for" statement in "C" derives from the great freedom the programmer has in choosing what to include between the parentheses. Certainly there is nothing which restricts the calculations to integers, as the next example will demonstrate.

Example 13

```
3949 signal (tp, sig)
     {
       register struct proc *p;
       for (p=proc;p<&proc[NPROC];p++)
         if (p->p ttyp == tp)
           psignal (p,sig);
     }
```

In this example of the "for" statement, the pointer variable "p" is stepped through each element of the array "proc" in turn.

Actually the original code had

```
for (p=&proc[0];p<&proc[NPROC];p++)
```

but it wouldn't fit on the line! As noted earlier, the use of "proc" as an alternative to the expression "&proc[0]" is acceptable in this context.

This kind of "for" statement is almost a cliche in UNIX so you had better learn to recognise it. Read it as

> *for p = each process in turn*

Note that "proc[NPROC]" is the address of the (NPROC+1)-th element of the array (which does not of course exist) i.e. it is the first location beyond the end of the array.

At the risk of overkill we would point out again that whereas in the previous example

```
i++;
```

meant add one to the integer "i", here

```
p++;
```

means "skip p to point to the next structure".

Example 14

```
8870 lpwrite ()
     {
      register int c;
      while ((c=cpass()) >= 0)
        lpcanon(c);
     }
```

This is an example of the "while" statement, which should be compared with the "do ... while ..." construction encountered earlier. (Cf. the "while" and "repeat" statements of Pascal.)

The meaning of the procedure is

> *Keep calling "cpass" while the result is positive, and pass the result as a parameter to a call on lpcanon.*

Note the redundant "int" in the declaration for "c". It isn't always omitted!

Example 15

The next example is abbreviated from the original:

```
5861 seek ()
     {
       int n[2];
       register *fp, t;
       fp = getf (u.u_ar0[R0]);
       ..........
       t = u.u_arg[1];
       switch (t) {
       case 1:
       case 4:
         n[0] =+ fp->f_offset[0];
         dpadd (n, fp->f_offset[1]);
         break;
       default:
         n[0] =+ fp->f_inode->i size0&0377;
         dpadd(n,fp->f_inode->i_size1);
       case 0:
       case 3:
         ;
```

```
    }
    ...........
}
```

Note the array declaration for the two word array "n", and the use of getf (which appeared in Example 4).

The "switch" statement makes a multiway branch depending on the value of the expression in parentheses. The individual parts have "case labels":

- If "t" is one or four, then one set of actions is in order.
- If "t" is zero or three, nothing is to be done at all.
- If "t" is anything else, then a set of actions labelled "default" is to be executed.

Note the use of "break" as an escape to the next statement after the end of the "switch" statement. Without the "break", the normal execution sequence would be followed within the "switch" statement.

Thus a "break" would normally be required at the end of the "default" actions. It has been omitted safely here because the only remaining cases actually have null actions associated with them.

The two non-trivial pairs of actions represent the addition of one 32 bit integer to another. The later versions of the "C" compiler will support "long" variables and make this sort of code much easier to write (and read).

Note also that in the expression

```
fp->f_inode->i_size0
```

there are two levels of indirection.

Example 16

```
6672 closei (ip, rw)
     int *ip;
     {
       register *rip;
       register dev, maj;
       rip = ip;
       dev = rip->i_addr[0];
       maj = rip->i_addr[0].d major;
       switch (rip->i_mode&IFMT) {
       case IFCHR:
         (*cdevsw[maj].d_close)(dev,rw);
         break;
       case IFBLK:
         (*bdevsw[maj].d_close)(dev,rw);
       }
       iput(rip);
     }
```

This example has a number of interesting features.

The declaration for "d_major" is

```
struct {
   char d_minor;
   char d_major;
}
```

so that the value assigned to "maj" is the hiqh order byte of the value assigned to "dev".

In this example, the "switch" statement has onlv two non-null cases, and no "default". The actions for the recognised cases, e.g.

```
(*bdevsw[maj].d_close)(dev,rw);
```

look formidable at first glance.

First it should be noted that this is a procedure call, with parameters "dev" and "rw".

Second "bdevsw" (and "cdevsw") are arrays of structures, whose "d_close" element is a pointer to a function, i.e.

```
bdevsw[maj]
```

is the name of a structure, and

```
bdevsw[maj].d_close
```

is an element of that structure which happens to be a pointer to a function, so that

```
*bdevsw[maj].d_close
```

is the name of a function. The first pair of parentheses is "syntactical sugar" to put the compiler in the right frame of mind!

Example 17

We offer the following as a final example:

```
4043 psig ()
     {
       register n, p;
       .........
       switch (n) {
       case SIGQIT:
       case SIGINS:
       case SIGTRC:
       case SIGIOT:
       case SIGEMT:
       case SIGEPT:
       case SIGBUS:
       case SIGSEG:
       case SIGSYS:
           u.u arg[0] = n;
           if (core())
             n =+ 0200;
       }
```

```
        u.u_arg[0]=(u.u_ar0[R0]<<8) | n;
        exit ();
    }
```

Here the “switch” selects certain values for “n” for which the one set of actions should be carried out.

An alternative would have been to write a “monster” “if” statement such as

```
    if (n==SIGQUIT || n==SIGINS || ...
               ... || n==SIGSYS)
```

but that would not have been either transparent or efficient.

Note the addition of an octal constant to “n” and the method of composing a 16 bit value from two eight bit values.

This issue’s CD

Greg Lehey <Greg.Lehey@auug.org.au>

This quarter we include CD 1 of the latest update to the Debian GNU/Linux stable release, 3.0, also known as *woody.* Here’s an excerpt from the original release announcement, dated 19 July 2002 (*http://www.debian.org/News/2002/20020719*):

> ## Debian GNU/Linux 3.0 released
>
> July 19th, 2002
>
> The Debian Project is pleased to announce the release of Debian GNU/Linux version 3.0. Debian GNU/Linux is a free operating system, which now supports a total of eleven processor architectures, includes KDE and GNOME desktop environments, features cryptographic software, is compatible with the FHS v2.2 and supports software developed for the LSB.
>
> With the addition of the IA-64 (ia64), HP PA-RISC (hppa), MIPS (mips, mipsel), and S/390 (s390) architectures, Debian GNU/Linux now supports a total of eleven architectures. It now runs on computers ranging from palmtops to supercomputers, and nearly everything in between, including the latest generation of 64 bit machines.
>
> This is the first version of Debian that features cryptographic software integrated into the main distribution. OpenSSH and GNU Privacy Guard are included in the default installation, and strong encryption is now present in web browsers and web servers, databases, and so forth. Further integration of cryptographic software is planned for future releases.
>
> For the first time, Debian comes with the K Desktop Environment 2.2 (KDE). The GNOME desktop environment is upgraded to version 1.4, and X itself is upgraded to the much improved XFree86 4.1. With the addition of several full-featured free graphical web browsers in the form of Mozilla, Galeon, and Konqueror, Debian’s desktop offerings have radically improved.
>
> This version of Debian supports the 2.2 and 2.4 releases of the Linux kernel. Along with better support for a greater variety of new hardware (such as USB) and significant improvements in usability and stability, the 2.4 kernel provides support for the ext3 and reiserfs journaling filesystems.
>
> Debian GNU/Linux 3.0 features a more streamlined and polished installation, which is translated into numerous languages. The task system has been revamped and made more flexible. The debconf tool makes configuration of the system easier and more user friendly. Debian GNU/Linux can be installed from CD, or from the network and a few floppies. It can be downloaded now, and will soon be available on CD-ROM from numerous vendors.
>
> Upgrades to Debian GNU/Linux 3.0 from earlier releases are automatically handled by the apt package management tool. As always, Debian GNU/Linux systems can be upgraded painlessly, in place, without any forced downtime. For detailed instructions about installing and upgrading Debian GNU/Linux, please see the release notes.
>
> This is the first release of Debian that is compatible with version 2.2 of the Filesystem Hierarchy Standard (FHS). Debian GNU/Linux now also supports software developed for the Linux Standard Base (LSB), though it is not yet LSB certified.
>
> Current Debian users may be interested to know that this release of Debian supports build dependencies, to aid in building packages from source, and apt pinning, to ease partial upgrades to our testing or unstable branch. This release of Debian features aptitude as an alternative for the venerable dselect program, which will make it easier to select packages. About four thousand new software packages were added to the distribution in Debian GNU/Linux 3.0.

The version on the CD is the third update, released on 26 October 2004. Here’s an except from *that* announcement (*http://www.debian.org/News/2004/20041026*):

Debian GNU/Linux 3.0 updated (r3)

October 26th, 2004

This is the third update of Debian GNU/Linux 3.0 (codename "woody") which mainly adds security updates to the stable release, along with a few corrections to serious problems. Those who frequently update from security.debian.org won't have to update many packages and most updates from security.debian.org are included in this update.

Please note that this update does not produce a new version of Debian GNU/Linux 3.0 but only adds a few updated packages to it. There is no need to throw away 3.0 CDs but only to update against ftp.debian.org after an installation, in order to incorporate those late changes.

Upgrading to this revision online is usually done by pointing the *apt* package tool (see the sources.list(5) manual page) to one of Debian's many FTP or HTTP mirrors. A comprehensive list of mirrors is available at: *http://www.debian.org/distrib/ftplist*.

The future of AUUGN CDs

Greg Lehey <Greg.Lehey@auug.org.au>

This article contains important information about your AUUGN subscription. Please read it.

For the past couple of years we have included a CD-R of some description for every issue of AUUGN. In general they have been very well received. Nevertheless it won't have escaped your attention that it hasn't always been plain sailing. We've had a number of issues:

- Timing issues. AUUGN has had significant issues being on time in the last year or two. Hopefully they're now over, but one of the biggest hold-ups has been CD-R production. In addition, a year ago we had problems because the CD-Rs intended for AUUGN were taken at the annual conference, leaving not enough for AUUGN.
- CD-R quality issues. These have been particularly bad in the past twelve months. Earlier this year we managed to send out CD-Rs that were illegible.
- CD-R capacity issues. Our budget is sufficient for one CD-R per issue. We can occasionally go to two, but that's about the limit.

  Unfortunately, there's not too much that will fit on a single CD-R any more. A few months ago we had to drop support for the Microsoft version of OpenOffice because there wasn't space on the disk. We also found ourselves out of pocket with Fedora Core II because it turned out that our original intention of distributing only CD 1 was a non-starter, so we had to include CD 2. Even so, some people might have had problems because we didn't include CD 3. This quarter we're distributing CD 1 (only) of Debian Woody upgrade 3. That will work, but for anything serious you'll need to get software from the net.

  This problem is exacerbated by our decision to move to publishing AUUGN in electronic form. That requires extra space. It's only a couple of megabytes, but that's often note available: release engineers shoe-horn everything they can into an ISO, are they're often full to bursting point.

We're addressing the quality issues: starting with this issue, we have a new CD duplicator. Hopefully they will deliver consistently good quality. It should also address some of the timing problems we've been having. If you *do* have problems, tell let Liz Carroll, our executive director, immediately. See page 2 for contact details.

The other issues are interrelated. Now that we don't have the expense of the paper edition of AUUGN, we can afford to distribute a DVD instead of a CD; the price is less than the price of two CDs. This gives us plenty of space, enough for a complete distribution of most operating systems, and we should still have enough for AUUGN as well.

Everybody has a DVD reader, right? That's our assumption. If you have a problem with the transition, please let Liz know now: starting next year, all you'll get will probably be a DVD.

A Hacker's Diary

Greg Lehey <greg.lehey@auug.org.au>

This article contains the more technically interesting parts of my online diary, which you can find at *http://www.lemis.com/grog/diary.html.* There is no reason to believe that it has any specific relationship to the views of other AUUG members.

The original diary contains a number of URLs, which makes sense for an online diary. The URLs are mainly irritating in this medium, and they cause big typesetting problems for the dual-column format, so I've removed most of them. To minimize the ugliness of the format, many of those that I've kept are given as relative to the directory in which the diary is kept, *http://www.lemis.com/grog/.*

Several entries in this diary refer to "my program". That's what I'm doing in real life; it's not appropriate to say what "my program" really is.

Thursday, 1 July 2004

As I've said many times before, setting up a new machine is just too difficult. At least I now have the feeling that I've found a better approach than before: basically, keep all mods to the system and ports configuration under RCS on a separate file system. Then, along with a list of the *real* ports I've installed (not including the myriad dependent ports), I should be able to create a complete new system almost automatically. At least now I have time to debug the procedure.

Today it wasn't helped, however, by a couple of problems. It seems that FreeBSD 5.2.1 runs really badly on the new MSI K7N2 Delta motherboard but 5-CURRENT runs well, so ended up doing the installation with the system disk from *beeble.* This is in fact probably the best way to do it, rather than installing from CD-ROM. In this case, it was somewhat confused by the fact that somebody had broken world, and I had to re-*cvsup* to get a consistent source tree.

All that was in the background; the good news is that I can get back to some kind of work on my program.

Saturday, 3 July 2004

Back to working on AUUGN today, in the process debugging some macros and working out an automatic way to build the *download* file that *groff* uses to decide which fonts to include in a PostScript document. In FreeBSD, the file is called */usr/share/groff_font/devps/download*, and it contains a lookup from font name to the file which contains the description:

```
# List of downloadable fonts
# PostScript-name   Filename
Symbol-Slanted              symbolsl.pfa
ZapfDingbats-Reverse        zapfdr.pfa
```

The second entry on each line is the name of a file in the same directory. They start with a line like: `%!PS-AdobeFont-1.0: Garamond-Bold 001.003`, which gives the information needed for *download.* The following single line script adds to it:

```
grep AdobeFont *.pfa | sed 's/:.*://' | \
   awk '{print $2 "" $1}' >> download
```

Apart from that, more work on the automatic system upgrade. Some of the tools are not really laid out for this kind of work: the `install` target in */usr/src/etc/* does nothing more than to build a couple of *whatis* files. The real install target is called `distribution`, and if you invoke it without any special options, it overwrites your */etc* file system, provided that all the subdirectories it expects are present: there's a target `distrib-dirs` to create them, but `distribution` doesn't (yet) depend on it. Played around with that for a while and was able to install */etc* on my virgin disk.

