



The **Scenic** route

Chris Bidmead is going away, and along with his shorts and sunscreen, he packs that most essential holiday essential; a laptop with Linux. Here's how he made sure it would travel well.

I'm off on holiday, so I'm in the process of refurbishing a laptop (a smart little Scenic Mobile) with an updated version of Linux to take with me: the pressure is on and I don't need amusing installation quirks at this stage. So, I turned to Caldera's latest release, OpenLinux Standard 1.1 <www.caldera.com>. The previous version went on like a dream once I'd got help via the internet with the XF86Configfile that defines the X Window settings.

It took only a couple of hours to do the install and get the new operating system bedded-in, including installing several

hundred megabytes of additional software, setting up the networking via an Acctron 2212-2 PCMCIA card and getting the thing to dual boot with Windows 95, using Partition Magic.

OpenLinux Standard isn't based on the latest version of Linux: out of the box it installs the 2.0.29 kernel, but frankly, on this holiday I need a machine that will do real work, not a bleeding-edge operating system with entertaining nuances. Even so, taking a close look at this relatively old "stable version", I realise how nicely the Linux development is coming along. The air isn't thick with marketing promises about "Linux 98"; unpaid Linux and Gnu developers around the world are just getting on with improving their software.

For example, the technique of configuring

kernel function by dynamically loading modules (rather than having to recompile) is now well advanced. The install process automatically detected my Acctron network card and loaded the PCMCIA handler, pcmcia_core, and the network modules needed to drive it. In the early days of Linux you'd either have started with an overblown kernel pre-compiled with unwanted stuff, or you'd have had to recompile to tailor the kernel to your system requirements.

Exploring modules

Okay, I know Windows 95 does all this, too, or tries to, but the thing I like about Linux is that none of this need be mysterious. If you want to explore Linux' module handling for yourself, a good way to start (well, the module HOW-TO, but that's too obvious) is to run the lsmod command. This not only shows you the modules running on your

Fig 1 Running the lsmod command

```
[root@scenic col]# lsmod
```

Module	#pages	Used by
vfat	3	1
nkfs	3	2
pcnet_cs	2	1
ds	2	[pcnet_cs] 3
i82365	3	2
pcmcia_core	7	[pcnet_cs ds i82365] 4
ipx	3	5
psaux	1	1
misc	1	[psaux] 1
ppp	5	0
slip	2	0
slhc	2	[ppp slip] 0
lp	2	0
isofs	5	0
nfs	12	6
sg	1	0
st	6	0
sr_mod	4	0
sd_mod	3	0
scsi_mod	10	[sg st sr_mod sd_mod] 4
8390	2	[pcnet_cs] 0



I'm writing this in the garden, networked to my office machines so I can run mission-critical applications like MineSweeper from my NTrigueserver (the Winmine script I knocked up to evoke it is in the window behind)

Questions & Answers

Q What is UNIX, and why should I care?

A UNIX, like any other piece of software, is just a bunch of ones and zeros. It is an operating system (nearly 30 years old) that grew out of a radical, rebellious spirit. But then it got fat, put on a suit and made a lot of money. This excites some people but probably not your average PCW reader. What is exciting is the emergence of son-of-UNIX in various guises including Linux, BSD, and Apple's imminent Rhapsody.

Q What is Apple doing here? This isn't Hands On Macintosh.

A No, but I'm talking about the underlying BSD-like operating system of NeXTStep which, with some tweaks, is set to become Apple's industrial-strength alternative to MacOS. After two massively wallet-vacuuming attempts to come up with a next-generation operating system, it has admitted defeat and fallen back on something that, like Linux, is essentially UNIX in all but name. This move of Apple's

has profound implications.

Q Steady on — it's only ones and zeros, remember...

A Yes, quite right. And that's the point. The failure here is not just Apple's. Over the past decade, Apple (IBM, Microsoft and others) got caught up in a whirlwind of escalating aspirations about what operating systems should do. This turns out to be a Bad Idea. Moore's Law says that hardware keeps getting smaller and faster but the same rules don't apply to software: software bloats. The move by Apple to UNIX signifies a return to the "keep it simple" principles on which much of UNIX is founded.

Q Er... but UNIX is very complicated.

A Windows 95 and NT are at least as complicated. The difference is that Microsoft has a mission to make software appear as simple as possible; an honourable enough goal, but hiding complexity means you sacrifice choice

and configurability. An operating environment can be bulging with function, but unless the ordinary user can get at that function and put functions together to do unique new things the designers didn't think of, you have to stick to the tramlines.

Q But Microsoft is continually producing software with more function.

A Laying more and more tramlines: "Where do you want to go today?... OK. That'll be in the next version." And because its earnings come mostly from upgrades, Microsoft has an imperative to keep us trading-up to increasingly function-rich new versions.

Q So this column is anti-Microsoft?

A No, it's pro-choice. This column isn't about being UNIX hermits: most of us have at least a bit of Microsoft in our lives and many of us run systems that dual-boot into UNIX or Windows — like the laptop I'm taking on holiday.

system, but also lists the dependencies. Fig 1 shows what mine looks like.

You can see from this that pcmcia_core is used by three modules called pcnet_cs, ds and i82365. But how are they loaded during startup? The startup files are kept under /etc in a directory called rc.d. (There are two rather different philosophies of boot design, System V and BSD. Linux uses the System V approach.) Rc.d contains a script called rc.modules. You'd need to know a lot more than I about shell scripting to understand intimately what rc.modules is doing, but it's not hard to get the general idea: it loads a bunch of modules that are listed in a file. A complete beginner might have some difficulty identifying the location and name of this file, because rc.modules sets it up with the line:

```
MODULE_LIST="/etc/modules/`uname -r` /`uname -v`$Sep$1"
```

This is an example of "command substitution" (which I touched upon last month). Right in the middle of a file path we have a command wrapped in back-quotes, so the string returned by that command becomes part of the path. But as you can see, this happens twice: once in the path and once in the file name itself. The expression finishes by throwing in a couple

of shell variables: \$Sep and \$1 (these boil down to ".default" if rc.modules is evoked with the parameter "default").

Try running **uname** from the command line, first without a parameter, then with each of these two parameters in turn. Unadorned, the command returns the string "Linux", and the parameters respectively give you something like:

```
"2.0.29"
```

```
and
```

```
"#1 Tue Feb 11 20:36:48 MET 1997"
```

I say "something like", because these strings will vary depending on the version of the operating system you're using, and its release date. As you can find out from **man uname**, the command is a GNU utility that returns information about the machine and operating system on which it is running.

The point of using these strings to name key files is to make sure they don't get evoked by a version of the operating system that can't handle them. Left to someone like me, the module list would probably be called something imaginative like "module.lst". Then when I upgrade the kernel to a new version that doesn't support the earlier modules, things would start breaking and the boot might even hang. Using command substitution like this makes

sure that rc.modules is only pulling in a module list designed for the current version.

Linux also puts all its filesystem handling into modules, and there are plenty of them, which is why it's such a good operating system in a mixed environment. As well as most of the standard UNIX flavours of filesystem, Linux can handle partitions that belong to OS/2, DOS and even the Amiga. In a dual-boot machine like the Scenic I normally establish a permanent mount so that Linux can read and write to the whole of the DOS partition, and I did this by setting up **/etc/fstab**, the file that sets out what is mounted where. But the FAT filesystem I was using truncates the name of files written by Windows 95 to 8.3 format. No matter. Load the VFAT file system with the command **insmod vfat.o** (or better still, add it to the list we discovered above, so it automatically loads at boot time). Then create a new directory, **/mnt/vfat** (say) and you can mount the Windows partition in a way that shows the full filenames:

```
mount -t vfat /dev/hda1 /mnt/vfat
```

PCW Contact

Email Chris Bidmead at unix@pcw.co.uk.



Weird and wonderful

Chris Bidmead's aim is to inform you of the many treats in Unix, so this month he returns to a Procmal poser from Peter who has discovered Popover. Plus, Bash comes to the rescue.

Peter Balfie from University College London thinks I was "very silly" to mention Procmal (*PCW*, August 1997) because "*It makes punters like me, who've never been anywhere near it, due to the huge amount of 'unixspeak', try to use it!*"

Excellent! It reminds me of the poet Christopher Logue's alleged response to that visa waiver form they hand you on aeroplanes: against the section that asks whether you're coming to the US to peddle drugs, overthrow the government or engage in acts of moral turpitude, he reportedly wrote: "Sole purpose of visit".

People are put off Unix because it looks weird, it's old, and it doesn't come wrapped in yards of marketing. The sole purpose of this column is to alert you to the fact that it's bristling with possibilities within surprisingly easy reach. What you can do with a few lines of Procmal is a case in point. Except that Peter couldn't do anything. He put the `.forward` file into his home directory, as the documentation suggested, but it seemed that Procmal wasn't being called.

The first thing is to take out the `.forward` file and send an email to yourself. If you're using sendmail on a dial-up connection you can do this while you're offline. Check this mail item has arrived and write a `.forward` line that simply names an arbitrary file:

```
/Users/bidmead/Test.forward
```

The file can be anywhere and doesn't need to even exist, but make sure it's in a writeable directory. Quote the pathname in full because sendmail's forwarding mechanism won't expand shell conventions like `~` as an alias for your home directory.

Now send a mail item to yourself again. It shouldn't arrive in your usual mailbox. If the forwarding mechanism is working it will have created a new file to receive the

message, or will append your message to an existing file of that name.

On Peter's system this didn't work, and we found that this was because he wasn't using sendmail to do SMTP (Simple Mail Transfer Protocol) but had a POP (Post Office Protocol) account. The latter is appropriate for a standalone machine that only intermittently connects to the internet, because it leaves the mail transport entirely to the remote server. POP just collects a "mail drop" at each logon.

Peter's setup was different. "*I'm on a campus POP3 mailserver for email,*" he told me, "*and I use 'Popover' run from a crontab file to pull down new messages, which end up in usr/spool/mail.*" Peter's mail-user agent picks them up from here and displays them in the usual way. What's needed here is for Popover to pass the mail directly to Procmal. Peter found the instructions for doing this in the help files for Popover.

Popover lets you name the directory in which you want mail to be deposited, but instead of a location you can use a pipe to another program, like this:

```
|/usr/local/bin/formail -e -s
Procmal
```

This puts Procmal in the loop, and now Peter's managing his mail across multiple mailboxes and finds he can also use Procmal and Formail to handle the mounds of historic mail in his mailboxes. "This is a potentially huge hidden benefit," says Peter. "My 'in' mailboxes together add up to 2,500 messages, stretching back over 17 years."

Peter is working with NeXTStep, the officially defunct Unix-based operating system set to be resurrected by Apple as Rhapsody. So his Procmal pipes its output through the NeXTStep utility, `appnmail`, rather than delivering it directly into a file or

directory. `Appnmail` transforms the mail into a form NeXTStep's MUA can handle, and also takes care of the file locking.

Last month, I teased you with the question: "Why don't my own Procmal recipes include the trailing colon that requests Procmal to manage automatic lock files?" Well, it's because in the NeXTStep environment, `appnmail` does all the lock handling for me.

While I was experimenting with the `.forward` file I discovered that you can put multiple lines in there, and sendmail's forwarding mechanism will respect all of them. A handy use for this is to create a raw backup file of all your mail before Procmal gets its hands on it. In the early days of your experiments with Procmal, this should guarantee an ability to roll back on your incoming mail in case anything goes wrong. To the single `.forward` line I suggested in the August column, add a second line mentioning a filename (Fig 1, p260).

The mail will go to both locations, and if you ever need to recover it from the file, you can use `formail` to pipe it back into your mail-handling system (Fig 2).

Batch deletion

Several of you have written in asking how to delete a set of files whose names are stored in an ASCII file. "Man `rm`" doesn't give much in the way of clues about this. You might think this is remiss of the designers of the `rm` command, but you'd be missing the point: in Unix, you combine the talents of the various utilities to achieve what you want: we want to write a command line that feeds the names into `rm`, individually.

One suggestion, from someone using the Bash shell, was to use the built-in "read" command, so, just for fun, let's explore that



Quantrix is one of a dozen NeXTStep apps for which Lighthouse is giving away two-user licences (see [this page](#))

filenames for the filenames themselves. So, instead of Fig 6 we'll write Fig 7.

The backquotes (don't confuse them with single quotes ' ') inform Bash that it should carry out the command substitution.

BT Internet revisited

Last month I relayed the saga of trying to connect a Unix workstation to BT Internet and my frustration at the refusal of BT Internet's support people to face up to the cross-platform implications of the net.

Since then, I've had a long chat with Raj Kanthan, BT Internet's product manager, who confirmed that the intention is definitely to offer a platform-independent service.

"I can see what's happening here," he says. "When you say Unix, they say: 'Oh, we only support Windows'. That's true for the client software, but it's a service that will work with any machine."

What the help desk people don't tell you is that you don't need BT Internet's Windows software to register. The company seems to have revamped its registration procedure since my early struggles, and if you can borrow somebody's internet connection from a machine running any kind of web browser, you can sign up by visiting www.btinternet.com. Raj says: "You can get on with any client and operating system that supports IP and HTML and CHAP."

Once you've registered, and at any time after that, the full platform-independent details you'll need to set up your chap-secrets file will be available as a personal web page that you can access with your password. They're also sent to you in a welcome email.

Lighthouse

Devoted NeXTStep users have just received a considerable bonanza. Lighthouse, the long-standing NeXT application provider now absorbed by Sun, is giving away free two-user licences on a dozen of its commercial-class NeXTStep applications, all of which are downloadable from www.lighthouse.com/CurrentProductVersionNumbers.html. Included is the magnificent Quantrix (pictured, above), the best multi-dimensional spreadsheet I've come across on any platform.

```
/Users/bidmead/Test2.forward " |IFS=' '&&exec
/usr/local/bin/procmail -f- ||exit 75 #bidmead" 1
```

```
formail -s /usr/lib/sendmail -t < /Users/bidmead/Test.forward ** 2
```

```
read: read [-r] [-p prompt] [-a array] [-e] [name ...] 3
```

```
bash> while read < killem ; do echo $REPLY ; done 4
```

```
bash> while read ; do echo $REPLY ; done < killem 5
```

```
bash> rm -f file1 file3 file3 file4... 6
```

```
bash> rm -f `cat killem` 7
```

possibility. Bash is available across most platforms in a GPL'd distribution and I've just installed it on my NeXTStep system in the interests of consistency.

One nice feature for beginners is the way you can pick up tips on individual aspects of the shell behaviour by typing "help" followed by the command in question: a sort of online mini-manual. For example, Fig 3 shows what "help read" reveals.

One line is read from the standard input, and the first word is assigned to the first NAME, the second word to the second NAME, and so on, with leftover words assigned to the last NAME. Only the characters found in \$IFS are recognised as word delimiters. The return code is zero unless end-of-file is encountered. If no NAMEs are supplied, the line read is stored in the REPLY variable.

To read line by line you might be tempted to turn to the Bash shell's "for" construction, but it doesn't work the way BASIC or C programmers might expect. "While...do...done" is more promising, and

you can test this non-destructively with a command line using "echo" instead of "rm" (Fig 4). But consider why this doesn't work: all that happens is the "while" loop keeps reading the first line of the "killem" file, and never exits. But why? Well, the line loops between "while" and "done" and each time it comes back to the "while", it reconsiders the proposition of opening the file called "killem" and reading it.

This is exactly what we've told it to do and is why only the first line gets read. If you want to feed the file to the loop, you have to take the file *out* of the loop (Fig 5).

This gets the job done, but in Unix there's usually a simpler method just around the corner. You can extract your filenames from the list directly with the "cat" command and what you want now is some way of laying out these filenames so that rm is able to read them. This calls for an often-used trick known as "command substitution".