Also working on how to build ports, somewhat confused by the fact that some of them don't appear to work correctly with `DESTDIR` set:

```
/mnt/usr/local/man/man1/s2p.1
/usr/bin/strip: /mnt/usr/local/bin/perl5.8.4:
   No such file or directory
```

I'm also still running into various problems with the new motherboard, so since it's apparently as good as identical with the motherboard in *wantadilla*, which has been running reliably for months, decided to swap the systems around. That was more work than I thought: apart from random issues like a monitor cable disconnecting itself from the monitor, old *wantadilla* didn't boot properly with the new *beeble* system: I got watchdog timeouts on both Realtek and a 3Com Ethernet boards, and they didn't go away until I disabled the APIC. Clearly we have problems with recent -CURRENT.

Sunday, 4 July 2004

In March two of our video recorders died in rapid succession, and we sent them down to an el-cheapo place to have the symptoms (tape eating) fixed. This turned out to be a bad idea; yes, it didn't cost much, but one still had such bad tracking that we can only use it as a tuner for the TiVo and the other one died again with the same symptoms this evening. It doesn't seem to be worth replacing it. We bought a special offer video recorder from Coles last month for only $129, with 6 head technology and NTSC playback. The firmware's a bit baroque, but we'll survive until I finally get my computer-based system together, so we'll look for another one.

Monday, 5 July 2004

The new motherboard *does* seem to be defective. Now it's in the old *wantadilla*, which has been stable for months, but today I had multiple problems: first *Netscape* crashed multiple times, not in itself such an uncommon occurrence, but it hasn't happened for a while; later I had two spontaneous system resets. How I hate flaky hardware!

Tuesday, 6 July 2004

Another hardware day. Finally decided that the new motherboard was flaky enough to send back, even though I couldn't demonstrate obvious problems. Also sent back the motherboard and disk from *echunga* that died last week Of course they couldn't replace the motherboard imemediately—it's 9 months old and thus obsolete—so I'll have to wait a couple of weeks for a replacement.

Yvonne came back with the new hardware and also a Samsung cordless mouse. She also brought back another video recorder, the same as the one she bought from Coles last month. The last one cost $129; this one cost $99, probably less than the cost of repairing the old failed one. Didn't have the time (or necessity) to install it today.

More interesting than either was my new laptop, a Dell Inspiron 9100 that had been left me by somebody else in the company. In general, hand-me-downs are less than successful, but in this case it was almost exactly what I want, especially since it was a top-end unit. Somehow it's typical, though, that the one thing that interests me—the display resolution—wasn't top-end: only 1680x1050, barely more than 4-year-old *sydney*. I suspect that the real issue is that Microsoft platforms can't use higher resolution.

Put the new motherboard into new *wantadilla*, and no longer had the stability problems I had seen before. Instead, I got an interrupt flood on the Ethernet cards. I had seen this before on the previous motherboard, too, but there I could get rid of it by disabling the APIC. This time, disabling the APIC didn't help. I had to reallocate the IRQ to get rid of the problem. Why does each motherboard have its own problems? And yes, of course I set the BIOS settings to the same values.

Still, the system upgrade procedure is working well. I think I've made enough progress now to start again from scratch and confirm that it works.

Wednesday, 7 July 2004

How long these things take! The admin work, which I had hoped to have out of the way this time last week, is still with me. To get the system update thing working properly, decided to start again, this time with the new 80 GB disk that I had got yesterday in exchange for the defective */src*. There's enough space on that to have two system partitions of 10 GB each and a separate */home* partition. That way future upgrades use the alternate system partition, and you don't have to worry about changing hardware around: just boot from the other partition. If it doesn't work as expected, going back to the old version is as simple as rebooting from the other partition.

Also accepted the fact that I had to do something about my mail. My inbox was hitting the 10,000 message mark, and more importantly coming towards the magic 48 MB after which *postfix* stops saving messages. I thought I could get rid of 5,000 messages relatively quickly. Maybe I did: interleaved with other work, it took me 4 hours, or about 4 seconds per message. Still too long. I should unsubscribe from some lists.

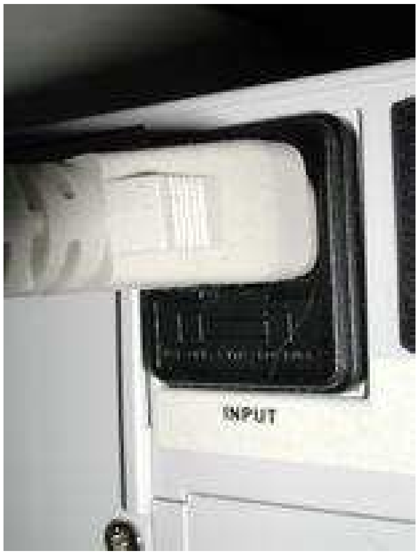
Then turned my attention to the new Inspiron 9100 which runs Microsoft "XP Professional". Getting that to run was quite an experience. Though the machine had already installed all the software, I had to add myself as another user, which required reinstalling the software. "Outlook 2003" required remote product registration, either via the Internet or by phone. I have no intention of taking the risk of connecting any Microsoft machine to the Internet, so I chose the phone method. Type in 42 digits, and get another 42 digits back. You'd think that people would make more of the advantage of Open Source software that you don't have to do this kind of nonsense. It has nothing to do with the cost of the software, just the time and inconvenience it takes to install it.

Thursday, 8 July 2004

Still more admin work today, and got to understand the Microsoft box better; it's running a version called "XP Professional", and although it's pretty much as painful as other versions of Microsoft, it has one good feature: I can access the "desktop" from another machine with the *rdesktop* program. *rdesktop* has some strangenesses, and I couldn't get it to stop reversing the mouse buttons, but it means that I have almost a real keyboard and a reasonable sized display to work with.

Also continued work on the upgrade to *wantadilla*—the machine paniced once in `bzero`, which doesn't make me very happy. Maybe it's the memory after all. I'll have to keep an eye on it.

In the evening, the Digitrex DVD recorder hung itself up again. It's been less frequent, but it's still there. The only way to reset it is to pull the power plug and reinsert it: if ever a device needed a power switch, it's this one. This time it caused enough of a surge to blow the residual current sensor, quite a feat for a device without a ground pin. When I turned it back on again, the UPS continued to beep, although it had power. Took the thing apart ("no user-serviceable parts inside") and found, well, no user-serviceable parts. After some investigation discovered that the power connector itself included a well hidden fuse:



In case that's still not obvious, it's the black area between the connector and the inscription **INPUT**. It even contained a replacement fuse, quite a nice idea. Now if only that had been more obvious from the outset.

Friday, 9 July 2004

Finally things are almost back to normal again. I'm continuing my upgrade of *wantadilla* in the background, and I managed to sort out my problems with the Microsoft box as well: it seems that *rdesktop* problems were due to a strange feature of "XP Professional" that turns the mouse buttons round for USB mice only (well, also for *rdesktop*, it would seem). I wonder why anybody would want his buttons reversed only for specific mice. Also managed to install a newer license manager for the Oxford English Dictionary which actually looks a lot better on *rdesktop* than it ever did on the machine itself.

Back to work on my program, and I'm roughly where I was two weeks ago, before I decided to rewrite the endian layer, and before I had the hardware problems. The difference is that things look so much clearer now.

Having a Microsoft box on the network means opening yourself to many more vulnerabilities, and I've been working on tightening up my firewall. The output of *'iptables*-vL' shows that in the course of 24 hours I have bounced in the order of 12 MB of traffic—not only unwanted, but indiscriminate.

> The online version of the diary includes example output, but doesn't fit in the magazine format.

That's about a third of my total traffic. It's about time that this kind of abuse were made illegal, world-wide. I suppose it will be one day, but first the legislators need to understand what it's all about. And yes, probably most of the traffic comes from innocently affected Microsoft boxes. That's not an excuse, any more than driving an unroadworthy car is an excuse for having an accident.

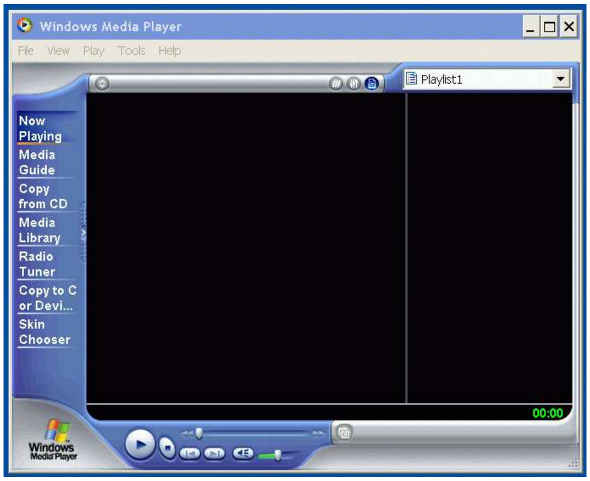
Saturday, 10 July 2004

Back to working on AUUGN today, and again made relatively good progress. If this continues, I may be able to continue to edit it. But it would be a whole lot better if I could get more people to participate.

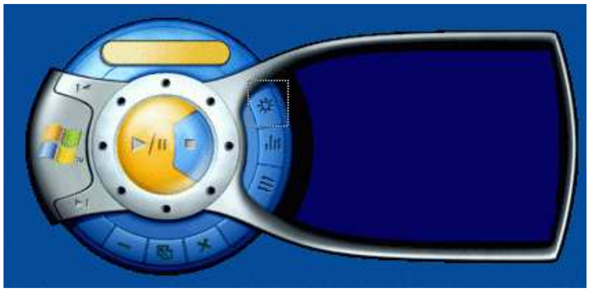
In the afternoon, Gareth Andrews came around with a computer on which he wanted to run my temperature control software (*brewing/temperature-control.html*). Installed FreeBSD 4.10 on it (and, as I later discovered, left the CD in the drive when he took it home), and then the temperature control software. That was worthwhile: I found numerous buglets which needed correcting. Still, all told, including installing FreeBSD, the hardware and the temperature control software, not to mention a bit of beer tasting, we were done in three hours. Gareth knows nothing about FreeBSD, though, and I fear he might have a bit of difficulty coming to terms with different concepts.

Sunday, 11 July 2004

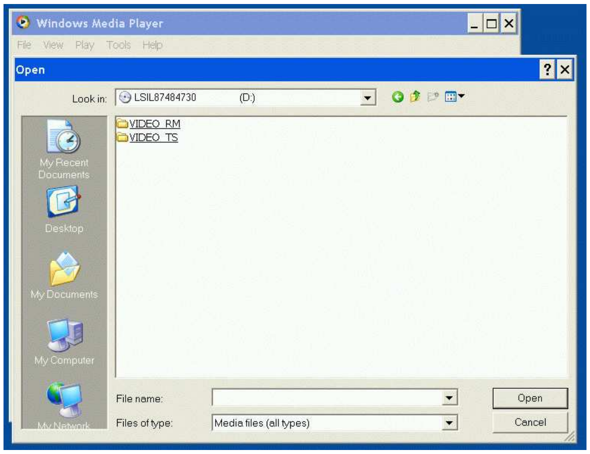
Somehow another day where nothing much happened. Chris Yeardley showed up to pick up a laptop I had got for her, and spent some time looking at that; it's a hot-off-the-press Asus W1000. It has lots of multimedia stuff on it, including a remote control, and I was left wondering whether it was a computer or a TV. Spent some time playing with the multimedia stuff, and in the process brought out my new Inspiron 9100 and discovered it had much of the same stuff on it. I can't help marvelling, though, how bad all this stuff is. It has a thing called "Dell Media Experience™" which claims to have "the features and controls of a normal living-room DVD player". I suppose that depends on your definition of "normal", but I'd expect at least slow motion and search functions, all of which are missing. Comparing with Chris' machine, it appears that the "Dell Media Experience™" is really a front end to the equally emetic "Windows Media Player", though I can't see where the interface is. About the only thing that the latter has in common with the living-room DVD player is the stupid button panel at the bottom of the window:



It looks like a toy, but if the "toy" impression isn't strong enough, it's relatively trivial to convert this stupid window into an even more stupid-looking "skin":



On the other hand, I haven't found out how to display a DVD with it. I put a DVD in the drive, but it didn't find it. When I tried navigating the windows that Microsoft makes you use for finding files, they were there:



Look in: LSIL87484730 (D:)

VIDEO_RM
VIDEO_TS

File name:

Files of type: Media files (all types)

I can also display them if I go via the "Dell Media Experience™", so it's obviously just a matter of pressing the right mouse buttons. What a pain!

After that, to complete the joy of Microsoft, we tried to set up satellite networking with another machine that Chris had brought. We finally succeeded, but like with so many things about Microsoft, we don't know why. Very frustrating.

Monday, 12 July 2004

Up bright and early this morning to get back to "real" work. First, noticed that one of my nightly backups had failed because of lack of space, so decided to put in the new 200 GB drive that I had bought for that purpose. A couple of minutes into the backup, the system panicked with a divide fault. Tried to take a processor dump, but it didn't work. On rebooting, the system panicked again, this time with a different problem. Still no dump possible. Tried to do remote debugging and discovered that the sources were on the same system. But I didn't want to move them to where they belonged, because that was the file system I wanted to back up.

Tried a number of different things and established that it was definitely a software bug, presumably somewhere in UFS, triggered by the somewhat unusual parameters I set up for the drive, possibly the `-g 1000000000` ("average file size is 1 GB"). Gave up on it for now; the backup is more important. Hopefully I'll be able to reproduce the bug when I have time; it could be an interesting one for my debug tutorial.

After that, thoroughly exhausted, back to work on my program. Didn't get very far. I'm coming to the conclusion that programming is like running: you can sprint for a short distance, but you can't keep that pace up for very long, and you need to find a (slower) sustainable pace.

Tuesday, 13 July 2004

More stability problems; *zaphod* hung up during a level 0 dump shortly after 9 pm last night, presumably the old NFS deadlock. I've been trying to do a dump for 24 hours now. Grrr.

More work on my program today, and finally got back on track with what I had been doing before my current spate of hardware problems. Things are looking good again.

In the afternoon, another dump failure, this time caused by *kondoparinga* panicking. I wish I knew why I can't get a dump on this machine. Started the dump Yet Again.

Later in the afternoon into town for a normal and a special general meeting of the IT Council of South Australia The latter was to approve a new constitution, which limited the number of people on the board of directors to 9 (isn't that a popular number?),

> Both the AUUG board of directors and the FreeBSD core team have 9 members.

and elected new members, one of whom was me. It'll be interesting to see how things go now.

In the evening, *zaphod* hung again. Still no dumps!

Wednesday, 14 July 2004

My dump failed again! Only half an hour after I started it, *kondoparinga* (provisional name for new *wantadilla*) panicked in the `rl` Ethernet driver with a VM fault. That's certainly nothing to do with the disk system, but I still couldn't get a dump. I suspect a conspiracy of hardware and software. Addressed the `rl` problem by replacing the Realtek card with a 3Com, though it's not clear that it's a problem with the card or the driver.

One thing is becoming clear: *dump* is not the tool to use. It has a number of great disadvantages:

1. It only works on real file systems.
2. As a result, you can't make dumps of individual directories.
3. You also can't dump a network mounted file system.
4. *dump* has different formats on almost every system. Even in FreeBSD, *restore* on release 4 can't read dumps from release 5.

Years ago I recommended *tar* for these reasons; it has none of these disadvantages (well, there are still some minor format differences, but there are solutions to them). Time to return to *tar.* Today I started doing a backup of the top-level directories in */src*, which took at least as long. It's possible that *dump* has a slight performance edge. Also, *bzip2* must be one of the most CPU-intensive programs I know. It took over 2 hours of CPU time on *kondoparinga* (currently running as an Athlon XP 2500+) to compress a 17 GB tar archive. At least—by the evening—I completed the dump.

Thursday, 15 July 2004

More work with the air conditioner today, which has been icing up continually. Took the thing apart and put the sensor in the coolest part of the coil; hopefully we'll have some peace now.

Friday, 16 July 2004

Spent most of today looking through some code to add to my program, and finally decided that the easiest way to find out how it worked would be to look at it in action with other code that used it, not made any easier by the fact that my tree was out of date, and that I had unwittingly added updates since. Managed to get that sorted out and found another 5 levels beyond where I had got in the documentation. Debuggers are really useful, especially since the documentation can't say *which* parser is used. Spent the rest of the day analysing what I had found.

Another letter from Canon today, relating to my scanner. They're not prepared to do anything beyond refund the money for the scanner. I'll probably do that in the end, but I'm left with a bad taste in my mouth. It's clear that this scanner was brought to the market with inadequate software, and since they refuse to release the specs—by no means typical of the industry—they're effectively releasing non-functional products and just putting up with the not-too-frequent complaint. That's not the behaviour of a reputable company.

My hopes to have fixed the air conditioner proved to be premature, as I came out and found the de-icing sensor probe covered in ice. Decided that it was obviously defective, so removed it from the machine for Yvonne to take and get a replacement. When I had done so, I saw the underside of the switch (where the contacts are) for the first time:

The third contact (between the other two) is unused; it's normally open. But what's that screw in the middle? It looked like an adjusting screw, and some experiments with the deep freeze and some iced water proved it was. Put it back in the air conditioner, this time with the screw facing upwards, and was able to adjust the thing so that it de-iced properly. What a surprise! It's been over six years since we first reported this problem, and it's all a documentation problem, one that obviously no air conditioner technicians know about. Spent some time adjusting the screw, which is surprisingly sensitive.

I'm also left with the question: should I turn on early, which results in a de-icing cycle of only a few seconds, but which happens relatively frequently, or turn off late, which can take up to 30 seconds to de-ice, but which doesn't happen very often?

Saturday, 17 July 2004

Quiet day. I had intended to finish my work on AUUGN today, but didn't make much progress. Bottled some beer in the morning, and spent repeated short periods of time adjusting the de-icing switch on the air conditioner.

I'm beginning to think that the de-icing switch sensor is still in the wrong place. First, some background about the machine and how it works: like all air conditioners and refrigerators, it compresses refrigerant, which heats it up, and then allows it to expand, which cools it down significantly. The trick is maintaining reference temperatures.

Normally, fridges and air conditioners first pass the heated refrigerant through a coil (think: "radiator") to cool it down to ambient temperature; after further cooling through expansion, the refrigerant is used for cooling something, such as a room or the interior of a fridge.

It can also work in the other direction: after cooling, the refrigerant is warmed up to ambient temperature, again through a coil. Then it's compressed and gets a lot hotter, and that's used to warm things. This is the way we use it in winter.

There are two problems with heating:

1. The cooled refrigerant enters the coil at temperatures below 0 °C, which causes atmospheric humidity to condense and freeze. That's why we need de-icing. In a completely dry atmosphere, it wouldn't be necessary, and the coil temperature could easily drop far below 0°.
2. The switch measures temperature, not icing, and the temperature curve is very non-linear. It drops rapidly to 0°, but if enough ice forms, it then stays there: freezing water creates a lot

of heat. I can adjust the temperature to just about anything I want, but it needs to be very exact.

All this makes me think that maybe the sensor is mounted in the wrong place, or mounted incorrectly. I can think of two ways to take advantage of the physical situation:

1. Mount the sensor less closely in contact with the coil. It won't cool down until some ice forms, creating better thermal contact.
2. Mount the sensor higher up the coil, where the temperature is also higher (the refrigerant enters at the bottom and leaves at the top), and wait for the ice to form there before de-icing.

The sensor *was* originally mounted higher up, so maybe I should do something about that.

Sunday, 18 July 2004

Spent most of the day working on AUUGN, and more or less got finished. A large part of the job was just getting a framework together; next time round should be a lot easier, but I suspect if anybody else takes over editorship, he'll change the tools again, and it'll all be a throwaway implementation. Still, since I'm using my standard macros (*tmac.G*), it gives me a chance to fine-tune them, particularly the macros for generating the table of contents.

Monday, 19 July 2004

Back to normal work today, researching some existing code that I have to put into my program. The main issue is that it's part of a library, and I need to disentangle it. Made reasonable progress with that.

Also paid some attention to mail; I've now passed the 10,000 message mark, and I'm coming dangerously close to the 48 MB above which *postfix* refuses to store messages.

Also spent more time working on the new system installation stuff. It's getting better, but it's still not done. This is a real pain.

Tuesday, 20 July 2004

More work on my program today. Incorporating the existing code into my program turned out to be easier than expected. I had planned for the rest of the week to do it, but by midday I had the first half finished, the half that I thought would be more difficult. That probably means that the other half will be a pig and will keep me busy until the end of next week...

Wednesday, 21 July 2004

Today spent most of the morning looking at the second half of this week's work, a library function in our library which I needed to extricate from the functions with which it's bound. Then I discovered that both FreeBSD and Linux have the same functionality in the library. By that time I was just about done with the extraction, so I wrote a test program with both versions and confirmed that they both did the same thing. That's the rest of this week's work done already.

Thursday, 22 July 2004

More tidying up work today; it's about time I finally cut over *kondoparinga* as the new *wantadilla.* Installing OpenOffice still proves to be a pain, not made any easier by some Ethernet problems: yesterday a card died on me, and today one started reporting ridiculously long packets (over 32 kB). This caused a panic in `bcopy`, called from `rl_rxeof`, presumably because the interrupt handler hadn't checked the length before calling it. Unfortunately, I still can't dump the system, so wasn't able to do much except put in another card.

Some performance measurements on the work side of things, with interesting results.

Friday, 23 July 2004

Spent some time debugging today, most of it spent debugging the debugging macros. Didn't find the main bug I was looking for.

Apart from that, decided that I should finally complete the transition to the new *wantadilla.* Yvonne brought back some hardware today, notably a new hard disk for my laptop, so I can now install FreeBSD on the Inspiron 9100. A surprising amount of preparation is necessary for that. Managed to install Microsoft “XP Professional” on the new disk—hopefully moving to a new disk doesn’t constitute breach of their EULA—which will free up the big disk for a real operating system or three. Installation was surprisingly easy, and it didn’t even overwrite my FreeBSD partition, but at the end of it the only network interface it could find was firewire.

Saturday, 24 July 2004

Spent most of today playing around with hardware. We’ve decided that we can’t use the Digitrex DVD player any more, so spent some time looking for alternatives involving our TiVo but beyond finding a *vserver* already installed on the machine, didn’t make much progress.

Most of the time was taking up building software on laptops. On the Microsoft front, discovered that none of the Dell-specific drivers had been installed. The Microsoft install procedure doesn’t cater for installing vendor-specific drivers; considering that Microsoft itself is only half an operating system (native hardware support is minimal), that’s particularly bad. At the end of the install, I had to install each driver individually from the Dell CD-ROM, requiring multiple key clicks per driver, with almost no help as to whether I really needed the driver or not.

At the end, rebooted and discovered that, although I had asked Microsoft to install in “Drive `D:`” (the second partition on the disk), it had installed in the first two partitions; the first is a Dell diagnostic partition, only 39 MB in size. There seemed to be no way past that. In the end, deleted the partition (isn’t it nice to have a *dd* backup of the partitions?) and got it to work. In the process, discovered that FreeBSD’s *fdisk* can also work on copies of MBRs.

> There’s a copy of the output in the online diary. Basically, it looks like normal *fdisk* output.

In the meantime, also installed FreeBSD on the new disk on the other laptop (*eucla*), using the system upgrade procedure I’m working on. Had a few additional hiccoughs, but it seems to work.

Sunday, 25 July 2004

Somehow got nothing done today. Continued installing FreeBSD on *eucla*, which went a little better than yesterday, and by the evening had the machine pretty much the way I wanted it. The big success story was the new X system from X.org which installed without problem and started without a configuration file—this on a machine with a 1680x1050 wide screen display. Very impressive.

Spent some time burning CD-Rs for AUUGN. Given the failures we have had in the past, made two copies of each on different burners and checked that the other burner could read the results. They could, but there seems to be something strange about the way they’re burnt: I can use *dd* to read in a CD-R, and it reads in correctly, but then it gets an I/O error trying to read beyond the end:

```
# dd if=/dev/acd0 of=fedora2.iso bs=128k
dd: /dev/acd0: Input/output error
5093+0 records in
5093+0 records out
667549696 bytes transferred in 241.148143 \
  secs (2768214 bytes/sec)
```

The same thing doesn’t happen with CD-Rs burnt on other systems. I wonder what causes it.

Monday, 26 July 2004

Somehow didn’t get much done again today. Spent some time debugging allocation problems on my program, and found a couple of bugs, but they weren’t the cause of the problem. Mañana.

Tuesday, 27 July 2004

More work on my program today, and found a number of bugs, including the one I was looking for. Using *gdb* macros certainly makes things easier, but it also has its down side: at least one of the incidences of corruption that I thought I saw was in fact due to a macro misinterpreting data. Still, got past all the current problems and moved on to the next bug.

Wednesday, 28 July 2004

Still more debugging, and discovered more supposed bugs which were in fact due to bugs in the debugging macros. If only the *gdb* macro language weren't so obscure! Found all bugs and fixed all but one: there's one remaining endian issue in the index pointers. That's going to have to wait until tomorrow.

Creating the CD-Rs for AUUGN proves to be more of a headache than I expected: this quarter, amongst other things, we're supplying a CD-R of Fedora Core 2 In the past, we've always only supplied one the first CD of sets, and things have worked well. Not so this time: the first CD apparently doesn't even contain the base X window system. Looks like we'll have to supply CD 2 as well, which makes it look silly that we're not also doing CD 3. Grrr.

Thursday, 29 July 2004

Finally finished download disk 2 of the Fedora Core 2 CD-Rs today, so burnt one and tried to install it on *eucla*. The installation failed in the middle: the installed system couldn't read the second CD-R, which I had burnt and tested on the same drive. The same thing happened with one burnt on *adelaide*. Spent a lot of time checking, without coming to any firm conclusion, but it looks as if there might be something about the way FreeBSD burns CD-Rs that doesn't appeal to Fedora's kernel. An installation of Red Hat 9, almost the same system, worked fine and was able to read the CDs. It looks as if both Fedora and FreeBSD have to share the blame here: Fedora should be able to read CD-Rs that other systems can read, and FreeBSD should not burn CDs that are less readable than the ISOs from which they were derived. Grrr. David Newall came to the rescue with the CDs he used to install Fedora, and sent them off to the replicators.

In the process, took a cursory look at Red Hat 9. I hate it! This is a Microsoft lookalike. It took me several minutes just to find a way to start an *xterm*. When is the computer industry going to get out of sign language mode?

Off to Melbourne in the evening for an AUUG board meeting. Stayed at the Duxton Hotel not for the first time, but today had great trouble with the air conditioning, which seems to be a problem with the system: the temperature control is as good as useless. Why do people have so much trouble with thermostats?