We'll use rm -f to "force" deletion so we're not interrupted by any prompts, and substitute the command to create the

PCW Contact

Email Chris Bidmead at unix@pcw.co.uk.



Help or hindrance?

Our Unix chap has some trouble with CHAP and is forced to encounter the BT Internet helpline. Chris Bidmead recounts his efforts. And, caught red ANDed? A return to procmail.

UNIX is a user-friendly operating system; it's just choosy about its friends. That's often taken to be a joke but I actually think a modern UNIX implementation like Linux is more ready to yield up its secrets to the first-time user than other, more familiar operating systems. The core philosophy of UNIX is to have no secrets (from authorised users). Windows, on the other hand, positively goes out of its way to conceal what's going on behind a sometimes completely impenetrable façade of good-looking screens.

I was reminded of this when I tried to make use of the guest account offered to me by BT Internet. I wanted to connect via my NeXT machine, but BT Internet insists you use its own client software, at least for the initial registration, and, its client software only runs on... you guessed it... Microsoft Windows. (Oh, and Apple Macs.)

Regular readers of this column will remember that, last year, we went through a similar routine with another internet service provider (ISP). The ISP in question was using PAP (Password Authentication Protocol) to validate user names against passwords. But the ISP's help desk didn't understand the principle of this simple cross-platform protocol and could only talk me through the steps necessary to set it up on a Windows machine.

This involved mindlessly filling in replies to dialog boxes, which gives you no overall picture of what you're actually doing. The

data is then squirrelled away inside your system — securely? Retrievably? Capable of being backed up? Who knows? Certainly not the people on the help desk. And *voilà!* it just works. Or maybe not, in which case you will be invited to reinstall the software and start again.

It was odd to find myself going through all this again with BT Internet. Perhaps I'm naïve, but I was astonished that a company with the global reach of BT seems not to appreciate that the internet is a cross-platform interconnect, not just a way of getting Windows machines to talk to one another, or occasionally to Macs.

There was nothing for it but to log in initially from a Windows machine. I used the Siemens-Nixdorf portable, which is currently set up to dual boot into Windows 95 or Linux. I thought it would be simple, once I had registered and acquired my user ID and password, to transfer those across to my NeXT machine. Silly me.

BT Internet uses CHAP, or Challenge Handshake Authentication Protocol, to validate its users. CHAP itself wasn't the problem, as you'll see, because CHAP is a core part of PPP and any properly implemented PPP daemon ought to know how to deal with CHAP challenges. PAP works like an ordinary username-password login procedure and is therefore vulnerable to anybody who may be listening in. CHAP is more cunning. As a challenge it sends a randomly generated string of characters,

and its own hostname. If I'm trying to log in, my correct response is to return my own hostname together with — and this is the cunning part — our

agreed password specially encoded against the randomly generated string I've just received. The thing that makes CHAP particularly secure is that this challenge and response can be carried out at any time during the connect period.

You might think this all sounds complicated to implement. In Windows it was. I'm not quite sure where, when or how the CHAP stuff got set up but it was all part of a lengthy initial login and registration which, as it happened, didn't go at all smoothly. When I first installed the BT software it apparently took a look at my hard drive's dual boot arrangement, disliked what it saw and decided to fix a few bytes for me. On rebooting, the machine simply hung and refused to come back to life until I revived it with a DOS system disk. FDISK revealed that the Boot Manager partition was mangled.

OK, I fixed that, and at last I'm talking to a BT Internet web registration page that asks me my corporate ID, my personal registration code, my user ID and my mother's maiden name (yes, really!) and finally returns a number of variously mysterious strings. These are stuffed away inside Windows, so that next time I log on I'm automatically authenticated.

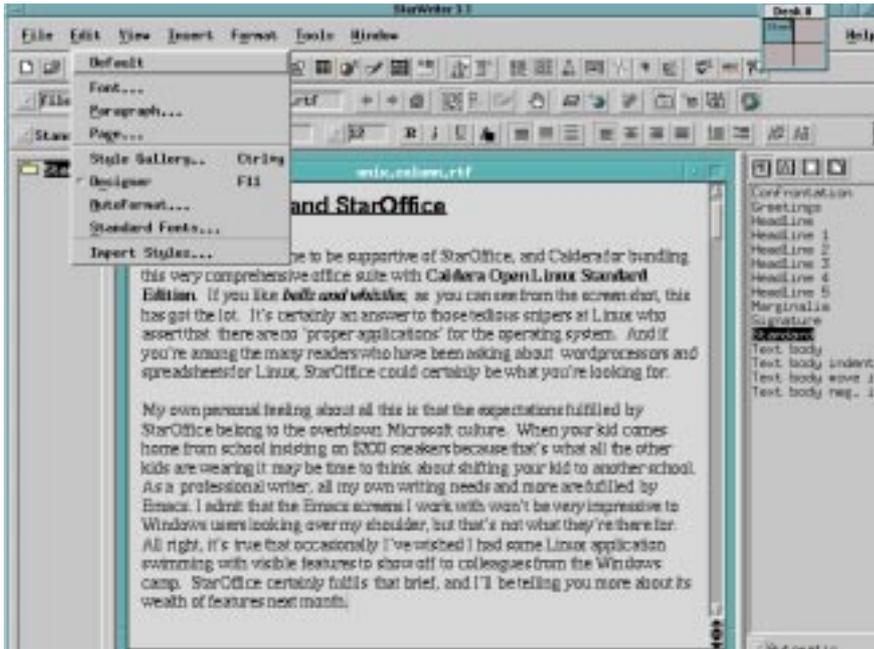
Now, to set up CHAP on the NeXT machine. Happily there's a standard UNIX way of doing this and it's simple. Operating as root, you create a file called chap-secrets, under /etc/ppp. It's a text file (so make sure that only root has read and write permission on it) and it's laid out like Fig 1.

The first line is a header (as with most UNIX config files, the # makes it a comment) just to remind me what the fields mean. There can be any number of lines below this, one for each CHAP-protected server I

p232 >

Fig 1: Creating "chap-secrets"

```
# client  server  secret  address
nextmachine  LondonT-NAS9  "Ice Cold in Alex"
```



Caldera's OpenLinux Standard version has arrived in my office. It is bundled with the StarOffice suite. This is the word processor, StarWriter. I've incorporated my own comments into this screenshot

Windows-to-Mac, service.

Careless colons in Procmail

I'm afraid I haven't left room to fulfil my promise of more on procmail this month, but I must make space here for a warning about what I wrote last month. The warning (which shouldn't be necessary for seasoned readers of this column) is RTFMP: read the flipping manual pages. Regard my remarks here as mere aperitifs to a proper digestion of the authentic facts the manual pages offer. For instance, I suggested that multiple rules in a procmail recipe are ORed. They're not. They're ANDed.

I was somewhat careless with my colons in referring to the O: flag when I meant the :O flag. Procmail wants to see :O as an indicator of the beginning of a new recipe, but that trailing colon, omitted in the recipes I quote from my own .procmailrc, is worth explaining because it has a special meaning.

In a multitasking operating system like UNIX you may have multiple instantiations of procmail running simultaneously. What happens if several procmail processes find themselves writing to the same mail file at the same time? Garbage is what happens. The classic way to avoid this is to have the first procmail process that opens the file put a lock on it. If procmail starts writing to a file called "mymail" you can have it create a temporary file called "mymail.lock". This only has meaning while procmail observes the rule that it won't try to open any file called <something> as long as there's also a file called <something>.lock.

Procmail is written to work like this if you so choose. To kick this into action all you need do is open your recipe with :O: instead of just :O. It's as simple as that. So why don't I use that trailing colon in my own .procmailrc on the NeXT machine? I'll tell you why not, and more, next month.

intend to log in to. Currently there's only one, the BT server, which returns the string "LondonT-NAS9" as its server name. I'm "nextmachine", so the first two fields should be self-explanatory. "Ice Cold in Alex" is my password, or more accurately, the secret string shared between my system and LondonT-NAS9. The address field at the end is optional — I can put the dotted quad address (of the form "192.168.1.4") of the remote server there as a belt and braces security measure, if I happen to know it. I don't, so I'll leave it blank.

There are, as it happens, several problems with this. Firstly, whatever BT Internet knows me as, it doesn't know me as "nextmachine". I need my BT username in that first field. The registration process has returned several strings that might be my username and there's no way of telling which is the correct one. I'll have to ask BT Internet's help desk about that.

The second problem is that the server only returns LondonT-NAS9 as its name the first couple of times I contact it (I can see this, because I have PPP debugging switched on to show me the exchange of strings). Subsequently it shows up as LondonT-NAS12, or some other number, indicating that I'm actually talking to several servers. A CHAP system really shouldn't do this. One solution, to keep things secure, would be to set up a separate line for each of BT Internet's various servers. So I need to ask the helpline how many I'm likely to run into, and what their names are. In the meantime, there's a useful cheat I can do. I put a "*" in that field and it will match any server name.

Field three. Another problem. Which of the many strings returned to me during the registration process is my "chap secret"? Is it the original password I chose? Or a hashed version of that password which ended up lodged in one of the Windows dialog boxes? Or (ah!) could it be the hex string labelled "password" that I discovered in a Windows config file called "pctcp.ini"? I needed to ask the BT helpline about that.

I rang the BT Internet so-called helpline with these three simple questions and they had no idea what I was talking about. They said they didn't support UNIX. I said I didn't need any UNIX support — I have a fully working UNIX system, thanks. What I needed was support for their implementation of CHAP. They didn't know anything about that. Would it help, I suggested, if I sent the questions as email? Yes, they replied, good idea. So I did. And here is their response: "Hello. I am sorry, but we only support our software on Win95, 3.1 and Apple Macs. As we do not support it we do not get trained in that area so we are unable to help you."

As it happens I have figured out how to connect to BT Internet from UNIX, but I won't give you the details here. At this stage, encouraging readers of this column to adopt BT Internet as their service provider would clearly put an intolerable burden on BT's help desk. What I will do, though, is talk to some senior people at the company and try to understand why it calls itself an ISP when all it feels qualified to offer is a Windows-to-Windows, or at best

PCW Contacts

You can email **Chris Bidmead** at unix@pcw.co.uk.

Due to space limitations, the Unix column has been reduced to two pages instead of three. Please email any comments to ellie@vnu.co.uk.



Mail order

Don't chuck your old email messages – they are a valuable information bank. If you need to keep them for future reference, Chris Bidmead explains how to do it in an orderly manner.

If your mailbox is anything like mine, it's being increasingly polluted by advertising spam. Even without this dross, you've probably found your mailbox becoming more unmanageable as the volume of traffic increases. The simple solution is to delete all incoming mail as soon as you've read it. But if you do this, you're often throwing away valuable information. I've found old email very useful as a way of tracking ideas, people and events, and with hard disk prices as low as they currently are, I like to keep everything. The trick is to organise it.

I'm assuming in what follows that you've already got a working mail system. If you haven't, check out man -k mail to discover the elements you'll need to swing into action. For most of us this will imply something like pppd to connect to the outside world, a mail transport daemon like sendmail or qmail to manage that connection, and a mail client like elm or pine to act as the user front-end.

The system I use most, on my NeXT machine, has a graphical interface called Mail.app which seamlessly handles rich media like voice and pictures. It looks exotic (Fig 1) but it's much the same Unix setup we're discussing here.

A typical mail client stores all incoming mail in a single "mailbox", actually a file, or in the case of NeXTMail, a hierarchy of directories. Most mail clients also allow you to shift email across a selection of mailboxes and you can use this to do manual sorting. Some come with a built-in filtering system allowing you to automatically direct incoming mail into different mailboxes, depending on the various criteria you set up. You may be able to achieve something similar by messing around with

your mail transport daemon, but I wouldn't recommend it. The best way is to leave the filtering to a specialist tool, as you will see.

For an overview of filtering, take a look at the Filtering Mail FAQ maintained by Nancy McGough <www.ii.com/internet/faqs/launchers/mail/filtering-faq/>. Procmal, a marvellous piece of free software conceived by Stephen R van den Berg <srb@cuci.nl> is the most powerful mail filter available, and here's a simple run-down of how I use it. The idea is to interpose procmal between the mail transport daemon and your mail agent. There's more than one way to do this, but the simplest is probably to use the mail transport system's automatic

forwarding system. With sendmail this consists of a control file called ".forward" which is kept in your home directory. One standard use of .forward is to invoke a utility called "vacation" that sends automatic replies to messages when you are on holiday, but you can also use it to redirect your incoming messages into procmal before your mail client gets to them. My .forward file contains a single line:

```
" | IFS= ' &&exec /usr/local/bin/procmal -f- | exit 75 #bidmead"
```

The line is processed by the shell and it first makes sure that the internal field separator (IFS) is a space, then executes procmal with the -f parameter (which in this

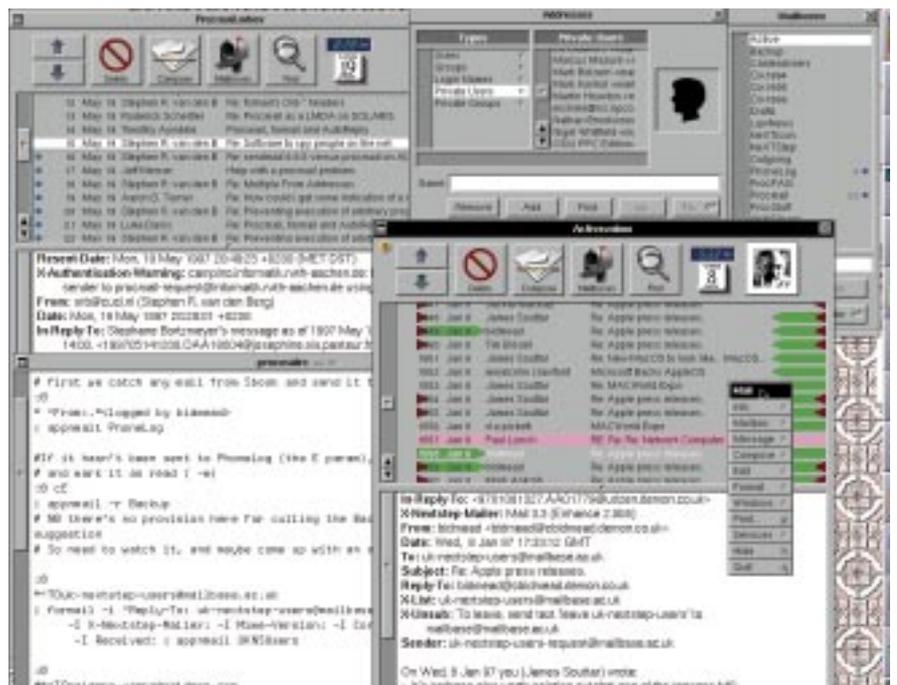


Fig 1 NeXTStep's luxurious Mail.app, augmented with some free add-ons (EnhanceMail and Colorizer) becomes a very useful information filing system. The (invisible) addition of procmal automates the sorting of incoming mail into multiple mailboxes

case effectively guarantees there's a From line in the header of the incoming mail, as required by most mail clients). The "I" that follows sets up a graceful exit if procmail isn't found. The #USERNAME comment at the end is a bit of a puzzle but the procmail manual page encourages its use as "a necessary kludge against over-optimising sendmail programs".

If the Unix-ness of this line seems somewhat overwhelming, don't worry. I copied it verbatim from the procmail manual pages and have been using it for months.

When procmail springs into action it looks for a file called .procmailrc, also in your home directory. Once you've set it up, this .procmailrc control file is the only thing you need to mess around with. The file consists of two parts. The first (which you'll probably set once and for all) establishes some environmental variables (Fig 2).