Friday, 30 July 2004

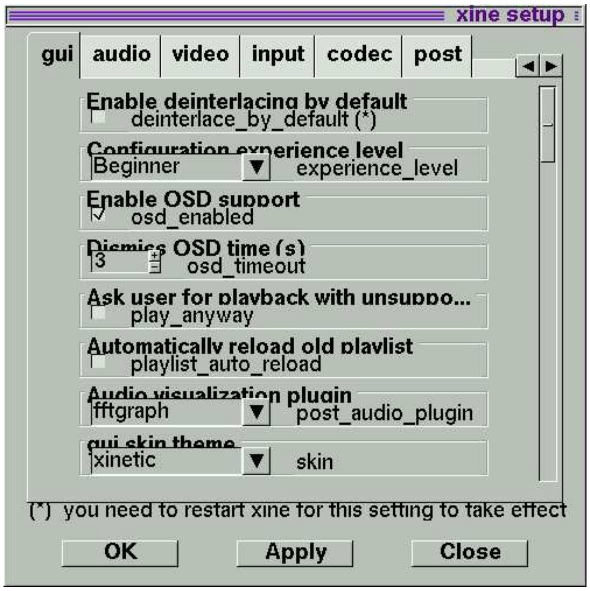
Up at the crack of dawn today, helped by the inclement temperature in the room, to start the board meeting at an atypical 8 am. As Immediate Past President, I'm taking a back seat this year. One thing that was agreed on (somewhat against my will) was that AUUGN will cease publication as a paper magazine, though we'll continue with some electronic form at least for a while.

Saturday, 31 July 2004

Another "new weekend" day, where I did no real work. Did spend some time looking for video software; it seems that most software is more concerned about silly-looking "skins" than functionality. *mplayer* seems to be the most popular one, but it doesn't have slow motion play, let alone reverse or slow reverse. After a while started installing *xine*, which at least promises slow motion, but I have this horrible premonition that I'm going to have to hack *mplayer* to do what I want. Somehow the free software scene is drifting offtopic.

Sunday, 1 August 2004

Took a look at the *xine* port I made yesterday. It's difficult to say whether it will do slow motion: all I got was three different SIGBUS crashes in a couple of minutes. Not very promising. It also has a configuration screen that looks like it has a web site in its ancestry, to judge by the poor layout:



Enable deinterlacing by default
□ deinterlace_by_default (*)

Configuration experience level
Beginner ▼ experience_level

Enable OSD support
☑ osd_enabled

Dismiss OSD time (s)
3 osd_timeout

Ask user for playback with unsuppo...
□ play_anyway

Automatically reload old playlist
□ playlist_auto_reload

Audio visualization plugin
fftgraph ▼ post_audio_plugin

gui skin theme
xinetic ▼ skin

(*) you need to restart xine for this setting to take effect

Monday, 2 August 2004

Back to work today, and managed to get my program working as intended, more or less. I have an important demo to do next week, requiring about 6 weeks of work to be done first, so spent some time looking at that.

Also looked a little bit at *mplayer* to see what would be required to do slow motion and frame advance. Very little, it seems. Within a couple of minutes I was able to influence the frame rate and do something approximating to a very poor slow motion. It seems that it should be relatively trivial (a couple of hours, as opposed to minutes) to get it to work properly. Why do people prefer to invent "skins" rather than implement basic functionality?

Just when I thought the air conditioner problems were over, the external unit failed completely this evening. Looks like the compressor has failed. *sigh*.

Tuesday, 3 August 2004

More slog today. Came to the realization that my attempt to do 6 weeks' work in one was not likely to succeed, and gave up, spending time filing down rough edges instead, with what looked like good results.

Call from Chris Butler of Multi-Tech today: one of the Fedora CDs David Newall sent him last week was marginal. This is a real pain! Asked him to recover it and compare it with the first ones we sent. Didn't hear back from him.

Phil from Mount Barker Air Conditioning came along in the evening and took a look at the air conditioner. Diagnosis: defective starter capacitor:

That was quickly changed, and the system was back up and running within 24 hours. A far cry from the times we've had to endure in the past.

Wednesday, 4 August 2004

Just when I thought we finally had our CDs sorted out! Got a CD in the mail from Multi-Tech: Chris had made a copy of the first Fedora CD I had sent, which had been OK in the first place, except that, once it had installed itself from it, Fedora couldn't read it. Confirmed that the same applied to the copy that Chris made. It's looking more and more like a bug in Fedora and less and less like a bug in FreeBSD though I can't completely exonerate FreeBSD.

Continued with tidy-up work today, and got some good results. Looks like things are in good shape.

Thursday, 5 August 2004

More work on my program today, and compared storage requirements with an existing product. The result: my program required 30 times more disk, and so of course was also a lot slower (though not by the same factor). There's obviously room for improvement there, which in the first instance will involve reassessing tradeoffs.

Still more problems with the CDs. It seems that the one I sent off yesterday was marginal as well.

About the only ones that are *not* marginal are the ones I burnt on the CD-R burner on my Dell Inspiron 5150, but Fedora doesn't want to know about them. What a pain!

Friday, 6 August 2004

Still no confirmation that the CDs are OK. What a pain.

Saturday, 7 August 2004

Had to do some scanning this morning, so installed the Canon software on my new machine and did so. Then decided to try out the "workaround" that they had claimed that would allow reliable scanning of 6 slides every time (it should do 8): leave out top left and bottom right slides, and the rest should be recognized correctly. Well, they're not:

Only one of the 6 slides is correctly framed; numbers 1 and 3 have a black bar on the right, number 4 has one at the bottom, and the software thinks that 5 and 6 are the wrong way round. This is no exception: it gets most of them wrong.

I've had enough with this thing It's clear that Canon are not going to honour more than their absolute minimum legal obligations—given the inability of this software to scan slides and their continuing advertisements that it can, even this is in doubt—I'm going to return it. Now to find a scanner worth using.

Sunday, 8 August 2004

Exceptionally (for recent times, anyway) did some work today to fix up the problems in my program ahead of a demo tomorrow. That was a resounding success, so much so that I'm left wondering what I've forgotten: I exceeded my performance figures by about 60%. Things are looking good.

There's finally been a break in the rain, so we loaded our horses into the float and headed off to Kuitpo Forest to go riding. We weren't fast enough; it started raining on the way, so we just turned around and came back. It must dry up soon.

Monday, 9 August 2004

Jim Johnson along today to look at the work I've been doing and to discuss where the company's going. We're living in interesting times. At least the deadline is gone, and in a few days I can relax a bit. Maybe.

Received a brochure for AUUG2004 in the mail today. Grrr. That was what I was going to pad AUUGN with. Now I need to find 11 more pages by yesterday.

Tuesday, 10 August 2004

Finally a breather! A good thing too: in the morning I has 10,000 mail messages in my inbox, and it was threatening to overflow. By the evening I hadn't quite made it below 1,000 (but it was below 1,023).

Finally also got AUUGN out of the way, after finding another article to print. Thank God for that! Now I have to start with the next one almost immediately.

Still more interest in my performance figures. I don't understand myself why they're so (comparatively) good, but I suspect it's not of my making.

Wednesday, 11 August 2004

Ross Williams out here today for a discussion of the work I've been doing and our future direction. A long discussion, but we came out with some useful stuff.

While talking, had a minor mail disaster. Five days ago, I tried to forward over 5,000 spam messages to `abuse@ctbcnetsuper.com.br`. I've never seen anything but spam from `ctbcnetsuper.com.br` (and, in fact, in my experience the Brasilians are currently very heavy spam senders, probably not the impression that the country wants to make in the rest of the world), but this one was extreme: the messages came in in less than a day. I should have been more careful, though: this seems to be a send-only site, and there's no way to send mail to them. As a result, I ended up with them all expiring today—and being returned to me with multiple error messages. Took me over half an hour just to delete them all again.

Thursday, 12 August 2004

The results of yesterday's mail crisis were evident today:

```
On 11 August 2004 you received 35771 mail \
   messages and sent 28 mail messages.
```

I still can't dump a FreeBSD -CURRENT system, but I now have the Fedora CDs back, the ones that David Newall sent in. They worked fine, and I was able to install Fedora on *eucla* with no particular problem. Then decided to update FreeBSD on a third partition, which proved very frustrating: the disk partitioning tools seem to be broken, probably as a result of GEOM changes, and at one point I thought I would have to restore the entire existing FreeBSD partition. Installed FreeBSD 5.2.1 from CD-ROM onto the spare partition, and miraculously the existing partition came back to life.

Was just starting an installation of CURRENT on the new partition when I got a message from Brett Lymn that a number of older SPARC machines were available for grabs from his company, so up to Edinburgh to take a look, and ended up coming back with 9 SPARC machines ranging from an IPX to a SPARCStation 20, as well as a couple of old DEC Alpha servers. None of them have disks, which makes things a little more complicated, but they could be useful as diskless workstations. I'm prepared to give some of them away to anybody who's interested and has an interesting use for them.

Friday, 13 August 2004

Testing went on in the background today, which was a good way to do it: I had some strangeness which only showed up after fifteen minutes of running; setting conditional breakpoints managed to get *gdb* to use twice as much CPU time as the program, making things even slower.

In the meantime, updated the kernel on *eucla*, in the process showing that my system upgrade procedure (*newsystem.html*) works relatively well. Dumping a processor from *ddb*, however, doesn't. at least not with the `panic` command; you need to use `call doadump` instead. It's been like that for I don't know how long.

Running a *tcpdump* on my right-hand monitor has proved useful on multiple occasions. This afternoon I saw a whole line of this on the screen:

```
16:22:35.753430 < 213.180.193.68.56911 > \
  192.109.197.80.27953: S 2706038967: \
  2706038967(0) win 3072
16:22:35.770356 < 213.180.193.68.56911 > \
  192.109.197.80.48381: S 2706038967: \
  2706038967(0) win 1024
16:22:35.770439 < 213.180.193.68.56911 > \
  192.109.197.80.2658: S 2706038967: \
  2706038967(0) win 4096
16:22:35.770530 < 213.180.193.68.56911 >  \
  192.109.197.80.2124: S 2706038967: \
  2706038967(0) win 3072
```

This carried on at a rate of about 50 probes per second for nearly an hour, by which time I had received (and firewalled) about 200,000 packets, or 10 MB. It was relatively easy to find out who was doing it:

```
$ host 213.180.193.68
68.193.180.213.IN-ADDR.ARPA domain name \
  pointer proxychecker.yandex.net
```

Send a message to `abuse@yandex.net` after which the bombardment stopped. Some time later I received a reply to my message stating (in lines too wide for the screen, let alone for this journal):

```
Date: Fri, 13 Aug 2004 12:58:31 +0400 (MSD)
From: abuse@yandex.ru
Reply-To: abuse@yandex.ru
X-Originating-IP: 213.180.211.108
X-FORWARDED-FOR:
???????????, Greg 'groggy' Lehey
-----?????-----
Dear sir,

Your've detected activity of our corporate
open proxy checker.  It is being used to
submit and validate entries to our
+corporate block list (insecure hosts part).
Somebody of our users has sent spam complain
related to affected hosts I guess.  It's a
reason to schedule automatic +proxycheck.
Pls note that further rechecks will be
suppressed for a significant time.  Though
full portscan is a little bit noisy (sorry),
unfortunatly it is the only method to detect
several modern kinds of +open proxy spam
sources (mainly trojaned and infected
hosts).  PS: Yandex LLC is a major russian
internet content provider.  Spam is a real
problem for millions users of our services.
+That's why we use any chance to detect and
block spamsources.

Sincerely,
Yandex customer support
```

Some things leave me at a loss for words.

Sunday, 15 August 2004

More working on web pages today. In the lead-up to the AUUG 2004 national conference I tried to find the photos (*Photos-20020908.html*) of the last conference in Melbourne, only to find that they weren't there. On further investigation discovered a large number of photos that I hadn't converted to web pages, so spent some time doing that; ended up uploading nearly 200 MB of images to the web site.

Also processed Yvonne's old cookbook which I had scanned in yesterday. Had some fun writing a shell script (*scripts/makepages.html*) to create a framework around the images; for such a simple thing, it worked surprisingly well, and left me wondering to what extent people have thrown out control in the sake of ease of use when creating modern software.

Monday, 16 August 2004

Things are starting to little more normal again, but somehow I didn't have much to show for today. Spent some time debugging my program; as suspected, some of the blame was with the *gdb* debugging scripts. It's a slow business.

Also looked at the SPARC hardware that I got on Thursday. It's a good reminder of how quickly hardware becomes obsolete. The SPARCStation 10, the fastest machine, only had 32 MB of memory, and the SPARCStation 20 only had 64 MB, but it also had 2 CPUs. The SS 10 had a defective power supply (well, at least it had been removed), so took the memory out of it and put it into the SS 20, making just enough memory to install Solaris 9. Also had to find a disk to install on. The disks from my Sun Disk Array (*diskarray.html*) have the correct connectors, but they're full height drives, and their construction means that they'll only fit the top slot: the top slot obstructs the disk if you try to install into the bottom slot. That also means, of course, that I need to prop it up, and that I won't be able to put the cover on. Finally I've found a use for my old Fujitsu magneto-optical disks:

How slow this installation is!

Tuesday, 17 August 2004

Finally got round to looking at the tutorial notes for the debugging tutorial I'm doing in two weeks' time. Hopefully by then I'll get them finished and in better shape than previously.

Carried on with my program debugging and found another bug, interestingly apparently a day one bug in code over 10 years old. It'll take a bit of thinking to fix it, not because it's so difficult to find a solution (that's obvious), but it'll be difficult to prove that it's fixed.

Wednesday, 18 August 2004

What a hectic day! It started off normally enough, installing a new version of FreeBSD on *sydney* and NetBSD on the SPARCstation 20. The latter proved to be almost impossible with the CDs I had (1.5.2 from Wasabi), so downloaded an ISO.

In the meantime, Alan Kennington showed up for a technical discussion about networking, about which I know nothing, so spent most of our time talking about file systems.

That time was somewhat curtailed by a phone call from Ollivier Robert, who was due to arrive from Melbourne by car this evening. He had discovered that the distances in Australia were larger than in Europe (he had expected the distance to be about 300 km, although Élodie told me had originally thought about 100 km; the real distance is about 750). As a result he had decided to fly instead, so he was arriving at 14:50. Reluctantly curtailed the discussion with Alan (which we wouldn't have finished today anyway) and into town to pick them up.

Back home, ran into more trouble on multiple fronts: updating DNS for Ollivier's monster laptop didn't update the secondary properly, building the software for *sydney* failed, the disk I had put in the SPARCstation 20 proved (after significant checking) to be defective (wouldn't even probe), and *sat-gw*, my satellite downlink and firewall box, developed not one but at least three problems: it wouldn't let data through for Olliver's IP, I couldn't get an *ssh* connection, couldn't get a console display (the video board seemed to have lost sync), and to add insult to injury the **g** key on the keyboard failed. Ended up crashing the machine and rebooting, after which Ollivier's machine was on the net, but I still couldn't get an *ssh*; looks like that one is my fault, probably a broken firewall config.

Thursday, 19 August 2004

Somehow another day where I got nothing done. The news of the day was the breaking of the MD5 algorithm, which we have been using. Looks like we're going to have to find something else. SHA1 is a possibility, but the hashes are even longer, and it's marginally slower. Also fixed (hopefully) the complicated program bug that I found last week.

Apart from that, got NetBSD installed on the SPARCstation 20, which I've decided to call *dumpty*. Some day I must write a web page describing the crazy ideas that lead to my system names. What I didn't get done was any further work on my tutorial; that's now becoming urgent.

Friday, 20 August 2004

ADUUG lunch today, with Ollivier and Élodie, of course, but also with Andy Johnstone of the Australian Democrats for the first time; Andy's the real reason for Ian Gilfillan's recent Open Source Bill (*http://sa.democrats.org.au/Media/2003/0625_a%20Open%20Source.htm*), and he aroused some interested discussion.

ADUUG lunches are always fun, and this time we had a total of 13 people show up, more than usual; but somehow it cuts the day in half, and I didn't get much done in the remainder of the day, just enough to confirm that, though I had fixed the bug I had been dancing around all week, it wasn't the one that was causing the corruption I saw last week. *sigh*

Saturday, 21 August 2004

Up early this morning and out to Clarendon with Ollivier and Élodie to meet Michael Hickinbotham, who showed us round his Clarendon vineyard. A nice place. They grow most of their fruit for big-name vintners, and I discovered that they're the owner of the "Old vine" Grenache vines (70 years old) from which Normans produced a wine of the same name some years ago, and which I had found particularly good.

Catchup time in the afternoon. Building a NetBSD (*http://www.NetBSD.org/*) kernel for the SPARCstation 20 proves more difficult than I had expected: the kernel finally built, but crashed immediately on booting. Did some puzzling over that, but given the speed of the machine, didn't finish.

Monday, 23 August 2004

Ollivier and Élodie left for Kangaroo Island this morning, leaving me to catch up on the work that has accumulated in the last couple of days. Somehow I didn't have much to show for it. Made some progress on the tutorial notes for the AUUG conference though not as much as I wanted; it seems that the kernel debugger really is still broken, which is a real nuisance. Tried to repeat the crashes I had experienced last month (12 July), without success. As I've noticed elsewhere, never rely on a bug. Even firewire debugging wouldn't work properly, which is disappointing.

Tuesday, 24 August 2004

Into town this morning for the inaugural meeting of the Board of Management of the IT Council of South Australia (*http://www.itcouncil.asn.au/*), again with Yana. There's something strange about these board meetings: I've been to about six of them, and today for the third time we encountered kangaroos while bypassing Hahndorf. It's the second time Yana was with me but every time I was on the way to an IT council meeting. You'd think that there would be some connection, but the only one I can think of is that Adelaide is a town not far from the bush.

The changes have made a big difference in the size of the meetings, which makes them more manageable, especially since not all positions on the board have been filled. On the other hand, that leaves fewer people to do the work, and I found myself chairing a membership subcommittee. That'll be interesting.

In the afternoon, more work on my tutorial, but also found enough time to find the next bug in my program; possibly it, too, is a day-one bug, but I'll have to investigate that more carefully, probably after the AUUG conference.

Wednesday, 25 August 2004

More work on the tutorial notes today, and finally got something worth submitting. It's really amazing how much work they are. Hopefully they're better than they have been. Didn't get much else done. Even managed to miss an AUUG board teleconference—the first time I've ever missed an AUUG board meeting, which left me less than happy.

Thursday, 26 August 2004

Spent most of the day working on my tutorial slides. It's difficult to know how to present this kind of material. In Taipei, as last year at AUUG and BSDCon, I presented directly from the tutorial notes, but particularly in the large lecture theatre in Taipei that was a problem. On the other hand, slides don't contain much information, and there are so many details to consider here.

Finally got our Digitrex DVD recorder back from "repair", after being away for nearly a month. According to the service report they changed the fan, and indeed it's a lot quieter. The other matters are allegedly "`USUAL FOR THIS MODEL`". What now?

Friday, 27 August 2004

Relatively quiet day today, spent preparing for tomorrow's journey. It wasn't made any easier by the fact that *zaphod* paniced during the night; it's not clear whether the new replacement disk is not in fact also defective.

Saturday, 28 August 2004

zaphod paniced again during the night with some file system consistency issue. Why do these things always happen when I'm leaving for a conference? Took a dump and put all the information, including the source and object trees and yesterday's dump, on *eucla* to look at when I get to Melbourne. Looks like there's something happening during the night to trigger it.

Sunday, 29 August 2004

Got to the Duxton Hotel through very heavy traffic shortly before 6 pm. Out for dinner with Steve Bellovin, Jeffrey Hsu and Carl Makin; Theo Deraadt was there too, but declined to join us: he had apparently already eaten.

Monday, 30 August 2004

Today was my all-day kernel debugging tutorial, which went smoothly enough. Five participants (got to remember that that corresponds to about 75 in America), all of whom were primarily interested in Linux. It looks like I'm going to have to add more Linux-related material. Deviated somewhat from the course notes, and spent some time looking at the *zaphod* dumps, which seem to be due to a vnode being marked VBAD and being caught in an `INVARIANTS` section because a pointer hadn't been recycled; possibly a harmless omission.

In the evening to Lygon Street for dinner, at a place on the South-East corner of Lygon and Grattan Streets, where we somehow managed to get all the drinks for free. Considering they charged $10.50 for a basket of garlic bread, I don't think they suffered too much, though.

Tuesday, 31 August 2004

Jeffrey Hsu's tutorial today, a Linux code walkthrough. I've been in this sort of thing before, but it seems that it's a relatively unusual thing in the Linux world. It was also Jeffrey's first tutorial of this nature, and he somewhat overestimated the time it took people to read and understand the code. From my viewpoint, things were different. After the pain of the code reading I did in April and May, I found it refreshingly easy to follow.

Jeffrey also had a number of interesting tools and methods. He brought a number of books, most of which I had, though not Comaen, Leiserson and Rivest: "Algorithms",McGraw-Hill 1989, which I think I'm going to have to buy. In addition he showed a new tool called Source Navigator (*http://sourcenav.sourceforge.net/*), in the style of *cscope* and *etags*, but ostensibly more powerful.

Spent quite some time trying to install the FreeBSD port, showing all the weaknesses of the Ports Collection in the process: *pkg-add -r* didn't work (claimed it couldn't find the URI), dependencies were wrong, and when I finally got everything installed, I didn't know how to start it, and there was no obvious documentation. When I finally got it started, discovered that it would generate about 2 GB of cross-reference data (*TAGS* is 1.3 MB), and I didn't have space. Jeffrey used it during his talk, and I found I was consistently faster with *etags* than Jeffrey was with Source Navigator. It's doubtless of use, but the menu pushing slows it down too much for a number of uses.

Speakers' drinks at the Royal Melbourne Hotel in the evening, after which down to the other end of Bourke St to a NetBSD beer drinking evening. Late to bed.

Wednesday, 1 September 2004

First day of the conference today, and the first time for some time that I've been here almost purely as a participant. This year seems to be particularly good; we've had the highest attendance for five years, and everything seem to be going smoothly.

Started off with Senator Kate Lundy, shadow IT minister (and possibly the real one in a couple of months; today was also the start of the election campaign). I heard her at the "Open Source in Government" conference in January, but this one was a different talk, and much better.

Then listened to a few talks on system configuration. Adrian Close's talk on "Kara" was particularly interesting, since it's a different approach to the system configuration scripts I've been working on since the beginning of July. I suppose I'm going to have to look at that in more detail. Then Michael Paddon, ostensibly talking about scalable remote firewalls—an interesting idea in themselves—but in fact more a discussion of R-trees, which might be an interesting thing to look at in context of Monkey. He later gave me a link to some source code (*http://www.rtreeportal.org/*) which I must look at, though he says the code is not functional in that form.

After lunch heard more about VoIP from Andrew Rutherford; I still need to look at that in more detail. Steve Bellovin did an interesting and thought-provoking talk on identity and privacy, probably the closest we came to the conference theme "Who are you?"

The Annual General Meeting went fairly painlessly, though it's clear that we still have two different groups of opinion about what to do with AUUGN. Frank Crawford's prepared to get involved, which is a great idea.

Networking reception in the evening, then out to the Indian Restaurant around the corner, which was better than I recall it, though it's always a bit of a waste after the snacks at the reception. Relatively early to bed.

Thursday, 2 September 2004

Interesting talk this morning by Theo de Raadt about the work they've been doing in OpenBSD to make buffer overflow exploits much more difficult. I had heard of much of it before, but it was well presented and made a lot of sense. Later in the morning Warren Toomey talked about what they're doing now with TiVo It's interesting to think how much has happened since our first discussions (*diary-sep2000.html#28*) of this topic. He and Luke Mewburn showed me a few things that I didn't know. Overall, though, I still don't like TiVo too much, and much of the demonstrations just went to confirm my prejudices.

In the afternoon I was session chair for a keynote speech by Jess Healy the Democrats lead senate candidate for Victoria at only 19 years old. She gave a good talk within the time constraints allocated. After that Bdale Garbee with issues of trust in the Debian project; good food for thought for the BSD projects too.

In the evening the conference dinner with its usual madness. We got started earlier than usual, even before the entree, and we were decidedly flagging by the end. David Purdue had organized a wine tasting competition, unfortunately with a 1995 Lenswood Riesling which was so far beyond its prime that all the "experts" thought it was very young. David Newall won, or, as he put it, was the last one left sitting.

Friday, 3 September 2004

Up later this morning, as usual after the dinner last night, missing Steve Alford in the process. It's a pity they put good talks on on Friday mornings.

Most of the good papers seemed to have been in the first two days, and I didn't go to anything of particular interest; instead spent a lot of time in the secretariat arranging photos and taking more for people who wanted some. I've decided that this Nikon Coolpix 880 is not the best in the world for flesh tones, and it often leaves people looking as if they have been drinking all night (today was, admittedly, not a good day to test that).

The conference wound down rather suddenly, as AUUG conferences tend to do, though we spent a bit of time in the bar talking about all sorts of futures. Pia Smith, president of Linux Australia, had turned up rather suddenly in the middle of the night, and we spent a bit of time talking.

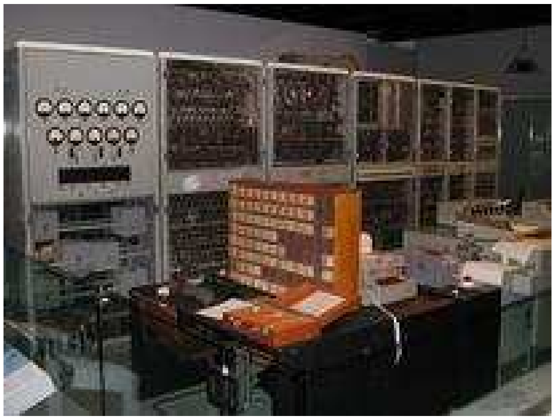
Then off with David Newall, Ollivier and Élodie to the Indian restaurant we had visited on Wednesday, this time with a little more appetite. David was telling me that CSIRAC, Australia's first computer, is now on display in the Melbourne museum, and that it's the oldest surviving computer in the world. Changed my return plans to see that.

Saturday, 4 September 2004

Intended to go to the museum with Ollivier and Élodie this morning, but their time schedules didn't mesh too well (I keep forgetting how much work a baby is), so off by myself to the museum in Russell Street. A good thing that they didn't come with me: the museum moved to Carlton years ago, and I had to come back and get the car.

The new museum reminds me in concept of the pyramid of the Louvre: it's right next to a staid old building, and it's ultra-modern. Both museum and Exhibition building don't match up to their Parisian equivalents, of course. The CSIRAC exhibition was interesting for a number of reasons: I had never seen a mercury delay line before:

Just the presence of the machine was imposing:



Unfortunately from my perspective, it was aimed towards predominantly non-technical people. The drum was labelled "disk drive", and though the machine was not running (understandably; maintenance and power would have been a nightmare), the display lamps had been wired up to a random on/off generator of about 8 Hz. There was almost no information about the technical details of the machine (I later discovered it was a 20 bit machine running at about 1 KIPS (0.001 MIPS), and that it had about 2,048 words of memory). I should have a chat with the people who did the exhibition; I consider that this is the world's most important old computer exhibition, and I could imagine a lot of interest from technical people world-wide. Also, it's possible that we could put some of Australia's early UNIX history in there too; this needs discussion before I can give details.

Sunday, 5 September 2004

Quiet day, spent getting back to "normal", though I didn't try too hard.

The repair of the Digitrex DVD player was too good to hold. In the ten days after getting the machine back, it didn't hang at all, a very long time by comparison with its prior behaviour. Today it did, however—exactly the same symptoms as before. In combination with the service report I suppose that this means that the machine is not repairable, and that this problem is a known (and evidently irreparable) fault. I shall contact Digitrex and ask for them to refund my money.

Monday, 6 September 2004

zaphod is still crashing regularly every night—in fact, twice a night, as Yvonne discovered. Debugging isn't helped by various bugs in *gdb* that stop me from finding the name of the file: it's passed to `namei` as a pointer into userland, and the new version of *gdb* in -CURRENT refuses to descend into userland. The development version *gdb6* simply crashes trying to dereference it. Looked at it with *ddb* and discovered a relative pathname (*sample3.bz2.uu*, FWIW; there must be about 30 of them on the disk) and no associated directory name, which is puzzling. Looks like more work with firewire.