The second part comprises all the

Fig 2 Establishing environmental variables

```
PATH=/bin:/usr/bin:/usr/local/bin
MAILDIR=$HOME/Mailboxes #this directory must exist
LOGFILE=$MAILDIR/procmail.log #where procmail writes its log
```

redirection rules that you're going to set up for the various mailboxes. You can have as many or as few of these as you please and the rules can be as simple or as complicated as you like. The first simple rule I wrote for my system instantly repaid me for the work of installing procmail. It looks like Fig 3. The first line is a comment. SBook is a NeXTStep phone dialler that optionally logs phone use by sending an email for each call (Fig 4). Without a filter this is just messy; with procmail catching and filing the output, it's a joy. The second line announces that the following lines are a list of rules, or "conditions", as the manual calls them. I use a single rule here but you can have as many as you like and they will be connected by "or" operators.

The asterisk in the following line characterises it as a rule, which in this case can be read as "look for 'From:' at the beginning (^) of a header line and test whether what follows includes the string '<logged by bidmead>'". If this rule succeeds (and only if) the action line (the first of the following lines that don't begin with *) is carried out. In this example, the action we

Fig 3 A simple redirection rule

```
# catch any mail from SBook and send it to the PhoneLog :0
* ^From:.*<logged by bidmead>
| appnmail PhoneLog
```

Fig 4 SBook call logging

An SBook call logging email looks like this:

```
Date: Tue, 22 Apr 97 16:52:09 +0100
From: Mike Pritchard, Euromation <logged by bidmead>
To: bidmead
Subject: Call received from Mike Pritchard, Euromation
Call received from Mike Pritchard, Euromation at Tue Jul 22 16:51:16
1997
Mike says that Euromation will be delighted to fly me to Cannes at
their expense if I promise to write a glowing article about their
products. I graciously declined.
Duration: 04:55
```

The simple "To:" line (omitting my full internet address) ensures that the email stays local and isn't relayed out through my service provider. The "From:" line shows the (alas, in this case, entirely fictional) originator of the call. The timeline is added automatically by SBook and the note that follows is made by me at the time of the call.

To make sure this is caught by procmail and sent to SBook's own mailbox, I need to look for a feature in the headers that is unique to an SBook-originated email. This turns out to be the "<logged by bidmead>" part of the "From:" line.

trigger is to pipe the output from procmail, the message in question, through a utility called appnmail into the appropriate mailbox. Appnmail is a special NeXTStep

tool created by NeXT guru Carl Edman specifically to link NeXT's mail client,

Mail.app, with procmail. It will create the target mailbox if it doesn't already exist and will parse any multimedia content the message may hold, preparing it to be read

by NeXT's Mail.app. In a non-NeXTStep system you would omit

the "I appnmail" and just supply the name of the target mailbox. But as the appnmail example illustrates, a power user could use the action line to carry out any number of activities. The whole block, condition plus action, is called a "recipe". Recipes can be clever without being much more complex:

```
:0 c
| appnmail -r Backup
```

Here, we're taking the elementary precaution of backing up all incoming mail that hasn't come via SBook (which has already been diverted), so we send it to a mailbox called Backup.

As we saw with the SBook recipe, processed mail normally won't fall through to the next recipe. But this time, we don't want to intercept the mail, just make a copy of it. This is achieved by the "c" parameter that follows the "O:" flag. (Appnmail, which by default marks the mail it handles as unread, needs a -r parameter so that the copied mail in Backup doesn't continue to clamour for our attention.) With just those two recipes in place, all my incoming mail is copied to Backup and falls through to my standard mailbox. Now it's safe to experiment by appending some new

```
:0*^TOcaldera-users
| formail \
-I X-Nextstep-Mailer: -I Mime-Version: -I Content-Type: \
-I Received: | appnmail CalderaUsers
```

recipes. Let's capture all that useful traffic from the caldera-users' mail list [above]. The recipe introduces new ideas which we'll look at next month. Meanwhile, you can join the procmail mail list by mailing procmail-request@Informatik.RWTH-Aachen.DE with the word "subscribe" in the header. It's busy, so you might need this:

```
:0*^TO .*procmail@Informatik.RWTH-Aachen.DE
Procmail
# or, if you're using NeXTStep.
# | appnmail Procmail
```

PCW Contacts

You can email **Chris Bidmead** at unix@pcw.co.uk.

Due to space limitations, the Unix column has been reduced to two pages instead of three. Please email your comments to ellie@vnu.co.uk.



/bin there, done that

Chris Bidmead throws light on Unix directories — /bin, /dev, etc. It's an informal layout, but you can get a feel for how it's put together. Plus, there's ping, SCO, and the great awk.

Last month I described how the XFree86 configuration keeps its information in a file called XF86Config, which is usually to be found in the /etc directory. When you're just getting started with UNIX, finding your way around the various directories can be confusing. A number of them have been set aside, by convention, for more or less specific uses, but the catch is that the set of directories and their uses tend to vary between different flavours of Unix. So, as for much else in this column, what follows isn't prescriptive, but is to help you get a handle on how it works.

- Expect to find the standard binary files intimately associated with the operating system itself in /bin.
- Devices (pseudo files which are actually connections to device drivers) are in /dev.
- /tmp is reserved for temporary files.
- The /usr directory will typically contain nothing but a number of other directories, like /usr/sbin, for binaries used for system administration, and /usr/bin, for frequently used binaries that aren't already in /bin.

The distinction between /bin and /usr/bin is fairly arbitrary. You might think that putting them under /usr implies they are the operating system enhancing binaries added by users, but these belong more properly (on many systems) in /usr/local/bin.

The /etc directory sounds as if it might be for everything else, but it, too, has a (roughly) defined role. These days the /etc location tends to be the place where all the configuration files are kept. But for historical reasons, /etc is also where you'll find networking utilities like ping and ifconfig. Oh, and the executable init scripts are probably in there, too. Yes, it's a bit of a mess, but it's a widely understood mess!

More recently, there's been a drive to formalise this directory tree more strictly, but in the meantime hold on to the idea that there is at least some rhyme and reason behind the layout. You get a feel for where the various files are lurking, and you can always supplement this with the system search utilities /bin/find and /usr/sbin/whereis. Or is that /usr/bin/find and /bin/whereis?

Sign of the times

I may not get the chance to reply to all your email (I do try) but feedback from you is definitely the making of this column, so please keep it coming.

There is, however, one kind of email I really don't like getting. No, I'm not talking about abusive email (rare, I'm glad to say) but the all-too-common mailing that expects me to do all the work and doesn't give me anything to go on.

Here's an example: *"I'm trying to connect my new Windows 95 PC to our IBM AIX database server for client-server access. The IBM is expecting a TCP/IP connection to service the client-server request. I've set up my PC with Microsoft's TCP/IP as one of its protocols (using Network Neighbourhood Add & Properties). What do I need to do now to establish a connection?"*

My first question is, who are you? The email isn't even signed. I had to fish about in the headers to discover the name, Ian MacDonald. What does it cost to spend a line or two introducing yourself? For example, "Hi, Chris, I'm a nuclear physicist at Cern, and we're using FreeBSD to control our particle accelerator. I happened to pick up a copy of PCW..."

Another point that is almost mandatory in mailings, not just to me but to

newsgroups or whoever (and we talked a bit about this last month) is to indicate that you've gone some way towards trying to solve the problem yourself and you're not just leaning back, expecting someone else to do all the work. Even if you were, you'd need to define the problem rather better than Ian does here: *"The IBM is expecting a TCP/IP connection..."* doesn't help much. If there is such a connection and nothing's working, you have one set of problems. If you haven't managed to make the TCP/IP connection at all, that's a different problem.

The power of ping

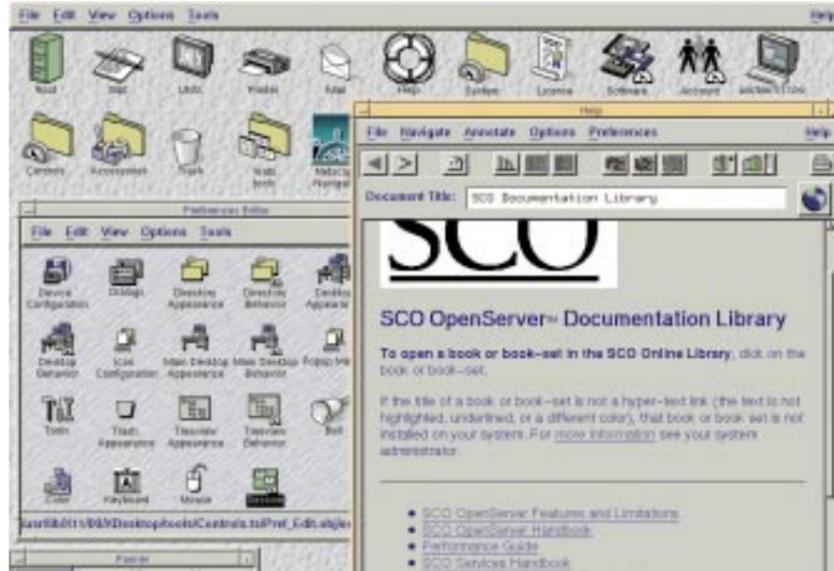
The way you test whether a simple TCP/IP connection exists is to access the AIX machine's IP address from the client machine using the elementary utility called "ping" which should be available on any TCP/IP setup, and is certainly there in Windows 95. Now, I acknowledge there's a real problem for Windows 95 users: if they don't happen to know about the existence of ping (and why should they?) it's very hard to find out about it. Despite the extensive so-called "Help" Microsoft offers, there's no way (that I can see) for the complete novice to get from "network" to "ping". Contrast this with any decent UNIX operating system, where entering "apropos network" at the command line will return a list of associated commands: a little research among these will lead you to ping.

So ping may be all it takes to diagnose Ian's problem, or it may not. Who could possibly tell on the basis of the skeletal information provided?

SCO — sorted

By way of contrast, here's the sort of email I much prefer: *"I read your column regarding*

p268 >



Paul Rowlands' email mentions the HTML documentation that comes with SCO Open Server. Here's what it looks like. The help screen is a simple web browser (I've added Netscape 2.0.2 to my own Open Server setup, downloadable from the SCO web site) and the underlying HTTP server seems to be Apache, so this is more or less the same setup you get with Linux distributions these days

the problems you were having running SCO and DOS on the same (Apricot) system: seems like you had more trouble than I did ... I have SCO OpenServer 5 (free), NT4, and DOS 6/Windows 3.11 all on the same system ... My only problems are:

1. When I boot, I have to use fdisk to set either the Unix or the two Microsoft partitions active. The NT loader doesn't want to know about Unix.
2. SCO doesn't support my graphics card (a Number 9 Reality 332) so I have to run it in 640 x 480 mode — a bit grim, since it can support 1024 x 768 x 32K colours under NT and Windows...

"Another gripe about SCO is that it seems impossible to feed back problems or get support. I know the product is free (but for a 'small handling charge'), but what is it there for? Is it to encourage me to buy it in my professional capacity?

"Have you noticed that SCO comes with an HTTP server already configured and running (dunno what particular flavour it is, but it contains O/S documentation)? Anyway, all the best, keep it up — we need a non-Microsoft perspective on the world."

Paul Rowlands

Paul also provided details of his hardware and made a few other useful points about SCO, which I've omitted for the sake of brevity. This is the kind of mail to which I can respond. I suppose the most

serious point is the one about SCO's unapproachability. The good news is that this shouldn't be the case for readers of this column because Dave Gurr, marketing development manager at SCO UK, has been very helpful to me over setting up Open Server, and he's allowed me to publish his phone number and email address (see "PCW Contacts", p269) for any of you who need further help.

Please remember that the same rules (about which I've been writing) apply here, too: do your homework first. You'll find help about Open Server installation at www.sco.com, so please investigate here before contacting Dave.

As luck would have it, several other readers have written in with help for the problem about needing fdisk to switch between bootable partitions. My apologies for not having spotted this myself, but SCO has actually included some elementary dual boot software in the distribution. Robert Warner robert@softdesign.demon.co.uk discovered this when setting up a disaster recovery server that used two partitions, one for SCO and one for Windows 95.

He says: "I installed SCO first, having the usual drivers problem because I was using a non-Adaptec SCSI controller..."

(Minor technical carp, Robert. These aren't controllers, they're host bus adapters. With SCSI, the controller is embedded into the drive device itself.)

"...Once I had that up and running, I proceeded to install Win95. This all worked fine except that when I booted the PC, it went straight into Win95 instead of getting the SCO boot prompt..."

This is because Windows 95 assumes that it is the only operating system on the disk, and on install sets its own partition as the active partition. Robert fixed this using fdisk to make the SCO partition active. This gives you the SCO Unix boot prompt when you switch on, and if you just hit return at this point you'll be booted into Unix.

"But," writes Robert, "if you enter 'dos' and press return, it will boot from the Win95 partition. You get the best of both worlds."

That weasel word, "Open", again

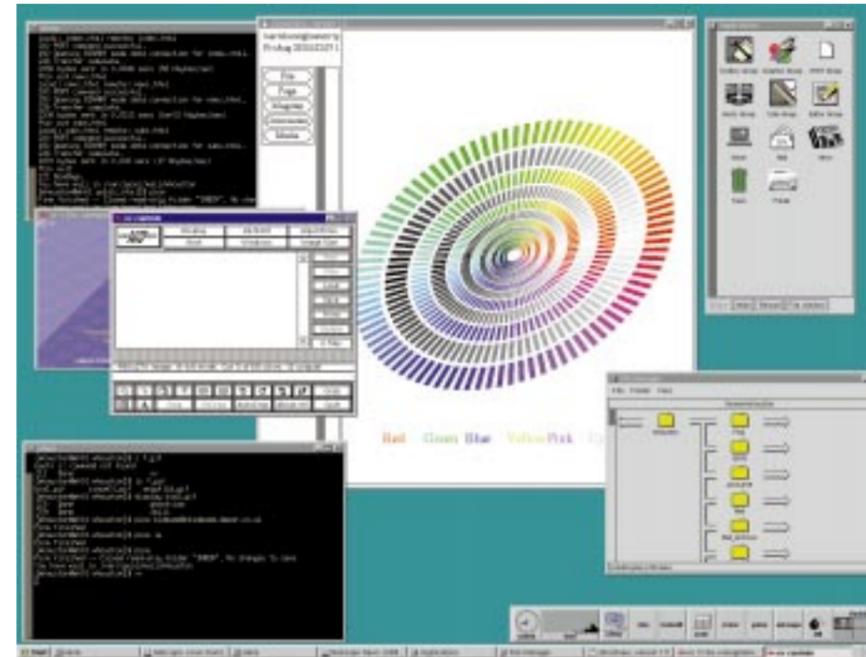
A couple of months ago I quoted SCO's Michael Tilson, who was complaining that: "The term 'open' has been hijacked. Because 'open' is good, everyone labels whatever they sell as 'open'."

It's a point I had raised with Caldera when it first came out with OpenLinux™ at the beginning of this year. But more recently, I found myself becoming increasingly irritated by this new development because of a supposed Caldera product called "OpenLinux Lite".

My initial objection to "OpenLinux" was that it capitalises on an established (non-proprietary) brand, but somehow manages to suggest that it is more "open" than Linux. Depending on your interpretation of the word "open" you might not accept this. A plain Linux implementation like the RedHat one, on the PCW CD a couple of months ago, can be shared and copied without restriction; Caldera OpenLinux can't. So which is the more "open"?

Well, the argument goes: Open with a capital "O" refers to adherence to standards and is not about whether or not you get the source code, or a reference to unrestricted redeployment. The new Caldera distribution is being aimed towards full UNIX compliance according to the specifications laid down by The Open Group. This is something most Linux users don't care about, so in this sense their plain old Linux is arguably less "Open" than Caldera's.

The OpenLinux Base product that currently sells for about £55 includes the Looking Glass desktop, the Netscape Navigator 2.02 web browser, Metro-X's proprietary accelerated X-Window system, the CrispLite text editor and other stuff. The version stripped of all this, and so freely



This screenshot, supplied by Martin Houston, organiser of the Linux branch of the UK UNIX User group (www.mh01.demon.co.uk) shows RedHat Linux running the latest fvwm97 window manager (the resemblance to windows from another well-known operating system is not accidental). You'll find a number of optional Linux interfaces at www.PLiG.org/xwinman/

distributable is, as I understand it, to be called "OpenLinux Lite". This seems to me to imply that it's somehow a cut-down version of Linux, which it isn't.

Perhaps a more serious objection to OpenLinux Lite is that despite promises made at the beginning of this year, no mention of it has yet appeared on Caldera's web page and it doesn't seem to be obtainable (although you can download OpenDOS, Caldera's own version of DOS, bought in from Novell). I hope this will have changed by the time you read this. You can check www.caldera.com to find out, and if it still isn't there you can write to nancy.pomeroy@caldera.com to ask why not.

By way of contrast, I notice SCO has made good its promise to release a free single-user version of UnixWare on much the same basis as its currently very popular (to judge from your emails) and free Open Server. Find details of both operating systems at www.sco.com/offers/index.htm.

Why two operating systems? SCO is the largest commercial vendor of Intel-based UNIX systems, and when Novell's efforts to make a decent business out of UNIX fell apart a couple of years ago, SCO stepped in towards the end of '95 and bought UNIX from them. (At a bargain price, by the way: SCO needed to bring its own ageing version of UNIX up to date and paid Novell \$60m for something that Novell had paid AT&T \$320m for two years earlier!)

SCO's intention is to combine its Open Server with Novell's UnixWare, and the result of this will be appearing later this year as "Gemini". The free single-user version of UnixWare looks like something worth investigating. I've put in my own request for a copy, so expect news in this column soon.

More adventures with awk

As regular readers know, I like "awk". It's a good, simple, general-purpose language and much better than BASIC, in my view, as a way to get started in programming. It also comes free, in the shape of GNU's "gawk". The awk examples we've had in this column in the past have always required you to evoke gawk explicitly and add the program file and datafile as parameters, like this:

```
gawk -f progfile datafile
```

Recently, I decided I wanted to write an awk program as a utility that you can run directly as an executable in its own right. With shell files (written in csh, bash or whatever) the trick is to use what's called the "shebang notation". You do this by adding a magical first line to your script that tells the system where to find the language file that's going to run it.

```
#!/usr/bin/csh
```

The "#" (pronounced "hash" or "sharp") indicates that the line is a comment, and the immediately following "!" (or "bang") triggers the system magic. Most modern UNIX systems support this.

You can do the same thing with an awk script but what happens next isn't what you want: gawk loads and then tries to treat your script as if it were the data file. The trick (not immediately obvious from any of the documentation) is to use the -f indicator, just as you would at the command line.

Here's a skeletal example that also collects parameters from the command line:

```
#!/usr/local/bin/gawk -f
```

```
# How to pass command line params into gawk using ARGV, ARGC
```

```
# The ARGV array starts at 0, but that value will be "gawk"
# So we start at ARGV[1]
```

```
BEGIN {for (x = 1; x < ARGV; ++x)
        print ARGV[x]
        print "There were " ARGV - 1 " arguments,
not counting ARGV[0]"
}
```

There's an amusing catch to this, though, if you're also trying to pass one or more data files into gawk from the command line, and I'll deal with that next month.

PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. Email him at unix@pcw.co.uk.

SCO UK, Dave Gurr 01923 813548; email davidgu@sco.com



The joy of X

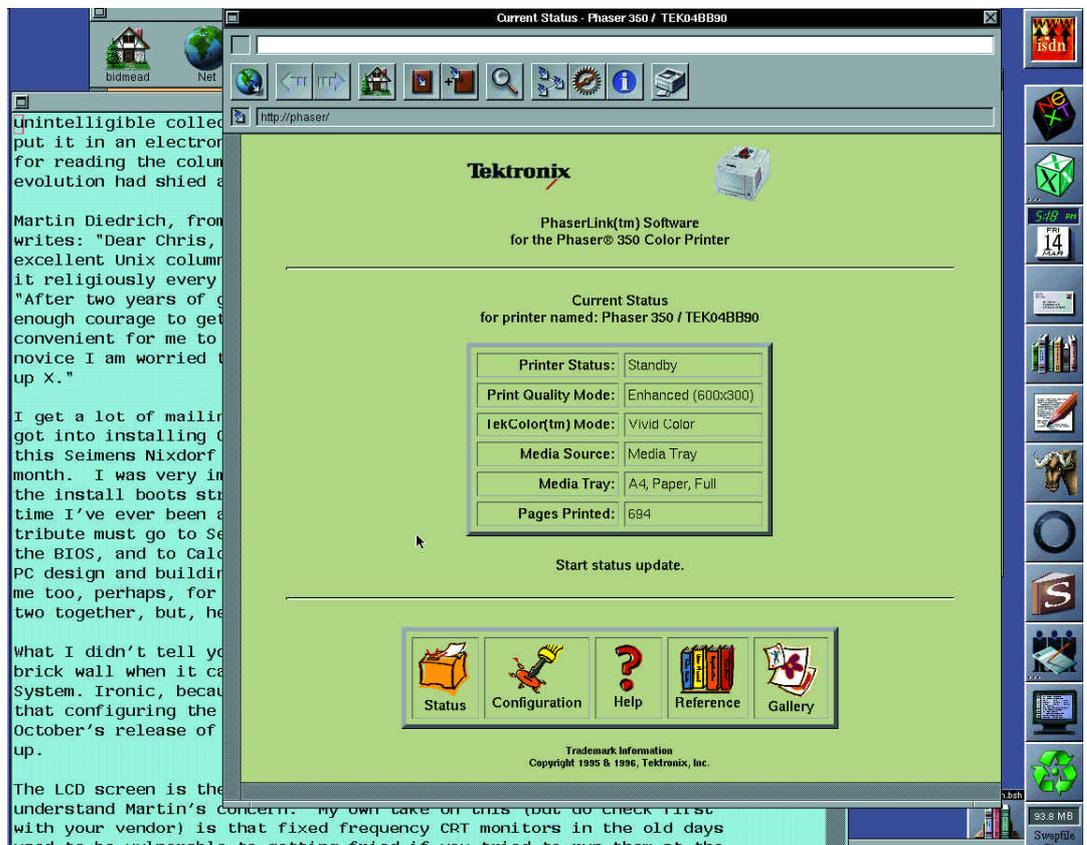
Chris Bidmead tells how he got X up and running on his Scenic Mobile 700 notebook, accepts praise on his technique, and gets down to some heavy vetting of Linux books.

Of course, it may be that you've all been well brought up, but a great deal of the email I get these days begins with burbles of praise for this column. And a frightening number of you are also writing in to tell me you've been encouraged by my ramblings here to junk Microsoft Windows and install one or other of the UNIXes, or at least create a dual boot system. For once, words fail me. I don't know how to express the sense of excitement, trepidation, pride and responsibility this gives me.

On the other side of the coin are the people who tell me that Windows is "good enough" and "if it ain't broke, don't fix it." I'm sympathetic about this too. Microsoft has done

a remarkable job of making computing accessible to everybody. And if by contrast you're left with the impression that UNIX is an unintelligible collection of arcane keystrokes — as somebody recently put it in an electronic conference — I do see your point (and thanks for reading the column this far). My response to this was: if evolution had shied at arcane keystrokes, we'd all still be rocks.

Martin Diedrich, from the Department of



Cross-platform interoperability isn't just about workstations and servers. I really like the idea of being able to control the network printer from any workstation on the network — security permitting. The Tektronix Phaser 350 has its own built-in web server, so any workstation running a browser can read its status and reset its parameters

Economics at Keele University writes: "Dear Chris, First of all, my congratulations on your excellent Unix column in PCW." (See, I'm not making this up!) "After two years of growing interest in Linux, I have finally gathered enough courage to get started... For various reasons it will be convenient for me to install Linux on a laptop... but being a Unix novice I am worried that I might do damage to my screen when setting up X."

I get a lot of mailings asking about UNIX

and laptops, which is how I got into installing Caldera's latest release, Caldera Open Linux, onto a Siemens Nixdorf Scenic Mobile 700 portable, as I mentioned last month. I was very impressed — and I hope you were too — by the way the install boots straight off the Caldera CD-ROM. It's the first time I've ever been able to do this with a PC-type machine, and tribute must go to Siemens Nixdorf for implementing this feature into the BIOS, and to Caldera for

CBOR — Chunky books ooze reassurance

The day that Windows NT finally established itself, I remember thinking at the time, was the day that the Windows NT Resource Kit arrived — three chunky volumes accompanied by a CD-ROM. It doesn't matter what my views of Windows NT are, I recall thinking, or what I know about Microsoft's support for its new baby ("Problems? Have you tried rebooting? Ah, OK, then the best thing is to reinstall..."), or indeed whether Windows NT fulfils its promises or not. The three chunky volumes ooze the kind of reassurance that is exactly what corporate customers need with a product like this. Microsoft knew this, of course, which is why the Microsoft Press produced them.

At the time I never believed that even its greatest fans could feel the same way about a freely distributable operating system like Linux. Sure, there's a ton of documentation out there on the web or buried inside the installation CDs. But these are nerdy monographs with spellings like "kernal", not glossy volumes that sit on your bookshelf glowing with confidence.

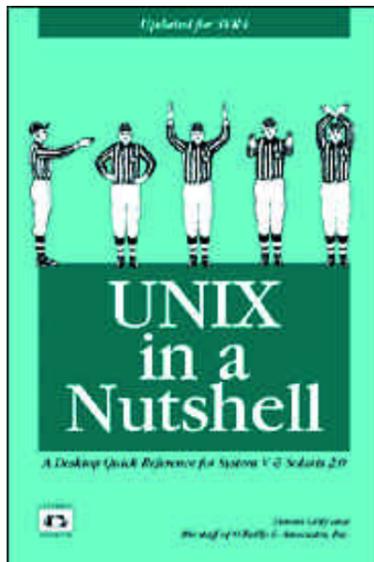
All that has completely changed now. Solid, informative books about Linux abound. When readers ask me, what books should I buy to get started, I still respond that the best way in is probably to hold off from buying books and get stuck in to the docs on the disks. But once you're through that, or if you really don't like reading on-screen, the book scene waiting for you is a toothsome banquet. Which ones to choose?

The starting point is *The Unix Philosophy* by Mike Gancarz, from the Digital Press, which asks — and answers — the rarely faced question "Why UNIX?". When it comes to "How UNIX?" my favourite was always *Running Linux* from O'Reilly, and there's a new edition out now. Supplement this with *Linux Network Administrator's Guide* if you're going to get hairy with networks. This too is published by O'Reilly, but it's also part of the Linux Documentation Project so you can download it from the web or consult it on-line as necessary.

But the must-have book for me has always been *UNIX in a Nutshell* (yes, that's from O'Reilly again!). It covers all the basic commands but manages to be more than just a command dictionary, finding plenty of room for worked examples and illuminating commentary.

There are several versions of *UNIX in a Nutshell* for different flavours of UNIX. I favoured the System V version but it had surprising (to me) omissions. When I was first struggling with the mount command I was alarmed to find no mention of it in the book. A seasoned UNIX jock patiently explained to me that this was because it was a user's manual, and mere users had no business messing with system commands like mount.

Well, there's now a (rather fatter) version of the Nutshell book specially for Linux, and, yes, mount is in there. This new version recognises that most Linux users are also going to be their own system administrators, so there's now a complete section on System and Network Administration at the back of the book. There's also a rather breathy introduction with headings like "The Excitement of Linux" which, while not inaccurate (sample: "Linux revives the grand creativity and the community of sharing that UNIX was long known for...") may help to obscure the point that, largely thanks to the weighty endorsement of books like this, Linux is clearly ready for prime time.



chat with colleagues and some recent experiments of my own, is that you just don't get to fry a modern LCD display with mere software.

Although X wasn't working on the Scenic, at least Linux was all in place. If the character-based consoles are as far as you're ever able to get with a particular UNIX installation, please don't despair. For the first four months after I introduced Linux to this column a couple of years ago, I couldn't get X working. That Linux installation wasn't pretty, but it was still powerful. I got a lot of things done with it and, thanks to all the loose documentation, the manual pages and the built-in Info hypertext system, there was plenty to keep me busy and keep me learning until I worked out how to get X going.

This time I didn't have to wait that long. X is now up and running on the Scenic (I'm writing this using XEmacs on the machine), and I'll tell you how I did it. Originally I anticipated filling the next 50 paragraphs with a detailed technical description containing a lot of example data like

```
640x480 @ 60 Hz, 31.5 kHz hsync
Modeline "640x480" 25.175
640 664 760 800 480 491
493 525
```

with some heavy discussions about dot clocks and horizontal and vertical sync frequencies. This column isn't afraid to venture into tough territory, but Bidmead's Law of Hard Work states: don't do it if you don't have to. Some of you may regard this Chronicle of How I Got X Going as something of a cheat. So be it. This is how it goes with Linux in real life.

Step one was to ignore any possible difficulties and just go for it. Caldera Open Linux comes with a pair of alternative X systems, the freely distributable XFree86 version and a commercial package called Metro-X. I'm dedicated to the cause of free software, but I also like an easy life, so given the choice I started with Metro-X. Alas, it turned out not to include any support for the Scenic's CT65550 graphics chip. Happily the XFree86 version does; as is often the case, the free software is ahead of the commercial equivalent, and this implementation had the CT65550 covered.

But there's more to a video subsystem than the graphics chip. Other key factors are the RAMDAC and the display. In the absence of any specifics on these I ignored the possible problems and sailed into the XFree86 graphical setup routine I

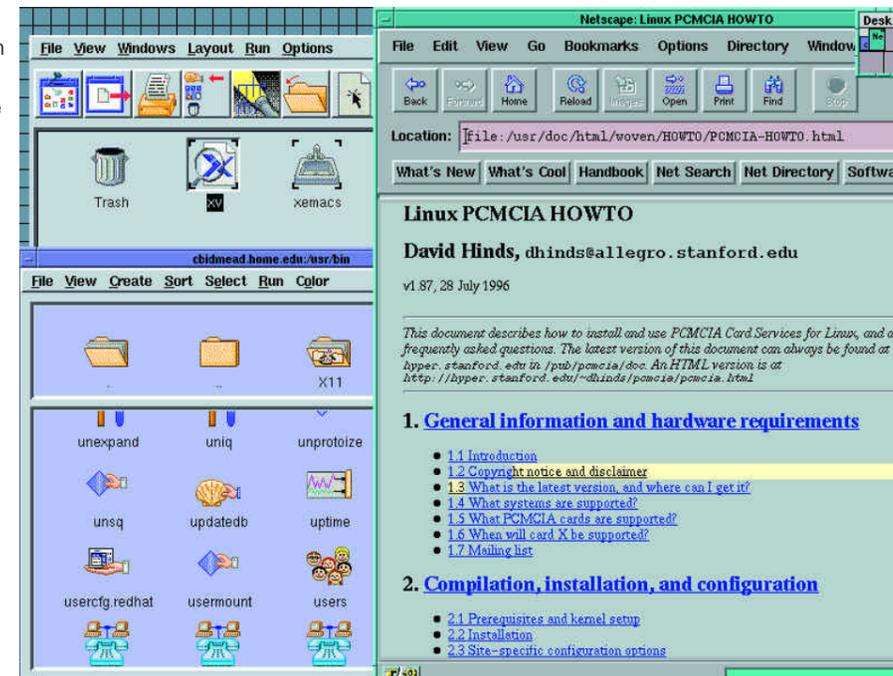
mentioned last month. It's called XF86Setup and it tries to establish an elementary X display straightaway, and then invites you to fiddle with the parameters through a dialog box with buttons and pull-down menus.