Apart from that, spent some time catching up and investigating userland exits to implement file systems. Linux has a "loopback file system", but it seems that there's nothing equivalent in the BSDs. Looks like another diversion ahead.

Tuesday, 7 September 2004

Another day spent mostly in catchup mode. Managed to get my mail down to more manageable proportions, but wasn't very successful in finding useful userland file system stuff. There are a surprising number of strange file systems in */usr/src/sys/fs/*. Things like *nullfs* and *portalfs* seemed to be possibilities, but they're not really what I'm looking for. Didn't find anything very useful on the web either. It's amazing how much time you can spend looking.

Wednesday, 8 September 2004

It's looking more and more like I need to create a VFS loopback layer, not something I'm looking forward to. I need to learn more about the structures, so what better way than to kill three birds with one stone? *zaphod* has been panicking daily for nearly two weeks now with the panic message `cleaned vnode isn't`. Spent most of the day investigating that and documenting the paths through the VFS layer, which will also be useful both for my understanding of the code and also for the next iteration of my kernel debugging tutorial. As was to be expected, I ended up in a number of dead ends, but it's probably instructive to document how I got there and why. As a result, didn't find the bug: in particular, it looks as if `VBAD` is the correct type for an unused vnode.

Thursday, 9 September 2004

There's another project coming up at work, and I've been asked to review. I've been putting the matter off for nearly a month now, so it was time to do something about it. Today was the day, and it kept me busy nearly all day.

Also looking at the *zaphod* panic issue, which is still puzzling me, not helped by problems getting firewire debugging to work. I wonder if the software is too old; since it's development software, there's no particular requirement for interoperability.

Friday, 10 September 2004

Some days I seem to get nothing done, and today was one of them. It's not that I didn't do anything; I worked all day long. But at the end of the day, there was nothing to show for it, except maybe a few loose ends tidied up. I suppose it comes back to the question of mail: "reading mail" is only part of the equation: a large proportion of my work goes into mail, but I look on it as mail, not work. Time to reconsider things.

Sunday, 12 September 2004

In the afternoon, more investigating the *zaphod* panics, which are getting more and more confusing as I investigate them. This may end up being a race condition and not a hardware problem after all.

Monday, 13 September 2004

More work on my crashing *zaphod* today. This is really quite a fascinating panic, and though I suspect I could fix it faster if I didn't document what I'm doing, I think that it's an excellent example of a kernel bug. Since the system only crashes at night and doesn't seem to do itself any harm in the process, there doesn't seem to be any priority in fixing the problem.

Conference call in the afternoon. Looks like I'll be implementing (or finding) a userland file system callout.

Tuesday, 14 September 2004

Gradually I'm making some headway about userland file systems, paradoxically after replying to a message about Vinum in a DragonFlyBSD mailing list. It seems that Matt is very keen on the idea, and is even thinking of migrating less important file systems, such as cd9660, to userland—once the interface works, of course. Also heard that ARLA (*http://www.stacken.kth.se/projekt/arla/*) does this, and that it's very portable. Unfortunately, the site was down, but there's still plenty to chew on.

Wednesday, 15 September 2004

More investigation of the file system layer today. The ARLA site was back up, and I spent some time looking at the code, which of course doesn't compile. ***sigh***. I suppose it's worth the trouble to fix things first.

Thursday, 16 September 2004

Into town today and to North Terrace to meet John Sanders, the new executive director of the IT Council of South Australia Interesting person; he certainly has a lot of energy, and I can see a number of changes coming.

Digitrex has finally approved a refund for my abortive purchase of the GKX-9000 DVD recorder That's mainly good news, but it means that I need to find some alternative. While in town, purchased a DVD+RW burner, a couple of tuner cards and a TV out card, then home to try them out and discover that an old version of -CURRENT recognizes one of the tuner cards and the TV out card out of the box. That's more than I expected, but I still fear that I'm going to have a lot of work ahead of me.

Friday, 17 September 2004

More work on ARLA today: looks like I have just become maintainer, though it looks as if there's a non-committer out there who really uses it. More administrative stuff.

In the evening, put together the TV equipment I bought yesterday. Given the fact that it all comes with software for Microsoft, it seemed reasonable to use one of the Microsoft licenses that I have had forced on me to try it like that first. How I hate Microsoft! It ***didn't*** work properly. The DVD playing software I have only works if the display card is set to output to monitor only; as soon as I set it to dual head (including TV), it hangs hard and needs to be restarted. What kind of use is that? Also, once again everything is done with toy "skins" like the ones I commented on a couple of months ago (11 July). To stop the program, for example, you need to find and click on the button marked POWER on the toy display. Even the skins themselves suggest some teenager having fun with developing his own fantasy boxes rather than anything you'd ever see in a HiFi shop. Am I the only person thoroughly put off by these toys?

Saturday, 18 September 2004

Spent most of the day investigating the issues surrounding computer TV. They took a surprisingly long time, not the least because of the old issues with installing new versions of FreeBSD. Still, TV out works, for a minimal definition of "work". It seems that the output is made by dithering the image, and the resultant quality is pretty terrible. Hopefully—when I get that far—an X display will be much cleaner.

Sunday, 19 September 2004

More work on the video recorder (*video-recorder.html*) today. What a pain! Installing MythTV (*http://www.mythtv.org/*) was complicated by some of the strangest code I've seen in a while:

```
 /* ci must be 0 */
  if( bitoff == 0 ) {
     (uint8_t)strm[co]= bitten;
     co++;
  }
  goto BAUCHWEH;

 HIRNWEH:

  (uint8_t)strm[co]= bitten;
  co++;
```

The `(uint8_t)` casts (of which there are several) are illegal, of course, but the comments seem appropriate. Fixed that, and at least got it to compile.

Investigating the Nvidia display board I had proved to be useful: rather to my surprise, I was actually able to get it to run two displays under X.

Later installed mplayer on *teevee*. Installation went OK, but getting it to run is a different matter. As in the past, I had trouble. Display across the network is impossibly broken, and for some reason I couldn't get it to work at all on the local (monitor) display `:0.0`. The message was:

```
SwScaler: using unscaled Planar YV12 -> \
  BGR 8-bit special converter
BGR not supported, please contact the developers
FATAL: Cannot initialize video driver.
FATAL: Could not initialize video filters \
   (-vf) or video output (-vo).
```

After a fair amount of discussion on IRC, and with help from google found out the problem: this is *mplayer*'s inimitable way of saying "I can't use an 8 bit display". After setting the default depth for the display to 24 bits, it worked fine. What a pain!

Monday, 20 September 2004

More work on the ARLA stuff today. I got a message from one of the people involved in the FreeBSD port, which is currently broken as the result of recent changes in the FreeBSD bio layer. So what do I do? I'm not married to FreeBSD. It's supposed to run on NetBSD and Linux as well. Considered the options and decided that running it on NetBSD would have some advantages (Linux is too different at this stage, and the NetBSD port would help me fix the FreeBSD port), so downloaded an ISO and started installing it. Too late, I realised I had installed it on a SCSI drive whose controller has no boot ROM, so I won't be able to boot from it. ***Sigh***.

Also reading about AFS, which I've never really looked at before. It's quite interesting.

Tuesday, 21 September 2004

Still working on getting test equipment set up today. For some reason, a NetBSD i386 CD-R of the latest release (1.6.2) doesn't install cleanly: it seems to contain the wrong kind of binaries. Spent a lot of time installing over the network (the first time I've ever done that), and then building the rest locally. I wish this didn't take so much time.

Finally got through to my father on another phone line. I'm beginning to wonder whether it's not just a very poor user interface to a voice mail system.

Federal elections are coming up in Australia (on 9 October). We've heard the slanging matches in the media (summary: "Who's the bigger liar?"), but I thought that was probably not a good basis for a vote, so I spent some time investigating the web sites of the parties. The results weren't very helpful:

- I can't find anything on *http://www.liberal.org.au/*. They have a link "Election Policies which looks promising, but it's a collection of large PDFs, totalling hundreds of pages and with misleading titles like "Supporting the Territory" (it turns out they mean the Northern Territory; though this is an accepted abbreviation, in this context it suggests that they don't know that Australia has other territories, nor indeed that it is itself a territory). The Liberals also expect me to use specific software, which I don't have, so I suppose I'm not welcome anyway.

- *http://www.labor.org.au/* doesn't seem to be interested in the election. Apart from using it as a way to slate the Liberals ("Liberal's IR Lies"). the only policy they mention (indirectly) is the "Indigenous Policies". The word "election" does not appear on their home page.

- *http://campaign.democrats.org.au/index.asp* get it almost right. They have a link to *http://campaign.democrats.org.au/minifesto.html* at the top of the front page, and apart from being somewhat unsure of itself ("Please take the time to consider this short version of the Australian Democrats? election platform") gives the kind of overview (not necessarily the content) that I was looking for. The Democrats are also the only party I've looked at that didn't try to slate one of the other parties.

  About the only downside is that they don't eat their own dog food: after bringing through an Open Source Bill in the government of the Territory (another one), their web page appears to expect us to use a Microsoft non-standard character set. Not surprisingly, their web page doesn't validate, not even after guessing the character set ("Windows" 1252). It's also sad to note that the Democrats don't mention IT policy on their policies page.

- *http://www.greens.org.au/* have more in common with the Liberals than I think they'd like to admit: they have a policies page with lots of PDFs, but no overview.

It looks as if we have a long way to go before Australian politics really enters the Information Age.

Wednesday, 22 September 2004

Feedback from the Australian Labor Party today: the web site *http://www.labor.org.au/* is not the official web site of the Labor party. For that, you need to look at *http://www.alp.org.au/*, not an immediately obvious name. So I did that and did find a number of policies, nicely marked "Election Policy". Makes you wonder what the difference between an Election Policy and a non-Election Policy is. Unfortunately, it's still too detailed: I still can't find an overall statement of Labor's policy and values. I asked the web master why they didn't have a link at *http://www.labor.org.au/* and he replied that they were separate organizations (he did, though, admit that it wasn't much of a reason).

Fewer concurrent things to do today; the big one was to get ARLA up and running on NetBSD, which required an all-day build to get to NetBSD-CURRENT. Took advantage of the time to finally migrate *wantadilla*, a task I had begun over two months ago I still don't have my instructions for system upgrade (*newsystem.html*) complete, but the time was ripe, so swapped things around.

It would be nice to say that things went smoothly, but of course they did not. I had a lot of trouble starting X. First, I had forgotten that USB was disabled, so I couldn't get the mouse to work until I realized what the problem was, shut down, reconfigured the BIOS and then rebooted. Then I still had a lot of trouble getting the X displays to work. Given my X configuration (*hardware.html#x2x*), that's a little advanced, but in the end had to give up the attempt to move to new display cards (SiS) and revert to the cards I was using in old *wantadilla*. All in all, I was down for about three hours. The good news is that almost none of this had to do with upgrading the software: it was new hardware that (without good reason) caused me the headaches.

Thursday, 23 September 2004

A whole day filled with meetings! Left at 7:30 am for Adelaide, where we had a meeting of the Board of Management of the IT Council of South Australia. For me the big issue was the membership committee, which met after the board meeting (not immediately after, as I should have liked; we ran short of time as a result). Discussed lots of things, but didn't get as far as we would have liked.

Lunch with David Newall in the "Corner Bistrot" in Bank St. Nice enough food, but I got the (incorrect) impression that they're pretty new, and they need more practice. Spent the time discussing the upcoming UNIX Developer's Symposium that we're trying to put together. It looks like it could be a real success.

Then back to the Radisson Playford for a meeting of the full IT Council; I get the impression that the effort was showing, and we finished early. With the current membership figures, the meeting looked a little superfluous. All that will change when we have our projected sharp increase in membership, of course.

Didn't get home until 18:30, to discover that my NetBSD build had crashed, probably as a result of the nightly crash of *zaphod*. High time to sort that out.

Friday, 24 September 2004

Yet another occasion where I find that simply building an updated version of the system takes forever. *chili*, the NetBSD-CURRENT system I started earlier in the week, has been running a 2.0H kernel since Wednesday, but building userland keeps causing problems. Yesterday it was because of the daily *zaphod* crash, and today NFS kept hanging. Finally decided to build it on the local disk (filling up a partition in the process), and experienced further hangs. I'd really expect Ethernet networking to be more reliable.

On the other hand, moving to the new *wantadilla* was less harrowing than I had expected. It seems that *postfix*'s pseudo-*sendmail* program has changed its syntax and now requires the names of the recipients on the invocation line (previously it was happy to find them in the headers). Possibly there's a command line option to do it, but for now I just added the names to the command line:

```
#!/bin/sh
# $Id: tv-programme,v 1.3 2004/04/04 \
  22:13:54 grog Exp grog $
# Get today’s TV program
PATH=$PATH:/usr/local/bin
D=‘date +%e | sed ’s: ::’‘
TVP=/tmp/tv-programme
TVPH=/tmp/tv-programme.html
# ebroadcast films
fetch -o $TVP ‘date +"http://www...."‘
cat <<EOF > $TVPH
From: Greg Lehey <grog@lemis.com>
To: Greg Lehey <grog@lemis.com>
Yana Lehey <yana@lemis.com>
Subject: Today’s TV films
Mime-Version: 1.0
Content-Type: text/html; charset=iso-8859-1
EOF
cat $TVP >> $TVPH
/usr/sbin/sendmail -i <$TVPH grog,yana,yvonne
```

Apart from that, some ports to add, and some configuration files I haven’t saved yet. I’ve kept the old *wantadilla* running as *blackwater* (a word play), which makes that aspect easy, but is otherwise a mixed blessing: a number of my systems still have the old file systems on *blackwater* mounted, not the new ones on *wantadilla*. The background here is that I also migrated the IP addresses, and *blackwater* has kept the old addresses. As a result, the wrong files were getting updated from the kitchen and the brewing fridge.