The X screen it came up with was about two thirds the size of the Scenic's LCD, which I could live with temporarily. But it was defaulting to the wrong mouse, which meant I couldn't pick my options by point and click. Happily, the keyboard navigation that XF86Setup offers as an alternative worked, somewhat awkwardly, and once I'd fixed the mouse (the touch-sensitive pad the Scenic uses is, conveniently, PS/2-compatible), it was a lot simpler.

With the majority of desktop machines, XF86Setup would probably get you all the way home. But notebook computers tend to be weird. The Linux Laptop Home Page at www.cs.utexas.edu/users/kharker/linux-laptop (or the RedHat mirror at www.redhat.com/linux-info/laptop) is run by Kenneth E Harker, who seems to be working very hard to keep it up to date. It covers a number of the popular machines, but the Scenic Mobile was too new to be on it. I was on my own, and, as it turned out after several hours with XF86Setup, on my own with a display that determinedly remained two-thirds of the size and fizzed a lot every time I wiggled the mouse.

The XF86Setup utility is really just a pretty front-end to a configuration file called XF86Config that sits (usually) in the /etc directory, which is the canonical place for these kind of files. So my next step on the road to The Joy of X was get out of XF86Setup and start mulling over the config file directly. Like all good UNIX config files, this one is in plain ASCII, editable by any text editor provided you're a user with read/write access to /etc, which on most systems implies you're root. Root is always presumed to know what s/he's doing, so the fact that you can dramatically mess up the entire system by tweaking these /etc config files isn't supposed to be a problem. My consolation was that messing up XF86Config could at worst only deprive me of my fizzing, shrivelled X display and leave me at the command line. From there I could at least restore a backup of XF86Config.

Past experience on other Linux systems



Here's another increasingly common use for web browsers — as readers for internal documentation.

The Caldera OpenLinux Base I've installed on the Scenic Mobile comes complete with a large set of HOW-TOs and other documentation set up for easy-on-the-eye reading through the bundled Netscape browser, so I can read up on Linux wherever I am

has taught me that the supplied autoconfig utilities like XF86Setup tend to be a lot smarter than I am. So if they couldn't produce a decent display, I had a lot of experimentation and twiddling ahead of me. I accordingly armed myself with "The Hitchhiker's Guide to X386/XFree86 Video Timing (or, Tweaking your Monitor for Fun and Profit)" by Eric Raymond et al, and those of you who are disappointed that I'm not going to parade the guts of video tweaking in this month's column had better hasten onto the net to procure it, if it's not already in your X11R6/lib/doc/ directory. Speaking of the net, the other (and really smart) thing I did was to put out a Mayday call. Not on the newsgroups, although I did first scour comp.os.linux.x and comp.os.linux.setup to see if this ground had been covered. (An excellent way to do this is to run a search from www.dejanews.com.) Instead I went to the Siemens Nixdorf web site at www.sni.de and found a discussion group set up there for problem logging. The net is full of so-called problem reports that just say something like "I'm having terrible trouble getting X to work on my laptop. Please can anybody help?", so I took the trouble to describe the problem, specifying the model number of the machine and the version of

Linux and XFree86 I was trying to set up.

The response from the Siemens Nixdorf engineers the next day wasn't wildly helpful: it just suggested I contact the UK help desk. From there I learnt that Siemens Nixdorf doesn't support Linux, but at least the help-desk guy gave me some pointers to existing help on the web. Following these up resolved down to "The Hitchhiker's Guide to...etc" so things seemed to be going round in circles.

I was settling down with the Guide when another bit of mail dropped into my mailbox. It came from Heiko Boch, a German computer science student at the Technische Hochschule in Darmstadt, Germany. Heiko had seen my *cri de coeur* in the Siemens Nixdorf discussion group, and was happy to step in and help. His mailing included a ready-to-go copy of XF86Config he'd hand tailored for the Scenic Mobile.

Thanks, Heiko. That's real-life Linux — people turn up and help. Now if I can just get this network card working...

PCW Contact

Chris Bidmead is a consultant and commentator on advanced technology. Email him at unix@pcw.co.uk.

recognising this important new trend in PC design and building their CD to be bootable. (A bit of a tribute to me too, perhaps, for reading the relevant manuals and putting two and two together.)

What I didn't tell you last month was that the installation hit a brick wall when it came to configuring the XFree86 X Window System. Ironic, because I had introduced the subject by saying that configuring the X server had got a lot easier since last

October's release of XFree86 3.2 with its easy-to-use graphical setup.

The LCD screen is the single most costly component in a laptop, so I understand Martin's concern. My own take on this (but do check first with your vendor) is that fixed frequency CRT monitors in the old days used to be vulnerable to getting fried if you tried to run them at the wrong frequency. I've never come across this with a modern monitor, and my impression, reinforced by a



Top hat time

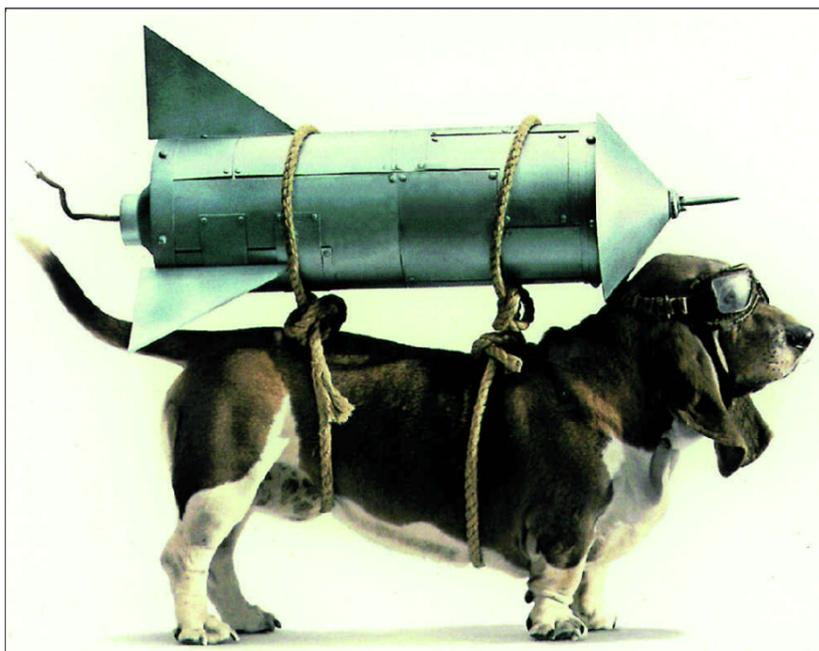
Chris Bidmead looks at RedHat and OpenLinux Base, and installation on a notebook computer using XFree86 3.2. There's the latest RedHat Linux for you on our CD-ROM, too.

One of the frequent themes of your emails to me is the request for Linux on the "front cover". This month you're in luck, because the very latest version of RedHat Linux is there, on our cover-mounted CD-ROM, ready for you to install.

Meanwhile, devotees of other UNIX-like operating systems want to know why I am forever harping on about Linux, at the expense of, say, FreeBSD? No doubt some of you will regard the inclusion of RedHat on our CD-ROM as not so much a positive blow for Free Software or the UNIX life-style, as a kick in the teeth for FreeBSD or NetBSD.

Operating-system wars are not very interesting. My network sports a number of different operating systems: Linux, NeXTStep, AIX, SCO and (as a link into the Microsoft world) Windows NT. And that happens to be just about as far as I can stretch it for the moment. I've nothing against the BSDs, or other UNIXes, and it's always good to hear from readers who have positive and interesting things to say about them. This column is certainly not written on the assumption that you're all running Linux, or that Linux is all you want to read about. Linux, for me, is just one instantiation of a UNIX-like operating system, chosen because it does make a pretty good reference point: it's probably the easiest for anyone to get hold of and it runs on the widest range of software.

To get a broad picture of the free UNIX products available, you might visit www.public.iastate.edu/~free-unix/homepage.html or www.ici.net/faq/unix-for-pc.html. And the free version of SCO on www.sco.com is proving popular, too.



Perhaps not a wildly informative screenshot, but I fell for this picture when UNIX workstation vendor, Silicon Graphics, used it to explain why its own new Octane graphics workstations (starting price \$25,000) are in a different league from anything else you could put together around a PC. This is supposed to represent a PC with a whizzy graphics card installed

There are also many readers who turn to this column to expand their horizons without necessarily wanting to get involved in installing any of this stuff. That, too, is fine by me. If you're among this group, please don't junk the CD-ROM: there is a lot of really useful documentation on it and Martin Houston, the Linux guru who helped put it all together, has fixed it so that you can read most of it from the comfort of your Windows installation, using a web browser.

Probably the best thing on the CD is Matt Welsh's *Linux Installation and Getting Started* book which you'll find on the disc, complete and in HTML format under the

`/instguid/` directory. Load the file called `gs.htm` into your browser and you're on your way. This is the way to read it from DOS or Windows, because this copy has been doctored to work with DOS-foreshortened filenames. If you're already running Linux or another UNIX, use the copy at [/doc/HTML/ldp/install-guide-2.2.2.html/gs.html](http://doc/HTML/ldp/install-guide-2.2.2.html/gs.html).

Be aware that Matt's book is a general guide to Linux. For specific details of how to install the Linux from the CD, turn to the instructions that Martin Houston has provided in the folder called "linuxxt". The instructions assume that you'll probably

want to preserve your existing DOS or Windows installation and will show you how to create an additional partition using only existing DOS software and the `fips` DOS utility provided on the CD-ROM. As Martin says, please do read the documentation thoroughly before you proceed, and back up your data first.

RedHat and Caldera

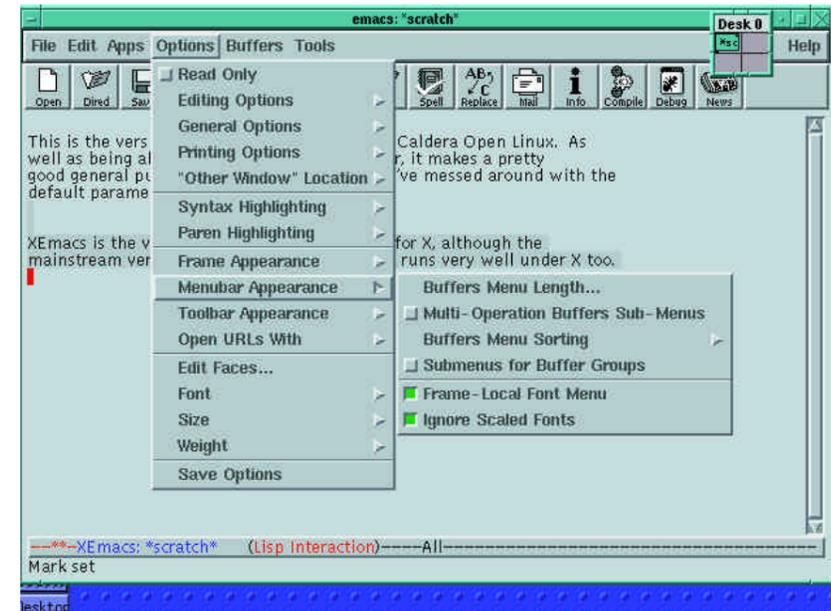
Until recently, RedHat provided the core operating system on which Caldera built the Caldera Network Desktop release. That has now changed, and the new Caldera product is called Caldera OpenLinux Base and it's built around the company's own implementation of Linux. Essentially, it's the same kernel as the one on our CD-ROM but with some additional Caldera features, including a proprietary implementation of PPP (the code used to dial up an internet connection), a commercial X-server and a fancy desktop called Visix LookingGlass. Also included is a licensed copy of Netscape Navigator.

Caldera plans a range of products under the name Caldera OpenLinux (COL) with Base as the entry level. In a month or so it will add OpenLinux Workstation which will include NetWare client software and a secure web server. Later in the year, the range will be expanded to include the Caldera OpenLinux Server; a multiprocessor version with a built-in SQL database intended as a high-powered web site, or as an enterprise intranet server.

I don't suppose many of you "home-brew" readers will be rushing out at the end of the year to spend over \$1,000 on this last item, but equally, I don't doubt that a lot of serious business customers will be flocking to Caldera to snap this up. As a full-blown commercial server it's not at all expensive for what it offers, and it will come with full technical support.

The interesting thing from the point of view of anyone using Caldera products is that they'll all be based on the same industrial-strength version of Linux. And if you're thinking "hang on a minute, you just said the high-end server version will be multiprocessor..." Yes, right. The truth of the matter is that the core of all the Caldera products will be multiprocessor. Bryan Sparks, CEO of Caldera, tells me that if you load any of Caldera's products onto a Quad Pentium box, it will find all four processors and know what to do with them.

You may also be thinking "Yes, but



The new Caldera OpenLinux Base comes with the latest update of Xemacs, the version of the Emacs text editor specially developed for X. I use variants of Emacs for all my writing, and as a totally cross-platform writing tool it cannot be beaten at any price (even though it's free!)

whatever happened to the spirit of free software among all this commercialism? Isn't Caldera just exploiting the labours of some dedicated, unpaid software engineers for its own commercial ends?" Well, frankly, I'm still in two minds about this. Caldera is certainly treading a delicate path in this case and we'll have to see where the company goes with it. In the firm's favour, though, this is probably the place to mention that it will be making available a freely downloadable version of OpenLinux from its web site at www.caldera.com. I understand that this will be stripped of any proprietary hinderences (so you will not get the LookingGlass desktop accompanying it, for instance) and will mostly be covered by the GNU licence, so the source code should be available.

The only exception, I am told, is that the installation routine will be copyright Caldera. This is not to deter you from passing a copy on to your friends once you've downloaded it, or making multiple copies across your own network: the intention is to prevent the code being exploited by certain CD manufacturers who have been bundling multiple Linux distributions into cheap CD sets and cornering the market. There's nothing in the GNU licence to prevent this. Indeed, the GNU licence is expressly designed to encourage distribution of all kinds, but the creators of easy-to-install Linux distributions have worked hard at writing installation routines and testing

them, and they feel they should be in charge of dishing them out.

Caldera OpenLinux Base is sold and supported in this country through a company called Avalan. It costs around £55 and the company's Ben Partridge can tell you where to get hold of it (see "PCW Contacts", p274).

Installing Caldera OpenLinux Base

Several readers have been asking me about installation on notebook computers. In the early days, the problem was getting the X-server to understand the LCD screen, but this has become much easier with last October's release of XFree86 3.2, which contains drivers for the main notebook video chips and includes an easy-to-use graphical setup. To help me gauge just how much easier, Seimens-Nixdorf kindly delivered one of its top-of-the-line notebooks, the Scenic Mobile 700, with a built in CD-ROM and stereo sound system.

It came with Windows 95 installed and my natural instinct was just to format the hard disk. However, I do know that many of you like to run machines that dual boot between the two operating systems and I get a lot of letters asking how to do this. Windows 95 was occupying the whole 1Gb of the Scenic's hard disk, so I needed to shrink this down to about half that size and create a second partition for Linux. Well, two new partitions, actually: one for the Linux root file system, and a second

partition dedicated to swapping. (If you are puzzled by this, look up "Linux partition requirements" in the Matt Welsh book, mentioned earlier.)

Firstly, I used Windows 95's own disk defragger to squeeze out the spare space and pack everything down into the lower 500Mb of the drive. To create the new partitions I could have used the fips utility, but this seemed a perfect opportunity to put Partition Manager 3.0 to the test. It runs from Windows 95 as an old DOS application, closing down the whole of Windows before loads and bringing Windows back again when it has finished. It shows a diagrammatic representation of the partitions on your drives and allows you to resize them visually by mouse dragging, and this gives you a much better picture of what's happening than having to calculate in megabytes or, as in the early days of fdisk, in cylinders.

In fact, I ended up with four partitions in all: one for Win95, a pair for Linux and Linux swap, and a fourth to house OS/2's Boot Manager... "Eh? How did OS/2 get in here?" you may wonder. It so happens that version 3.0 of Partition Magic comes with OS/2's Boot Manager as an optional way of handling multiple boots once you've got your partitions set up.

It's rather unfortunate that Boot Manager needs a whole partition to itself even though this is only as small as 1Mb, because the PC architecture only allows you a total of four primary partitions. But in this case it fitted in nicely. Installing the Boot Manager is a simple matter of clicking a pull-down menu from inside Partition Magic.

When I eventually quit Partition Magic I had just three partitions: in between Windows 95 and the Boot Manager was an unformatted wasteland onto which I was about to drop Caldera OpenLinux (establishing and formatting the Linux partitions is usually something you do during Linux installation).

Before you can do that, you first need to create one or more boot diskettes from the CD-ROM: at least, that is what I have always had to do so far; but not this time. Between them, Caldera and the Scenic Mobile had a neat trick up their sleeves — something I'd never seen or done before on an ordinary PC. The Caldera CD is set up to act as a boot disk on hardware that knows how to boot off a CD, and the Scenic's BIOS setup has an option to do just that.

A chat with Caldera's Bryan Sparks

Bryan Sparks, CEO of Caldera, was in the UK earlier in the year setting up distribution and support, so I thought I'd corner him and get the Caldera story direct from the horse's mouth, so to speak. He has a nice, soft-spoken manner and what he says sounds OK; not at all like marketing-speak.

Caldera started inside Novell as something called "The Corsair Project". Bryan Sparks was one of 15 researchers whose brief it was to look for new ways to develop quick time-to-market system software products. I wondered why he settled on Linux, rather than any of the other free UNIX offerings?

"We looked at NetBSD, FreeBSD and all the others," Sparks told me. "The truth is, FreeBSD is a good product and its networking has always been very, very strong. Linux's networking wasn't, at the time, but that was fixed when Linux 2.0 came out." (By the way, the Linux 2.0 kernel is the one used in our cover CD-ROM RedHat version.) Sparks calls picking Linux "a kind of gut move on our part". It was the originator of the kernel himself, Linus Torvalds, who finally decided the Novell team. "We flew him in to Utah to get to know him better because this was going to be a big risk for Novell. In fact, we got to know several of the key Linux people and we thought their personalities were perfect."

The idea for a commercial platform based on Linux was originally Sparks' own. He'd worked on several projects inside Novell, including the NetWare for UNIX that was called "Portable NetWare", and he was in on the early days of Novell's relationship with USL, the company that owned UNIX and which Novell eventually, and disastrously, bought. "It was a total mess," says Sparks. "I was working with USL for six months, and there were such irreconcilable philosophical differences that I just couldn't stand it any more." Tragically, Sparks could see the opportunities that UNIX represented, if only the politics hadn't been getting in the way. "I went back to Novell and said, 'Boy, you have the opportunity of a thousand lifetimes here'. Windows NT was totally unproven. UNIX System V Release 4 was decades ahead — well, I'm exaggerating, but it was a very good technology. But the personalities at USL just killed it."

However, the experience prepared Sparks for Linux. In fact, he liked Linux so much that he wanted to quit Novell and set up his own company around it with two other Novell employees. "But I owed a lot to Ray [Noorda], so before we quit we took the business plan to him." Sparks had put together a working prototype. "I had good friends at Visix giving me their user interface, and a bunch of other things I'd done myself. I showed it to Ray and he said 'Boy, we need to be doing that...'"

Ray Noorda OK'd the Corsair Project in early 1993, but in the following year, at the age of 70, he announced his retirement. "We ended up with Bob Frankenberg coming in," said Sparks. "I don't have any qualms about him: he listened to what we were doing (we were kind of a skunk-works, off-site). He was a very bright man and he said 'This is really good... but we're not doing it.' But Ray Noorda still believed in it and financed what became Caldera with his own money."

I felt I was cheating somewhat. This was just too simple: insert the CD, boot the machine, and up comes the skeleton version of Linux that Caldera uses to run its installation routine.

Anyway, I'll have more to tell you about Caldera on the Scenic Mobile, next month.



PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. Email him at unix@pcw.vnu.co.uk
Caldera OpenLinux Base is available through Ben Partridge at Avalan on 01923 208449; email Ben.Partridge@PSC.co.uk



The common good

... and bad, unfortunately. But Chris Bidmead is in combative mood as he attempts to install the Common Desktop Environment on his network, and defends a remark he made earlier.

A couple of months ago I mentioned that CDE, the Common Desktop Environment, is now available for Linux, and I showed you a screenshot of it running on LinuxPro, the version of RedHat Linux supplied by WGS (Workgroup Solutions) whose web page (unsurprisingly) is www.wgs.com.

I'd borrowed the screenshot from the WGS web site. But this is supposed to be the Hands On section, so I made a vow to myself that I'd follow through by getting hold of the software and installing it on my network. There's some bad news about this, some good news, some more bad news, but ultimately, I'm glad to say, some good news.

Before we get into that, I'd like to deal with some heated email from a reader who objected to my remark in the column that the CDE screenshot made LinuxPro look "uncannily like the AIX desktop — which of course is the point of CDE". When the major UNIX manufacturers got together in 1993 around an initiative called COSE (the Common Open Software Environment), of which CDE was to be the first component, the intention was to simplify things for system administrators and users by offering a "look and feel" that would be similar across all the UNIX platforms. One of the first manufacturers to implement this was IBM on AIX, and as I happen to run AIX here, it seemed worth mentioning the resemblance.

Andrew Lehane seemed to think I was suggesting IBM had invented CDE singlehandedly. If my very brief mention in the screenshot caption gave that impression, I'm sorry. Long-term readers of this column may remember that back in January 1994, speaking of IBM's

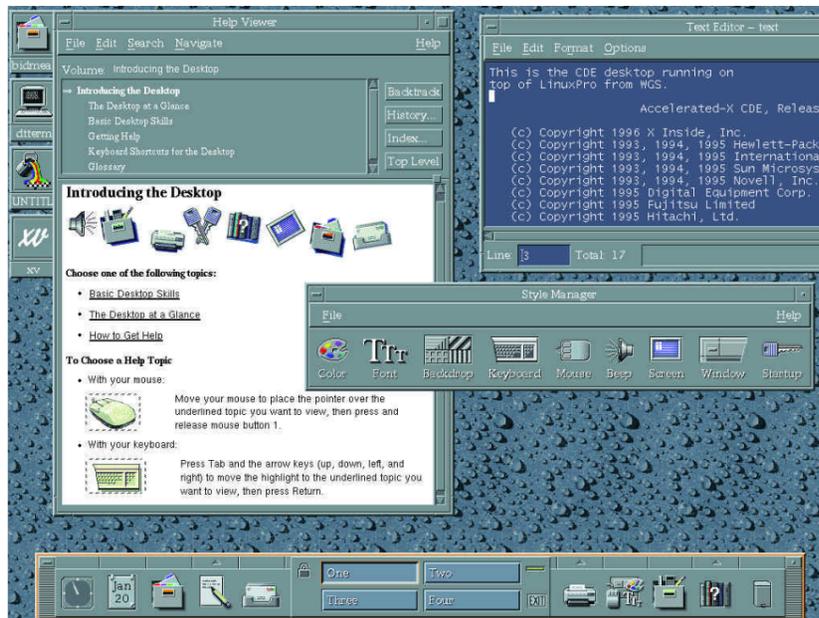


Fig 1 This is the Common Desktop Environment. It happens to be running on top of LinuxPro, but in theory it could be any UNIX and you wouldn't know the difference. It's not just a pretty face — there are internal subtleties like drag-and-drop which I'll investigate in later columns

introduction of the LaunchPad on OS/2, I said: "And the first of the promised COSE interface features also arrives with version 2.2 — a dashboard control based on Hewlett-Packard's VUE (Visual User Environment), a point-and-click program launcher and screen manager."

Yes, the panel at the bottom of Fig 1 derives heavily from VUE, but Andrew's complaint ("Why make these statements when, as I am sure you realise, CDE is in fact a decedent of HP's VUE Windowing System, an enhancement to OSF/Motif developed in post-Apollo days and released prior to 1991 for HP-UX 8.0?") seems to suggest that this is the whole story. In fact, all the COSE members

pooled their technologies to create CDE, as you can see from the copyright notice in the picture. As COSE was an initiative rather than an organisation in its own right — or as Sun's CEO, Scott McNealy, famously put it, "COSE is a verb and not a noun" — the product needed to be nurtured through the existing alliance of UNIX manufacturers, the Open Software Foundation (OSF) which has since been rolled into The Open Group. Andrew's loyalty to Hewlett-Packard (it turns out that he's working in its Telecom Systems Division) is commendable, and I'm delighted that he reads and likes the column, but I hope I've managed to deflect his accusation of "inaccuracy".

Good and bad

Now for that good news/bad news stuff. As soon as I discovered there was a Linux version of CDE, I dropped an email to Mark Bolzem, the Linux guru who runs WGS. The first lot of bad news was that he was very nervous about sending me review software because, as it turns out, it costs him an arm and a leg in royalty payments each time he ships the product. I find it ironic that someone whose core business is distributing royalty-free software should be held to ransom by the so-called Open Software Foundation, which charges him \$100 for the CDE and Motif components in each CDE shipment.

It's a tribute to Mark that he managed to get a package out to me, comprising the six-CD set that makes up LinuxPro, along with the CD and manuals for CDE. This was the good news. The implementation of CDE he uses comes from an outfit called Accelerated X, mainly known for its souped-up commercial version of the X server for UNIX on PCs. Its CDE requires the Accelerated X server, which is what Mark ships with his LinuxPro.

I installed LinuxPro and was all ready to go on to the CDE when the shattering news came through about Apple's acquisition of NeXT. I dropped everything and spent the next week or so on the internet, news gathering and chatting with gurus by email for some articles I was writing. After the dust had settled and it became clear that Gilbert Amelio may turn out to be the best thing that ever happened to Apple and NeXT (although it's a rocky road ahead) I finally caught my breath and went back to the relative tranquillity of LinuxPro and CDE. Or rather, just LinuxPro. Because — the next bit of bad news — I simply couldn't get my system to read the CDE CD.

I'd installed LinuxPro on the old Apricot Xen LS-II. You may remember the trouble I had with SCO OpenServer not being able to recognise the Xen's Sony CDU31a CD-ROM drive because it uses a proprietary non-SCSI, non-IDE interface. There's no problem with Linux though — a driver for the CDU31a is part of the standard distribution. One of the Linux system developers, Mark Evans, has even written a driver for the rather obscure on-board busmastering ethernet port on the Xen.

The Apricot Xen LS-II had read the LinuxPro installation CDs perfectly, and as far as I can remember had never shown any sign of trouble with other CDs before. But

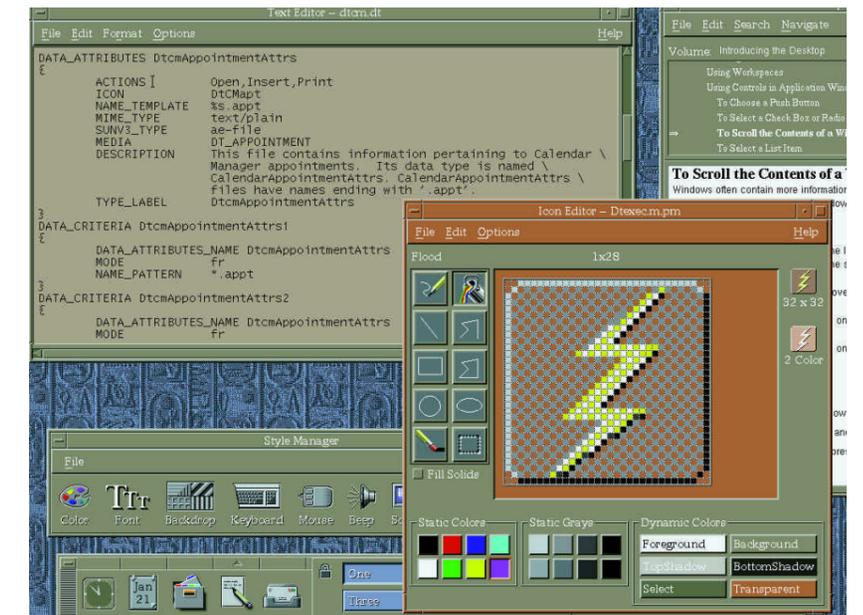
with this particular CD it was spinning its wheels, just at a time when I was desperate to install CDE and get a screenshot to you. Attempting to mount the CDE disk produced the error message (after a long hang) "can't read superblock". Even before I tried mounting the drive, its flickering LED indicated that it was labouring during the spin-up following inserting the CD. Oddly, the NeXT machine and the AIX PowerPC had no difficulty reading the CDE CD. It was just this combination of the Apricot drive and the CDE CD that was coming unstuck. After some experimentation — and I'm not going to tell you how many hours I spent messing around with this — I was forced to the conclusion that some drives, particularly older drives, just don't like some CDs.

This was distinctly bad news, and I found myself wondering how long it would be before I could free up another machine, reinstall LinuxPro on it, and then proceed with CDE. I was about to pack away the Apricot and sit down to write my excuses to you when I suddenly realised that this was the old, narrow, standalone-PC mentality that I'd supposedly escaped from three years ago when I started this column. The AIX machine, which had no trouble reading the CDE CD, is networked to the Apricot. Thanks to NFS, the Network File System that comes with every UNIX under the sun (or Sun, I should perhaps say), it's a trivial matter to hook up the AIX CD-ROM

drive straight into the Apricot and treat it as though it were a local drive. NFS is a client-server connection. Here, the AIX box is the server and needs to expressly "export" the drive in question. How you do this varies somewhat depending on the version of NFS. Under AIX, I used Smit, IBM's System Management Interface Tool, to mount the CDE CD in the drive and then export the drive — or rather, the directory I'd mounted it on — over the network. The directory doesn't actually go anywhere at this stage; it just gets added to a list of directories that the NFS daemon advertises as being available for network connection.

On the Apricot I then needed to mount the exported directory onto a local, empty directory. I created a /mnt/NFS directory and mounted the AIX CD-ROM there. If you read the manual pages for mount (you need to do this with the command "man 8 mount", 8 being the manual chapter that deals with system admin) you may come away reeling from the surfeit of options available. But the Linux mount command is pretty smart these days: if you just ask it to mount aixbox:/mnt/cdrom /mnt/NFS (mount this alien filesystem on this local directory), it should be able to figure out for itself that you're talking about an NFS mount and that it needs to be read-only because you're dealing with a CD-ROM.

Having done this, I had the full contents of my CDE CD available from the Apricot's



The CDE desktop reconfigured with a different background and colour scheme. Here you can see the icon editor together with a sample of the code CDE used to tie icons to executables and ascribe particular behaviours to them

/mnt/NFS drive. I switched to this directory, ran the dtinstall script I found there, and CDE loaded itself on top of my LinuxPro, with the handsome results you see in Fig 1.

Networking, modules and more

LinuxPro is based on the RedHat distribution, with some refinements added by WGS. Traditionally, Linux installation kernels come loaded with drivers for a huge number of devices and rely on autoprobing software to find out which ones to activate. This can leave you with an overlarge kernel, which is sometimes a tight fit where memory is limited. Once the installation is finished, you're normally recommended to recompile a slimmed-down version of the kernel tailored to your specific machine. For beginners, I should point out that this turns out to be a lot simpler than it sounds.