Saturday, 25 September 2004

I’ve got used to *zaphod* panicking every night, but this morning was different: after rebooting it wouldn’t connect to the network. Everything else seemed OK. I changed the network cards, cable and even the switch to no avail, but I just got timeouts on the network connection (which was otherwise showing normal values). In the end moved the disks to the CVR machine that I had started working on where everything worked fine. Looks like the motherboard is on the way out. Now the question: do the panics have anything to do with that?

Spent most of the day working on AUUGN, but somehow didn’t get very much done. One of the articles is a trimmed version of this diary, and I spent an inordinate amount of time getting the format right. I need to find a better way.

Sunday, 26 September 2004

As expected, the new *zaphod* machine crashed as before this morning. I really must get round to finding out what’s wrong here.

Spent most of the day working on AUUGN. Last time round seemed so easy, but this time it’s keeping me busy. Maybe it’s because I was no longer expecting it to be much work.

Monday, 27 September 2004

There’s been this funny smell of overheating boards in the Mike Smith Memorial Room (*msmr.html*) for a couple of days now. Nothing to be really worried about, but I spent some time wandering around sniffing, and found nothing.

This morning, after the obligatory reboot of *zaphod*, NFS didn’t come unstuck. Further examination showed that *blackwater*, the old *wantadilla*, had had a power supply fan failure and (apparently) shut down due to overheating. Spent more time rebuilding, and then sent Yana off to town to buy more hardware. I’ve had a really bad time of it lately.

Spent the rest of the day working on the *zaphod* panic, which proves to be much more complicated than I had expected. Also, finally, *chili* (the new NetBSD test box) was upgraded to 2.0-CURRENT, and spent some time trying to compile ARLA not helped by header file hell. How I hate GNU *configure*!

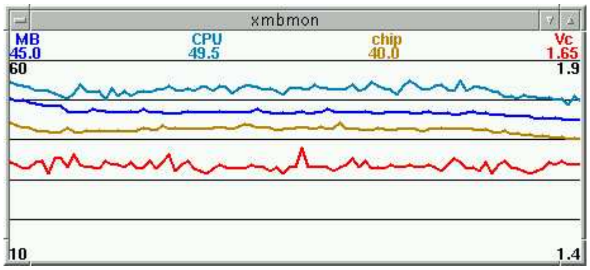
Tuesday, 28 September 2004

Still I can’t get away from hardware problems. Today started with a relatively minor one: the CPU fan on the ***new*** *wantadilla* (all of 2½ months old!) started making noise. Confirmed that it was still operating correctly, so decided to do what most people do anyway and put the covers on the case: I normally have this tendency to leave the covers off, as the photos of the Mike Smith Memorial Room show. That silenced the noise

One of the things that Yana brought back yesterday was a new (well, different) SMP motherboard that I had bought from somebody in town. Put that in the old *zaphod* case and discovered that it didn't power on. The lack of manual didn't help, and I spent over an hour before I decided that it wasn't going to work. Grrrr. The relationship between hardware price and the cost of time is changing to the point where I don't know why I bother.

On a more positive tack, managed to finally get ARLA to compile and "run" on NetBSD. I don't really know how to use it. Tried to build the documentation, but that seems to be broken too: it tries to reference an undefined file.

In the evening, just before bed, came in to find *wantadilla* dead. Further investigation showed it to have been overheating: when I rebooted it, the CPU temperature was still over 60°. Still further investigation linked that with this morning's well-intended replacement of the chassis covers. Took them off again, and the temperature dropped rapidly. The following screen shot was taken some minutes after it was back up and running ago. It shows the CPU temperature dropping from about 57° to under 50°:



I wonder how much effort has really gone into investigating air flow in computer cabinets.

Wednesday, 29 September 2004

Spent the day chasing up ARLA; somebody in the FreeBSD project had assigned a problem report to me yesterday, and I thought it was a new version of the port. but it's one I've seen already, and it doesn't build. Didn't have the stomach to try to debug it, so read some of the documentation instead; that could certainly do with improvement. In any case, it doesn't look as if I can do much with only a single system running ARLA.

Also continued on the panicking *zaphod* problem; it looks like I bit off more than I could chew here, though if I get it finished it will certainly be a good one. It's now fairly clear that the bug is related to corruption on the disk. I had suspected this all along, but now I have evidence. It seems that I had put a "do not *fsck*" entry into */etc/fstab* for this disk:

```
# Device  Mountpoint FStype Options Dump Pass#
# echunga:/src /src nfs    rw       0    0
/dev/ad1h /src       ufs    rw       0    0
```

The commented out line is the clue: this was originally an NFS file system, which doesn't need *fsck*, and when changing it I forgot to set the *fsck* pass number (should be 2). As a result, I have now had significant directory corruption, probably since a month or so. I could fix it, of course, and I'm sure that the problem would go away, but then I'd no longer have a case study.

Thursday, 30 September 2004

Into Adelaide today for a couple of meetings. I had originally planned them around a renewed attempt to have an ADUUG lunch, but once again it had been cancelled. As a result to an interesting discussion with Jim Johnson in the morning, then had a bit of time on my hands which I spent with Tim Aslat. He had sold me a dual CPU motherboard on Monday, but I hadn't been able to get it to power on. Brought it and the old *zaphod* in with me, and of course it powered up with no problems doing exactly what I had done on Tuesday. I hate looking like a fool, but at least it works.

In the afternoon had a face-to-face status meeting instead of the usual teleconference. We've hired a whole lot of new people, and one of them, Peter Pham, is going to be working in the storage systems area for a while.

Minutes of AUUG Annual General Meeting, 1 September 2004

Location: Duxton Hotel, Melbourne

- Apologies:
  Mark White, Kevin Sutton
- Acceptance of minutes 2003.
  - Dp mentioned that the acceptance of the minutes could not be done as usual since they had been lost in transit last year after the AGM.
  - It was agreed that Adrian Close will try and reconstruct the minutes based on memory and other people's input.
- President's Report

  DP mentioned that he had discussed his report with the AUUG executive to put forward the right ideals to the members. He started with saying "Hello".

  Mentioned that AUUG faces challenges, but they are not fatal challenges. Interest in AUUG has been increasing shown by the increase in sponsors and the success of this year's conference. Used the Olympic tag-line "Best conference ever!", certainly the best in five years.

  AUUG is focusing on its strengths and has to change with the times based around these strengths. AUUG is looking at re-igniting efforts for more small events around Australia.

  AUUG is currently working with Linux Australia for Open Computing in Government (OCG) conference in April 2005, plus other available co-operative events.

  Announcement of the promotion of Liz to executive director. This will allow Liz to delegate a lot of the secretarial work to others. Liz's work will now focus on sponsorships, promotion and the organisation of events within AUUG.

  AUUG is moving away from a paper-based AUUGN. This was not a decision justified by costs, but rather focusing on a sense of community and high quality content. It will be designed to make content available in a way that is cost-effective and timely. The web portal model that is going ahead offers advantages to allow for membership price reductions and other cost savings.

  DP made a call to all members to participate in the events and movement ahead of AUUG in the future. Members do not need to be on the board to participate or help out in events. Mentioned that CodeCon was being organised and run by Peter Miller on the Central Coast this month. AUUG will support that effort, but it will be run by a standard member. Any suggestions or proposals, write to `auugexec@auug.org.au`.

  Question:
  AC commented on the case of small events.
  Question:
  Asked about methods of addressing the debt and how it could be reversed.
  Answer:
  DP pointed out that AUUG was not deeply in debt. The answer was deferred until the treasurer's report which would address the question more specifically.

  Motion to accept the President's Report:
  Moved: ED
  Seconded: Peter Gray

  Carried....
- Secretary's Report

  A number of events run by AUUG were missed over the last number of years and this seems to have impacted on the AUUG's membership. This correlation seems to suggest that AUUG's members appreciate the events that are run by AUUG. With AUUG's new board, there is an emphasis on having more events that our members will enjoy.

  The number of attendees at AUUG2004 this year was in excess of 160 people. This number exceeded AUUG's expectations and is up on the previous five years. The number of new members signing up at this years conference are almost double that of the previous year.

  The total membership for this year based at time of the conference is approximately 350 members and over 800 subscribers to our AUUG mailing lists. These figures are not exact due to the organisation of the conference with changing memberships.

  JC mentioned that membership had improved with this year's conference and with an increase events, should see an improvement in the membership rates all up as well.

  Moved: Peter Gray
  Seconded: Paul Haesler
  Carried...

- Treasurer's Report

  GH mentioned that some previous years are still being done. The 2003 year shows losses of around $8000 and approximately $4000 for 2002. GH listed figures (shown in the official report). GH offered the full reports to be available for public scrutiny if people required.

  GH mentioned that a failed forecast made him wary of doing another for the future year. He does however expect a similar loss to that of last year. Pointed out that the income from the conference is up this year.

  Question:
  If losses, where is our balance?
  Answer: $180K standing, $160K expenditure. This included an amount of approximately $100K which is untouched by AUUG. This shows that AUUG can go on for many years, but want to turn the problem around to improve.

  DP mentioned that AUUG is not insolvent. The $8000 loss was around 3-4%.

  Question:
  What is the problem with accounting holdups?
  Answer:

  Timing and accounts make it difficult. Goal is to get to a stage of good reports for AGM. Next year should have them all ready. Still playing catchup with the past accounts.

  DP mentioned a compatible system with the accountant was being done.

  Question: Savings on the paper version of AUUGN going electronic?

  Answer:
  $20K. Main costs was with contributions.

  GL, the acting editor, stated that the reasons for the delay was not organisation or preparation, but printing and cdrom stamping.

  Question:

  Any drop in membership pricing coming up?

  Answer:

  DP mentioned that it has been discussed, but not disclosed at this point in time. GL mentioned the discussion and addressed follow-up question about printing AUUGN and willingness to pay for it.

  AfC mentioned that AUUGN was originally designed in a dead-tree world. Now there is the Internet. Want to find quality information in a timely manner. Getting the ball rolling to fit the current needs of the members and the technologies available.

  Comment:

  Look at it as a temporary move with re-introduction later.

  GH mentioned the option of having PDF files to be printed.

  This was acknowledged as being included. A comment was raised that the PDF should be printer friendly for members who wish to print AUUGN.

  Comment:

  AUUGN is not for announcements.

  DP mentioned the option for letters for legal reasons in making announcements. This would make it legal, and still save AUUG costs.

  Comment:

  GH mentioned the option for previous AUUGN material. GL mentioned that he is already following up on the archived material in previous AUUGN's. SL mentioned that the archives would be online in an archived format.

  Motion to accept Treasurer's Report:

  Moved: Peter

  Seconded: Susie Close

  Carried...

- Returning Officer's Report.

  Done by DP as Michael Tuke was absent. DP pointed out that insufficient nominations were made to be addressed by the returning officer. This meant that the whole board was elected un-opposed.

  ED and AfC were co-opted to the board. This requires ratification from the members. DP raised the vote for ratification.

  Moved: Frank Crawford Seconded: Michael Paddon

  Carried...

  Accepting the report:

  Moved: Susie Close
  Seconded: Steve Lynch

  Carried

  DP offered the position of returning officer.

- Other Business.

  No other business was raised.

- Meeting Closed: 5:40pm

AUUG Corporate Members

As of 1 June 2004

- Apple Computer Australia Pty Ltd
- Australian Bureau of Statistics
- Australian Taxation Office
- BAE Systems
- Cape Grim B.A.P.S
- Corinthian Industries (Holdings) Pty Ltd
- Cray Australia
- CSIRO Manufacturing Science and Technology
- Curtin University of Technology
- Cybersource
- Deakin University
- Department of Land & Water Conservation
- Department of Lands
- Everything Linux & Linux Help
- EWA-Australia Pty Ltd
- IBM
- IBM Linux Technology Centre
- IP Australia
- KAZ Technology Services
- LPINSW
- Macquarie University
- Multibase WebAustralis Pty Limited
- NSW Department of Commerce
- Peter Harding & Associates Pty. Ltd.
- Powerhouse Museum
- Squiz Pty Ltd
- Sydney Water Corporation
- Tellurian Pty. Ltd.
- The University of Western Australia
- Thiess Pty Ltd
- TMD Computing
- University of NSW department of Computer Science & Engineering
- UNiTAB Limited
- University of New England
- University of New South Wales
- University of Sydney
- University of Technology, Sydney
- Workcover Queensland

Letters to AUUG

This column contains selected messages from the `AUUG-talk` *mailing lists. To sign up for this mailing list, visit the mailman pages at http://www.auug.org.au/mailman/listinfo/talk.*

From: Greg Black <gjb@auug.org.au>
Date: Wed Sep 8 16:26:22 2004
Subject: [AUUG-Talk]: The future of AUUGN

Well, I've read Greg Lehey's editorial "The new face of AUUGN" in the current issue and it seems worthy of comment. I must say I'm disappointed the decision has been made "to stop production of the paper version of AUUGN by the end of the year."

However, if the cost of printing is so high and if we really could see a significant reduction in membership fees in return for this move, I'm happy to accept the decision.

I do hope that AUUGN will continue — preferably on the web rather than in the form of CDs, which also cost money to produce and distribute. In fact, I'd think a new form of AUUGN, perhaps built around a blog, might be a good thing. Then we could just subscribe to the RSS feed and be kept up to date.

I do like the inclusion of conference papers in the current issue and would hope that might continue.

Cheers, Greg

From: Greg 'groggy' Lehey <Greg.Lehey@auug.org.au>
Date: Wed, 8 Sep 2004 16:38:54 +0930

On Wednesday, 8 September 2004 at 16:56:14 +1000, Greg Black wrote:
> Well, I've read Greg Lehey's editorial "The
> new face of AUUGN" in the current issue and it
> seems worthy of comment. I must say I'm
> disappointed the decision has been made "to
> stop production of the paper version of AUUGN
> by the end of the year."

> However, if the cost of printing is so high
> and if we really could see a significant
> reduction in membership fees in return for
> this move, I'm happy to accept the decision.

This caused a surprising amount of discussion at the AGM. In general, the sentiment expressed was like yours: "pity, but understandable". I assume that this sentiment is not the only one: a significant number of members didn't read it, so they had no cause to complain about the decision. But I'd like to remind you about the option of having it printed. If people are interested (but only then), we can do some enquiries about the cost of such printing. It's possible that it would be comparable to the costs of producing AUUGN.

> I do hope that AUUGN will continue --
> preferably on the web rather than in the form
> of CDs, which also cost money to produce and
> distribute.

We're currently thinking of both.

> I do like the inclusion of conference papers
> in the current issue and would hope that might
> continue.

Thanks. That's an easy source of material for us, so if it meets with approval, then we can easily keep it up. That doesn't stop people from contributing, though.

Date: Wed, 8 Sep 2004 17:22:19 +1000
From: Greg Black <gjb@auug.org.au>

On 2004-09-08, Greg 'groggy' Lehey wrote:
> On Wednesday, 8 September 2004 at 16:56:14
> +1000, Greg Black wrote:
>> Well, I've read Greg Lehey's editorial "The
>> new face of AUUGN" in the current issue and
>> it seems worthy of comment. I must say I'm
>> disappointed the decision has been made "to
>> stop production of the paper version of
>> AUUGN by the end of the year."
>>
>> However, if the cost of printing is so high
>> and if we really could see a significant
>> reduction in membership fees in return for
>> this move, I'm happy to accept the decision.
>
> This caused a surprising amount of discussion
> at the AGM. In general, the sentiment
> expressed was like yours: "pity, but
> understandable". I assume that this sentiment
> is not the only one: a significant number of
> members didn't read it, so they had no cause
> to complain about the decision.

I've always read it, from cover to cover, and usually within a couple of days of receiving it. It's not always what I want, but I don't complain because I'm not prepared to provide the extra stuff myself and I know just how hard it is to produce a regular magazine.

> But I'd like to remind you about the option of
> having it printed. If people are interested
> (but only then), we can do some enquiries
> about the cost of such printing. It's
> possible that it would be comparable to the
> costs of producing AUUGN.

I would not be interested in a printed version if we turned it around into a web-based publication — so long as that version really was produced and accessible. (I note that there's a section of the AUUG web site entitled "AUUGN On the Web" that claims to include the text of AUUGN starting from December 2000, with issues added six months after publication — the last one listed is July 2002, so this is not promising.)

>> I do hope that AUUGN will continue --
>> preferably on the web rather than in the
>> form of CDs, which also cost money to
>> produce and distribute.
>
> We're currently thinking of both.

Are there plans to engage the membership in the decision making process, or would that be too difficult to manage?

>> I do like the inclusion of conference papers
>> in the current issue and would hope that
>> might continue.
>
> Thanks. That's an easy source of material for
> us, so if it meets with approval, then we can
> easily keep it up. That doesn't stop people
> from contributing, though.

Indeed, but having the conference papers on hand does provide for some meat in the magazine and it does get the papers out to a wider audience.

Cheers, Greg

Date: Wed, 22 Sep 2004 10:01:21 +1000
From: Russell Standish <R.Standish@unsw.edu.au>
To: Greg Black <gjb@auug.org.au>

I would definitely be interested in having AUUGN distributed and archived via the web. There should also be a table of contents distributed via email. I already read web versions of the IT sections of major newspapers, and I also read electronic manuscripts of scientific papers - these I download into my laptop, and I read them as I have time. Interestying papers are filed in a subdirectory of my "read" directory, otherwise I delete them as I read them.

Paper journals tend to just end up hogging space on my bookshelves. CDs just get lost!

Cheers

AUUG 2005 annual conference

Sydney, 9–14 October 2005

Call for papers

AUUG invites proposals for papers and tutorials relating to:

- Standards based computing
- Open source projects
- Business cases for open source
- Open source in government
- Technical aspects of Unix, Linux or BSD
- Performance measurement and management
- Software development
- Networking, Internet and the World Wide Web
- Identification, authentication and authorisation
- Applications of cyptography and cryptographic protocols
- Maintaining privacy
- Achieving anonymity on the Internet
- Internet security
- Other aspects of computer security

Presentations may be given as tutorials, technical papers, research papers, or management studies. Technical papers are designed for those who need in-depth knowledge, whereas management studies present case studies of real-life experiences in the conference's fields of interest.

A written paper, for inclusion in the conference proceedings, must accompany all presentations.

Speakers may select one of two presentation formats:

Technical presentation: a 30-minute talk, with 10 minutes for questions.

Research papers: a 30-minute talk, with 10 minutes for questions. Research papers must present original contributions to the field and will be peer reviewed by at least two reviewers. Each research paper will be judged on its originality, significance, relevance and presentation.

Management presentation: a 25-30 minute talk, with 10-15 minutes for questions (i.e. a total 40 minutes).

Panel sessions will also be timetabled in the conference and speakers should indicate their willingness to participate, and may like to suggest panel topics. Tutorials provide a more thorough presentation, of either a half-day or full-day duration. They may be of either a technical or management orientation.

The AUUG'2005 conference offers an unparalleled opportunity to present your ideas and experiences to an audience with a major influence on the direction of computing in Australia.

Submission Guidelines

If you are interested in submitting a paper you should send an extended abstract (1-3 pages) and a brief biography, and clearly indicate their preferred presentation format.

If submitting a tutorial proposal you should send an outline of the tutorial and a brief biography, and clearly indicate whether the tutorial is of half-day or full-day duration.

Speaker Incentives

Presenters of papers receive free registration to the conference (12–14 October), including social functions, but excluding tutorials. Tutorial presenters may select 25% of the profit of their session OR free conference registration. Past experience suggests that a successful tutorial session generate a reasonable return to the presenter.

Please note that in accordance with GST tax legislation, we will require the presentation of a tax invoice containing an ABN for your payment, or an appropriate exempting government form. If neither is provided then tax will have to be withheld from your payment.

Important Dates

Abstracts/Proposals Due:	2 June 2005
Authors notified:	3 July 2005
Final copy due:	1 August 2005
Tutorials:	9-11 October
Conference:	12-14 October 2005

Proposals should be sent to:

AUUG Inc.
PO Box 7071
Baulkham Hills BC NSW 2153
Australia

Email: `auug2005prog@auug.org.au`
Phone: 1800 625 655 or +61 2 8824 9511
Fax: +61 2 8824 9522

Please refer to the AUUG website for further information and up-to-date details: *http://www.auug.org.au/*

First Australian UNIX Developer's Symposium

Call for participation

AUUG is proud to announce the 2005 UNIX Developers' Conference, to be held in Adelaide on 3 and 4 June, 2005. Attendees will be accomplished programmers who wish to develop software for UNIX-like systems using open source tools. We are planning two concurrent streams. One stream will be aimed at programmers who have little or no experience with UNIX, and wish to learn the UNIX philosophy, environment and tools; the other will be aimed at developers who already have significant experience programming for UNIX, and wish to learn new or advanced tools and techniques.

The tutorial programme for the Newcomers stream will comprise four 90 minute presentations, which will give attendees a solid understanding of the mechanics of developing for UNIX:

- Introduction to the UNIX environment
- Shells and scripting
- Make and gcc
- Debugging with GDB

The Programme Committee invites proposals for tutorials and papers. In order to ensure complete coverage, the programme for the "newcomers" stream is a little more rigid. Our intention is to have four 1½ hour tutorials roughly covering the following topics:

- Introduction to the unix environment
- Shells and scripting
- Make and gcc
- Debugging with GDB

We invite proposals that don't disrupt this framework. For the advanced tutorials and the papers, the following list of topics is intended to illustrate the direction of the conference, but papers on other related topics will be considered:

- Introduction to the UNIX environment
- Shells and scripting languages
- Editors
- Programming languages
- Build tools (e.g. make)
- Version control
- Debugging programs
- Writing graphical programs
- Integrated development environments
- Kernel programming and debugging
- Network programming

This is an opportunity for you to help foster and strengthen the open source developers community in Australia.

Tutorials can be 90 or 180 minutes long, and papers should be 45 minutes, including time for questions. We would prefer a written paper or tutorial notes, for inclusion in the conference proceedings, for all presentations.

Submission Guidelines

Those proposing to submit papers should submit an abstract and a brief biography, and indicate whether their paper is intended for the Newcomers or the Advanced stream. Those submitting tutorial proposals should submit an outline of the tutorial and a brief biography, should indicate whether their paper is intended for the Newcomers or the Advanced stream, and should clearly indicate the duration of their presentation.

Proposals should be sent in electronic form to the Programme Committee at `developers2005@auug.org.au`.

Important Dates

Abstracts/Proposals Due	18 February 2005
Authors notified	18 March 2005
Final copy due	20 May 2005
Conference	3 and 4 June 2005

Please refer to the AUUG website for further information and up-to-date details. In particular, it is still possible that the date may change slightly.

First Digital Pest Symposium

Melbourne, 8 February 2005

The inaugural Digital Pest Symposium (DPS1) will be held in Melbourne on Tuesday, 8 February 2004.

The Digital Pest Symposium is a one-day event, focussing on the twin global scourges of "spam" email and malicious software (including, but not limited to, viruses, worms, trojans and spyware). Just about anyone who uses a computer these days (and that's most of us) has to deal with these in some way, so all are welcome and encouraged to attend, and indeed participate.

Goals

- to promote the sharing of information and experience relating to digital pest countermeasures,
- to raise awareness of the role of good software and systems design in controlling digital pests,
- to raise the awareness of open-source countermeasures within the broader ICT community.

Format

The event is made up of 45 minute talks, comprising mostly speaking time and an allowance for questions. We encourage both technical and non-technical presentations, on topics including:

- anti-virus and anti-spam solutions
- anti-virus and anti-spam software internals
- methodology—vulnerabilities exploited by spammers and virus authors
- user education
- legalities (e.g. fighting spam in the courts)
- ISP strategies
- the role of government

All speakers receive free registration

Call For Participation

Please send abstracts (around 100 words) or expressions of interest by close of business on Friday, 17 December 2004. A formal paper is desirable, but consideration will be given to presentations without a paper if of sufficient quality or interest. Likewise, shorter presentations than 45 minutes will also be considered.

Registration

Registration costs are as follows:

- $99 for full AUUG members,
- $125 for associate members, and
- $150 for non-members.

Please contact AUUG to register.

About AUUGN

AUUGN Editorial Committee

The AUUGN Editorial Committee consists of:

- Frank Crawford <frank@crawford.emu.id.au>,

  and

- Greg Lehey <Greg.Lehey@auug.org.au>.

You can always reach the current committee at auugn@auug.org.au.

Send physical mail to the following address:

AUUG Inc
PO Box 7071
Baulkham Hills BC NSW 2153

Contributors:

Thanks to the following people for contributions to this issue: Frank Crawford <frank@crawford.emu.id.au>, Greg Lehey <Greg.Lehey@auug.org.au>, Owen Mace <owen.mace@kaz-group.com>, Martin Michlmayr <tbm@cyrius.com> and Devraj Mukherjee <devraj@eternitytechnologies.com>.

Public Relations and Marketing: Elizabeth Carroll <liz@auug.org.au>

AUUGN Submission Guidelines

Submission guidelines for AUUGN contributions can be obtained from the AUUG Web site at *http://www.auug.org.au/publications/auugn/ subguide.html.*

AUUGN Back Issues

A number of back issues of AUUGN are still available. For price and availability please contact the AUUG Secretariat.

Conference Proceedings

A limited number of copies of the Conference Proceedings from previous AUUG Conferences are still available. Contact the AUUG Secretariat for details.

Mailing Lists

Direct enquiries regarding the purchase of the AUUGN mailing list to the AUUG Secretariat.

Disclaimer

Opinions expressed by the authors and reviewers are not necessarily those of AUUG Inc., its Journal, or its editorial committee.

Copyright Information

AUUGN is the journal of AUUG Inc., the organisation of UNIX™ and Open Source Professionals. Its interests include promoting knowledge and understanding of Open Systems, including, but not restricted to, the UNIX operating system, user interfaces, graphics, networking, programming and development environments and related standards.

Excerpts from John Lions' "A commentary on the Sixth Edition UNIX Operating System" by kind permission of Peter Salus.

Contribution Deadlines

Vol 25, No 4 (December 2004)	**15 November 2004**
Vol 26, No 1 (March 2005)	**15 February 2005**
Vol 26, No 2 (June 2005)	**15 May 2005**
Vol 26, No 3 (September 2005)	**15 August 2005**

Note that from January 2005, AUUGN will appear in electronic form only.

AUUG Chapter Meetings and Contact Details

City	Location	Other
Adelaide	Marcellinas Pizza Bar 273 Hindley Street Adelaide	Meetings are held at 7 pm on the second Wednesday of each month.
Brisbane	Inn on the Park 507 Coronation Drive Toowong	For further information, contact the QAUUG Executive Committee via email (`qauug-exec@auug.org.au`). The technologically deprived can contact Rick Stevenson on (07) 5578-8933.
Canberra	Australian National University	For updated information, see *http://www.canb.auug.org.au/cauug/*
Hobart	University of Tasmania	Chapter appears to be dormant. The last known URL for updated information was *http://www.tas.auug.org.au/*
Melbourne	Various. For updated information see *http://www.vic.auug.org.au/*	The meetings alternate between technical presentations in the even numbered months and purely social occasions in the odd numbered months. Some attempt is made to fit other AUUG activities into the schedule with minimum disruption.
Perth	The Victoria League 276 Onslow Road Shenton Park	For updated information, see *http://www.auug.org.au/wauug/waug.html.*
Sydney	Sun Microsystems Ground Floor, 33 Berry Street (cnr Pacific Hwy) North Sydney.	The NSW Chapter of AUUG holds meetings once a quarter in North Sydney in rooms generously provided by Sun Microsystems. More information at *http://www.auug.org.au/nswauug/.*

For up-to-date details on chapters and meetings, including those in all other Australian cities, please check the AUUG website at *http://www.auug.org.au/* or call the AUUG office on 1-800-625655.