My Apricot Xen LS-II has 16Mb of RAM, which is plenty of room for Linux. Nevertheless, I prefer to start with the smallest possible kernel and add features as and when I need them. In the early days of Linux you couldn't do this, but newer Linuxes let you load "modules" which can be drivers for physical devices, file systems, or translation code like the iBCS module that lets you run applications written for other versions of UNIX. Mark Evans has modularised the driver for the Apricot Xen's on-board i82596 ethernet controller, so I installed a kernel that had no networking enabled, made sure this was working, and added the networking later. If you haven't yet installed Linux I should point out that most Linux distribution CDs come with a large variety of pre-compiled kernels in the form of disk images. You choose the appropriate image, transfer it to a floppy (where it appears as a bootable kernel and a bunch of vital support files) and use that as the initial boot disk. During the installation process you get the opportunity to install that same boot kernel onto your hard drive.

Typically, the installation process will mollycoddle you through procedures like setting up the network. The upside of this is that you get a working system with the minimum of effort. The downside is that you don't get to understand the fundamentals. Because I was starting without a network driver, the rest of the networking naturally refused to configure itself. Doing this manually was instructive and, luckily, not too arduous.

I found myself telling reader Michael Butler <m.butler@ic.ac.uk> about this at some length, and what follows below is a

Fig 2 Results of ifconfig

```
eth0      Link encap:10Mbps Ethernet  HWaddr 00:00:49:20:26:2F
          inet addr:192.168.1.25  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1176 errors:0 dropped:0 overruns:0
          TX packets:555 errors:0 dropped:0 overruns:0
```

Fig 3 Kernel routing table

Destination Iface	Gateway	Genmask	Flags	MSS	Window	Use
192.168.1.0 eth0	*	255.255.255.0	U	1436	0	569
loopback	*	255.0.0.0	U	1936	0	48 10

Fig 4 Resorting to ping

```
PING 192.168.1.3 (192.168.1.3): 56 data bytes
64 bytes from 192.168.1.3: icmp_seq=0 ttl=255 time=1.7 ms
64 bytes from 192.168.1.3: icmp_seq=1 ttl=255 time=1.4 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=255 time=1.4 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=255 time=1.3 ms
64 bytes from 192.168.1.3: icmp_seq=4 ttl=255 time=1.5 ms
64 bytes from 192.168.1.3: icmp_seq=5 ttl=255 time=1.5 ms
```

condensed version of that correspondence. Michael had written to me about the trouble he was having with his network card, and he'd opened his mailing with: "I have been converted to Linux through reading your column in PCW. It's a great column and worth the cost of the magazine alone!". This is a great way of getting my attention.

Once the install of the minimal kernel is complete, I login as root and then check that there's a set of modules, including the apricot.o module, somewhere among the library files. In my present version of Linux this turns up under /lib/modules/1.2.13/net.

Linux comes with a number of tools for handling modules, and you can get the full list with the command `apropos modules`. Among these is the `insmod` (insert module) command. This knows where to find the modules, and knows that modules are .o files, so the command line to add my network driver is just

```
insmod apricot
```

We've now installed the ethernet driver for the device known as eth0, but it's not yet doing anything very useful. You connect it to the network in two stages: first we need to give our eth0 a TCP/IP address, known as the "dotted quad" address. We use the `ifconfig` networking utility to set this up:

```
ifconfig eth0 192.168.1.25
```

With my own small network setup this

means I've given this particular machine the identifying number 25 on a network whose name is 192.168.1.0. (The full story is much more complicated, but let's not worry about that now.) The next step is to tell eth0 where to find the network. For this we need the `route` utility. Again there are all sorts of complexities you can get into with `route`, as you'll see if you consult `man route`, but for now let's just go with

```
route add 192.168.1.0 eth0
```

And that's basically it. If you now run `ifconfig` and `route` without command line parameters you'll get a report on the state of play. `ifconfig` gives something like Fig 2 (there'll also be an entry for `lo`, the loopback interface). The output from `route` should look like Fig 3. To check that the network really is connected I can use the old standby, `ping`, to send test packets to another machine and have them echoed back at me. For this I need the dotted quad address of the second machine

```
ping 192.168.1.3
```

which should give something like Fig 4, confirming the network is now working fine.

PCW Contact

Chris Bidmead is a consultant and commentator on advanced technology. Email him at unix@pcw.vnu.co.uk



DOS gags on an Apricot

All Chris Bidmead wanted to do was install DOS on his “universally compatible” Apricot. Just about anything else would have been easier. He isn’t going to suffer in silence, though.

Last month I was boasting about the universal compatibility of the Mitsubishi Apricot LS550 that is currently running SCO OpenServer on this network. But I thought I might have to eat my words when I tried installing DOS on a partition I’d reserved for that purpose. This was an overture to installing, of all things, Windows 95. I’ve had a lot of mail asking how you make Windows 95 co-exist with Unix and I was beginning to feel a bit cheap shrugging you off by saying “Windows 95 isn’t my thing”. So I thought I’d learn to suffer with the rest of you.

NeXTStep remains a lively environment, despite stories of the mother company losing interest in operating systems now that it is concentrating on web applications. The other day I downloaded a new update to my NeXTStep mail system: EnhanceMail version 2.0b5. Among other extensions of the official NeXTStep Mail.app, this package can translate ASCII smileys into little yellow faces with the appropriate expression. To give you a better look, I’ve dropped in a close-up window using Magnify.app from Eric Tremblay’s magnificent Walnut Creek Nebula CD-ROM shareware and freeware toolkit (eric.cdrom.com). You’ll notice, too, the X-Face enhancement to Mail.app: that’s the black and white mugshot of me in the top right-hand corner. The picture’s contained in the X-Face header string which can travel as ASCII via any mail system and be resurrected, cross-platform, by any mail reader with an X-Face handler

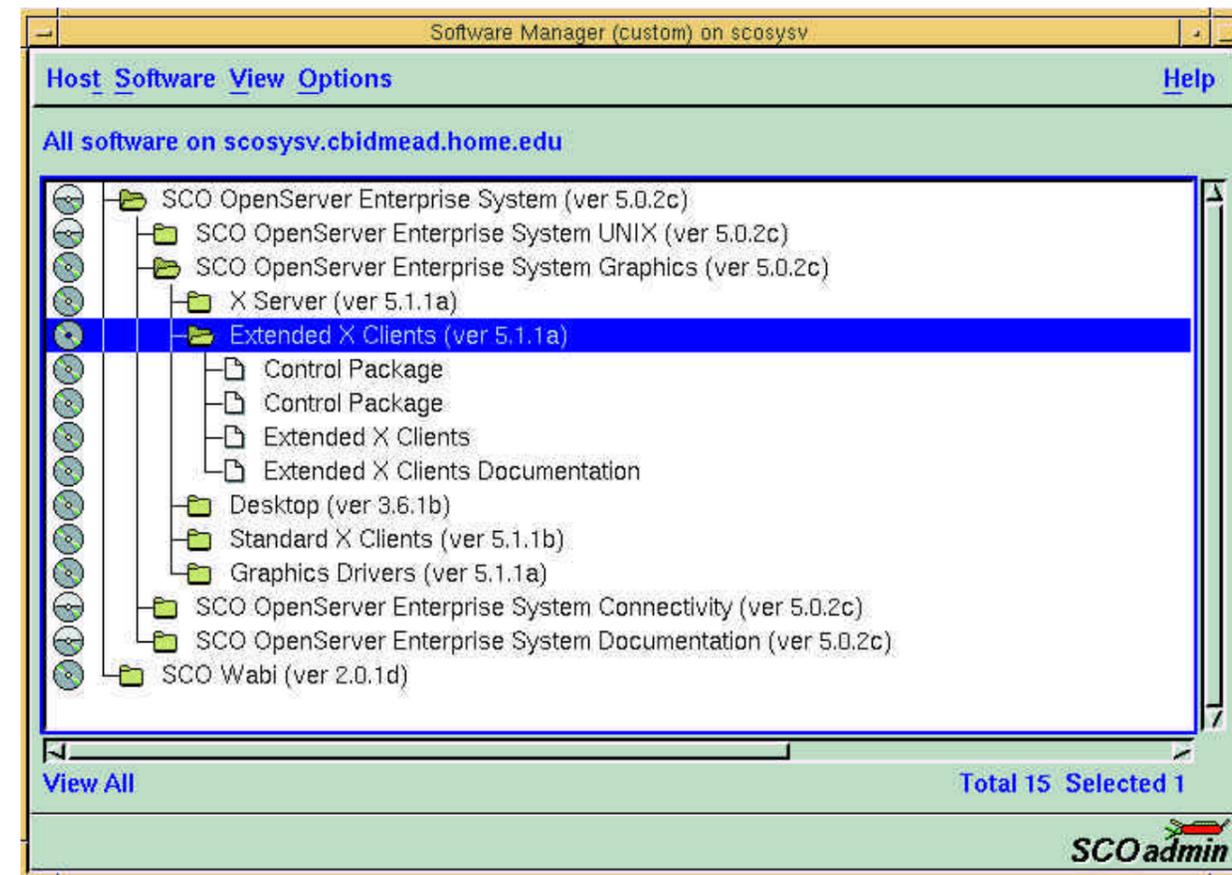
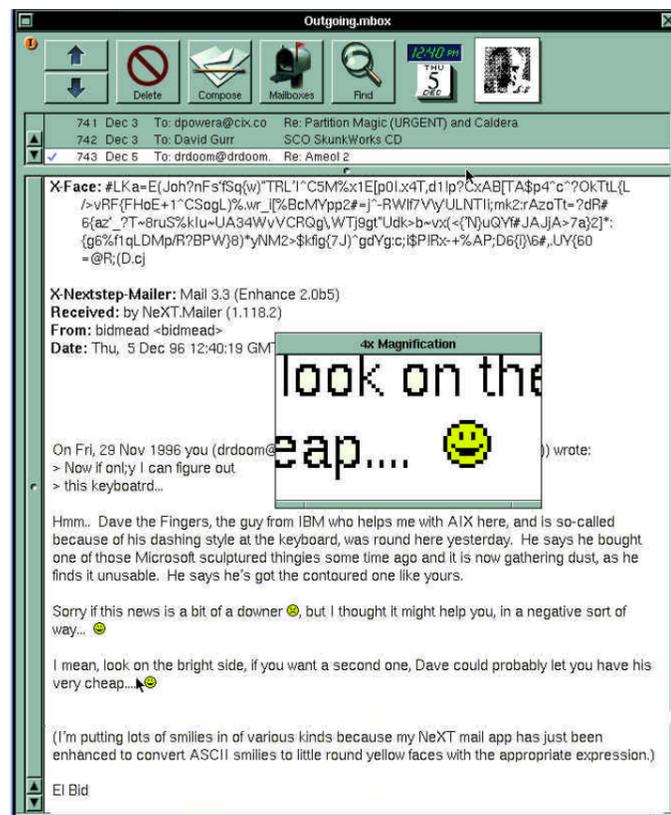
Well, I’m still suffering. But more of that in a moment. I was using a copy of MSDOS 6.2 that happened to be lying around, and the catch was that it was the Upgrade version. This looks for an existing version of DOS on the hard disk and offers to replace it. But if there’s no DOS it bows out politely, inviting you to contact your vendor for the full package.

This nonsense is all to do with the way IBM and Microsoft had agreed to carve up the DOS market, back in the eighties. I certainly wouldn’t encourage my readers to breach the fine print, but my personal

approach to this is to exit from the DOS installation at the first chance by hitting F3, and then run FORMAT C: followed by SYS C: from the command line. This puts the DOS 6.2 command.com and hidden boot files onto the hard disk. Then I run SETUP, which goes and looks at the hard disk and says to itself, uh-huh, seems there’s a copy of DOS on there already, so it’s OK to proceed. Well, that’s the theory. But in the case of the Mitsubishi Apricot, SETUP came back this time with “Incompatible Hard Drive”, and then a bunch of stuff about having to read things in manuals.

I started to worry about the 1.2Gb hard disk and went to look at the Apricot’s comprehensive BIOS setup. The BIOS is written by IBM, which reinforces my feeling about general compatibility, but I knew that DOS is easily upset by any departure from the straight and narrow, which at one time in its history included any disk greater than 32Mb capacity.

The BIOS offers a couple of disk-drive parameters that seemed relevant — “enhanced”, or “compatible”, which I guessed must be to do with whether the addressing is in logical blocks or old-style sectors. But changing these would wipe out the



SCO offers a comprehensive software manager for installing packages. But it needs to install a huge number of config files every time it looks at the CD, to see what’s there — a process that takes over five minutes. Having transferred them, you can install a single package, after which the software manager throws away the config files. If you want to go back for another package — yes, you’ve guessed it — the software manager has to spend another five minutes reinstalling the config files. Not a feature, admits SCO’s David Gurr; a bug to be fixed in the next version...

three hours or so it had taken me to install SCO, so I wasn’t inclined to mess with these as a first resort.

The next thing I did was to run DOS’s FDISK to assure myself that SCO hadn’t been so silly as to put the DOS partition high up on the drive where DOS couldn’t find it. DOS uses the BIOS to read and write from disk drives and traditionally the BIOS routines aren’t able to get to those parts of the disk above the 512Mb limit. But no, FDISK revealed the DOS partition to be the first one on the drive.

Nothing for it then but to read those manual entries to which SETUP was pointing me. With a sinking heart I turned to the DOS manual chapter called “Diagnosing and Solving Problems”. This is the wasteland where I seem to have spent much of the eighties — and Unix was supposed to be my escape from this. The section headed “Setup displays the ‘Incompatible hard disk or device driver’ screen told the whole story. Without apology, the DOS manual declares that

SETUP regards any drive on which it finds a Unix partition as “incompatible”, even if there’s a perfectly good, usable DOS partition waiting for the install. So much of this myopic megalomania surrounds Microsoft’s approach to alien products that I no longer regard it as an accident. I think Microsoft goes out of its way as a matter of policy to encourage the view that non-Microsoft operating systems are somehow weird and dangerous. Actually, the opposite is closer to the truth.

Got the T-shirt

If I may put the SCO/Windows 95 story on the stack for a moment... I got a T-shirt sent to me the other day with a button on it that looks like the famous Windows 95 Start Button, except that it says “Stop” with the words “Bill Gates” underneath. The legend on the back reads “Before he stops you...”. This came from Martin Houston (Martin.Houston@ukuug.org), a Unix consultant and organiser of the UK Unix Users’ Group which distributes the excellent

“news@UK” newsletter. Martin is an outspoken Linux enthusiast and although I share many of his views I’m not sure I’d go so far as wearing a T-shirt that states “Stop Bill Gates”.

I would like Microsoft to put a stop to these anti-social software practices, though. It’s time it learnt to live with the rest of the industry, rather than building software on the assumption that everything non-Microsoft is all going to go away quite soon. And I really hate the attempt by Microsoft to label its highly proprietary, uncooperative software “Open”. Around this time last year Michael Tilson, CIO of SCO, was reported in UniForum’s newsletter (www.uniforum.org/spool1/html/publications/uninews) as complaining about just this. “The term ‘open’ has been hijacked,” said Tilson. “Because ‘open’ is good, everyone labels whatever they sell as ‘open’.”

Incompatible hard disk — phooey!

Meanwhile, back at the Apricot, Microsoft is the company behind this version of DOS

that is declaring my hard disk "incompatible" simply because it has an SCO Unix partition on it. The straightforward solution turns out to be abandoning the automated SETUP routine and doing a manual install of the various files. Alternatively, there's an arcane command line switch, `SETUP /u`, that tells the installation not to make damn fool guesses about the drive and to just get on with the job.

The daft thing about all this is that most OS partition schemes have well-defined "partition signatures" (values buried at the beginning of the boot record) which identify them. So any operating installation procedure written by someone with half a brain ought to be able to come back and say "You have SCO partition on this drive, but there's plenty of room for DOS in another empty partition. Shall I go ahead and install?"

SCO OpenDesktop

I told you last month about how confused I became at Open Desktop's skulduggery in pretending that the Apricot's ATAPI host connector is SCSI. I was talking about this to Steve Perkins, the technical products manager for Western Digital in the UK, and he tells me that IDE's evolution has brought the interface very close to SCSI and the command sets are now similar. So what SCO is doing isn't as convoluted as it looks.

Having got over the SCSI/ATAPI hurdle, installation went smoothly. Admittedly it failed to automatically sniff out my network card, but it returned the right interrupts once I'd defined the type of card (a 3Com EtherLink II). I ran into a small problem defining the mouse: the one on the Apricot is what's normally called "PS/2 Type", except that this wasn't on the list of options. Foolishly I settled on "Bus Mouse", which completely immobilised my mouse cursor when X brought up the graphical desktop. This is a trivial problem, but one that can stop the show. After five years of messing about with Unix I am still a raw beginner in situations like these. I didn't know how to fix it, but what I have learnt in those five years is how to go about *learning* how to fix it. And Unix is a great teacher.

The first lesson of Unix is "DON'T PANIC". In real life this is just an empty slogan but inside Unix it's an eternal verity. With Unix you never *have* to panic, because it is an operating system that panics for you — panic being the term for what happens

The ISDN connection

There are at least three ways to use an ISDN connection from a personal computer. You can put a terminal adaptor (TA) where the modem was and connect to the PC with a serial cable. You can fit a TA into your PC like an internal modem. Or you can set up a kind of ISDN router on your network, so that all your workstations have access to the ISDN line. Paul Lynch, of P & L Systems, whose ZyXEL Elite 2864i TA I'm currently using, has offered to help me set up a router configuration based on the ZyXEL Prestige 2864i Remote Access Router which his company also sells. This sounds like a lot of fun and very much in keeping with the multiplatform tone of this column. But for now I'm getting to know ISDN the easy way — and it really is incredibly easy. First, you need an internet provider which offers ISDN (this seems to be most of them, these days). The ISDN line may be on a different number, in which case you need to change your dial string. While I was doing this I also changed the dial command from ATDT (tone dial) to ATDI, which is the Elite's string for initialising an ISDN connection. Most of the ISDN internet connections I've been trying, like the very friendly and fast Astra Internet service run by Chris Comley (ccomley@cix.co.uk), use the PAP authentication scheme I discussed in my January '97 column, so I had that already in place. And that's really all there is to it. It works!

when the operating system decides it's come across a fatal error and can't run any more. Typically it will put up a screen message that states something like "System panic — dumping core" and close itself down as gracefully as it can, creating a diagnostic log as it does so. You don't get this very often, and I wasn't getting it now. I just had a stuck mouse on the graphics screen, with the real Unix still chugging merrily away underneath. Happily, as with Linux, SCO allows you to switch out of the graphics screen into one of several virtual terminals by hitting Ctl-Alt and a function key. I logged into the virtual terminal as root and began sniffing around.

By long Unix tradition, the different peripheral devices like the mouse are stored in the `/dev` directory. They look like files and to some extent can be treated as such, but it's better to think of them as datachutes through to the actual devices they represent. If you know how, you can create your own `/dev` devices using a tool called `mknod`. SCO provides a friendly front-end to `mknod` in the form of a shell script named `mkdev`. I found this out by running `apropos` devices at the command line (`apropos` is an alias for `man` with the `-k` parameter, which states, "give me a reference to all the man pages that contain the following key word").

Then I consulted the manual (by running `man mkdev`) to find out how SCO does this. The manual told me to run `mkdev` with "mouse" as a parameter. When I did this, the script trotted me through the options for building the appropriate mouse device (or more probably, installing the correct link from a ready-made device; happily, `mkdev` shielded me from these grizzly details). It turned out that my PS/2 mouse is something SCO prefers to describe as "Low Resolution Keyboard Mouse"; a rather

weird description, so it's hardly surprising I missed it first time around. But call it what you will, my mouse was now working.

What still isn't working is Windows 95. You can install Win95 in the DOS partitions that SCO creates but then it goes and trashes the SCO boot. I reached for the proprietary boot manager, System Commander, to try to repair the damage Win95 had made to my master boot record, but it couldn't. The manual, however, goes into a description of the damage at some length and gives a very elaborate five-step procedure which it describes as "Risk Free Windows 95 Installation".

The manual also warns that once you have Windows 95 safely ensconced, you can still wreck your access to other operating systems when you install the bonus package of utilities and screen furniture that Microsoft calls Windows Plus. System Commander (from POW! Distribution at dpower@pow-dist.co.uk) offers ways around this, too, but having reinstalled SCO (which, in turn, seems to have trashed Win95) I decided it had all been too much excitement for one month.

As an added complication, Daniel Power at POW! has just sent me the latest cut of Partition Magic, originally an OS/2 partition mover and now a fully-fledged multiple operating system partition manager. This version comes complete with a licensed copy of OS/2's own Boot Manager so it's now an alternative to System Commander. If I'm feeling strong enough next month I'll give one or both of these a proper go and let you know how I get on.

POW! Contact

Chris Bidmead is a consultant and commentator on advanced technology. Email him at unix@pcw.vnu.co.uk



Read between the lines

Chris Bidmead passes on a salutary lesson, gained by trying to install SCO OpenServer without taking enough notice of the documentation.

I had to delay telling you about the new, free, SCO OpenServer (www.sco.com) because of installation problems. Yes, I know I promised a long time ago not to go on about installation problems in this column, but I do know a lot of you spend time struggling in this unfruitful area — as I do, too. I learnt some salutary lessons from (eventually) installing SCO, so I thought it would be useful to provide you with a quick rundown.

Initially, I completely failed to install SCO OpenServer on my trusty old Apricot Xen II 486, which has been a reliable home in the past for various different versions of OS/2 and Linux. SCO OpenServer is snotty about low-end hardware and in particular eschewed the Sony CDU31a proprietary interface CD-ROM drive. It thinks exclusively in terms of SCSI.

Apricot (or perhaps I should now call the company Mitsubishi Electric) stepped swiftly into the breach and sent me one of its LS500 range of desktop machines. This one is a 100MHz Pentium, equipped with a SoundBlaster 16 sub-system and a Cirrus Logic GD543x video chip on the motherboard. No SCSI, but the Triton chipset includes a dual ATAPI interface which drives the 1Gb hard disk and the CD-ROM. ATAPI, as I suppose everyone must know by now, is a recently standardised, simple, parallel interface that vies for performance with the lower end of the more traditional, more costly, SCSI connector.

What I like about the LS550 is its straight up-and-down compatibility. Buy a “bleeding edge” machine and it may turn out to be hard, or even impossible, to find drivers outside the world of Windows for the exotic Wide SCSI adaptor, or the gee-whizz 3-D

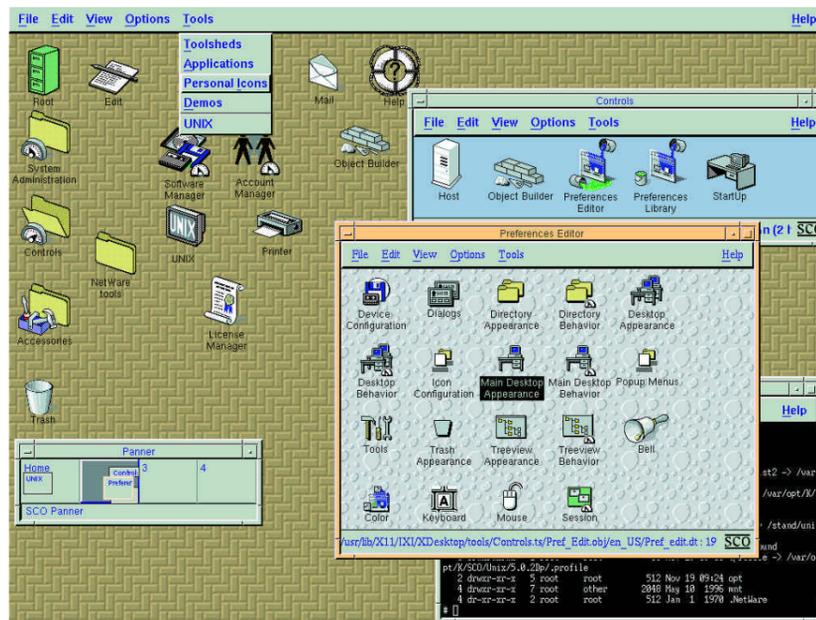


Fig 1 SCO OpenServer doesn't have the classy interface of NeXTStep, but the neat icons give a clean, efficient look to the desktop, and they can be cleverly animated

video system. All this stuff on the LS500 is widely supported, and is going to work very nicely with any of the operating systems I'm likely to throw at it.

However, SCO Openserver still refused to come quietly. The problem centred around that requirement for a SCSI CD-ROM. It seems there's a way around this if you read the installation notes buried somewhere on the CD-ROM. (Tip: if you can read the CD-ROM of a new operating system on your old OS, and you usually can, then do so. Scour it for information before you launch into a new installation).

Dave Gurr, market development manager at SCO UK, tells me that SCO realised the growing importance of the

ATAPI interface only after the architecture of OpenServer had been committed to SCSI. SCO's engineers will have thoroughly fixed the problem in the new release which is due out about the time you are reading this. On the version I was trying to install, there is an interim solution; you have to use a kludged loadable driver called wd that talks ATAPI to your system, yet kids the OpenServer that it's dealing with SCSI.

This works fine but confused me no end. The SCO boot disk detects the need for the wd driver, loads it and then invites you to fill in the various “SCSI parameters”. You have to go along with the deceit by telling the poor deluded system what your Host Bus Adaptor number is, along with the SCSI

address and logical unit number. ATAPI interfaces don't have these numbers, so you have to follow SCO's arbitrary scheme for translating between ATAPI and SCSI parameters.

An ATAPI system that hangs an IDE hard disk and a CD-ROM off the same interface will run the hard disk as master and the CD-ROM as slave. So I thought I was putting in the right numbers: HBA = 0 for primary ATAPI interface, SCSI address = 1 for CD-ROM as ATAPI slave. But the installation kept coming back to me with: “Can't Find SCSI interface”. Fair enough, I thought. I can't lie to you. There isn't one.

Following several attempts I re-read SCO's installation notes and the hardware spec for the LS550. The answer was staring me in the face, but you know what happens in these situations: you don't read it properly because you think you're covering ground you know already. If you ever find yourself getting as stuck as I did with this one, the only thing to do is to throw away everything you think you know and re-learn things from scratch.

I had picked up the idea from the SCO installation notes that the IDE drive and the CD-ROM were hanging off the ATAPI interface as master and slave. But when I had read the Apricot LS550 documentation carefully, I noted that the Triton chipset provides two ATAPI interfaces. The CD-ROM wasn't a slave, it was the master on its own secondary controller. The correct

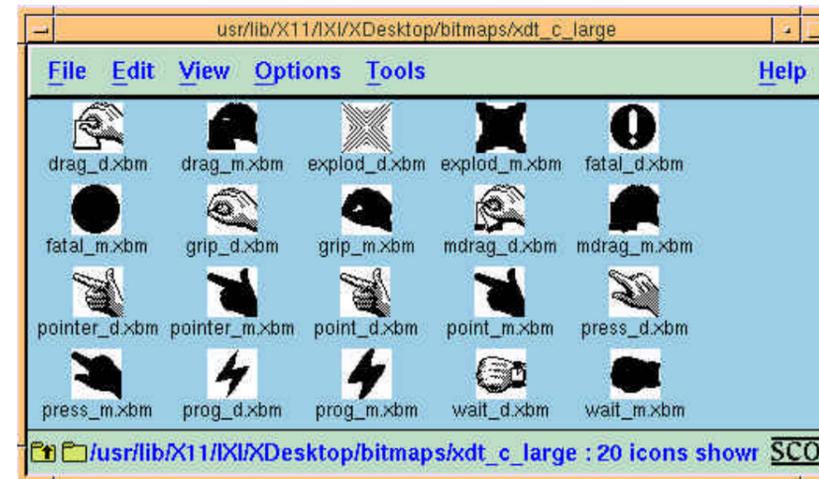


Fig 2 SCO: The pointing finger icon, which is the mouse cursor, pincers its finger and thumb to pick up objects, and rotates its wristwatch into view while waiting for a job to finish. Here are the icons that SCO's X.Desktop uses to do this

parameters for the wd driver were HBA = 1, SCSI address = 0. After that, the installation found the CD-ROM drive sweet as a nut, and the rest was plain sailing: a totally automated, go-away-and-have-supper, one-hour-and-a-bit data transfer from the CD-ROM to the hard drive.

Sterling work, Carruthers

One of the minor joys of Linux-FT is that when you install it to load the UK keypad and hit the pound sterling sign on top of the number 3 key, you actually get the £ sign. Try this with RedHat or any of the other US Linux distributions and the chances are you'll hear a beep and nothing will turn up on the screen. This is the console screen I am talking about — so let us leave X out of it for now...

One of my early excursions into the bowels of Linux a couple of years ago was an attempt to track down this pesky little problem. It turns out there's no need to do what I did, and start unpicking the source code. But on the other hand, reading chunks of the source during this venture was a valuable part of my Linux education. If you haven't explored the source on your system, why not cd to /usr/src/linux and have a mooch around. This is the one big advantage that the Linux folks have over devotees of, say, SCO. Even though SCO OpenServer is now “free”, you won't find any source knocking about.

You'll be asked during the installation of most Linuxes which keyboard you want to use, and thereafter the appropriate keymap will be loaded automatically each time you boot. You can see how this works in

Linuxes such as RedHat that use the System V initialisation scheme (most do these days) by finding a directory called rc.d under /etc. This is where all the initialisation scripts are run, like a kind of glorified archipelago of AUTOEXEC.BATs, every time the machine comes up or whenever you shift from one run level to another. On my RedHat system there's a sub-directory under rc.d called init.d which contains a script called keytable. Find your own keytable script and the related scripts to see how the appropriate keymap gets loaded.

You might think that loading the correct keymap would be enough to set you up with the £ sterling sign. But alas, not so. The keyboard mapping is only part of the story. You will also have to make sure that an appropriate screen font is loaded. This is achieved by a utility called, reasonably enough, Setfont. Setfont should be loading an ISO Latin font such as lat1-16.psf. You can do this manually while testing the system by running Setfont from the command line. See the man pages for Setfont for full details.

So you've set your screen font and your keyboard. What happens if, as I found, your keyboard still fails to deliver a £ sign? Well, frankly, I gave up. I got used to writing “100 UKP”. It didn't help that my favourite machine came with a US keyboard that doesn't even have a £ sign, and I've never bothered to change it. I completely forgot about the problem until I ran into a Unix veteran with the splendidly colonial name of Shaw Carruthers in one of the Linux electronic discussion groups running on the CIX conferencing system.

In a lather with Java

I came across a nice little anecdote in the comp.os.sys.be newsgroup, which I reproduce here with the permission of its author, Kurt Glaesemann who is at kurtg@iastate.edu. The discussion was about some of the more unreasonable things people expect of computers and Kurt mentioned a web site he'd visited where nothing seemed to work. He inspected the source code for the page, where he found a section that went something like this:

```
If (JavaScriptExists) then
...cool stuff...
else WriteJavaMessage("sorry
this page needs Javascript")
end if
```

I leave it as an exercise to the reader to work out why this might cause problems with a browser that doesn't support JavaScript. (Hint for beginners: the script above is written in JavaScript).

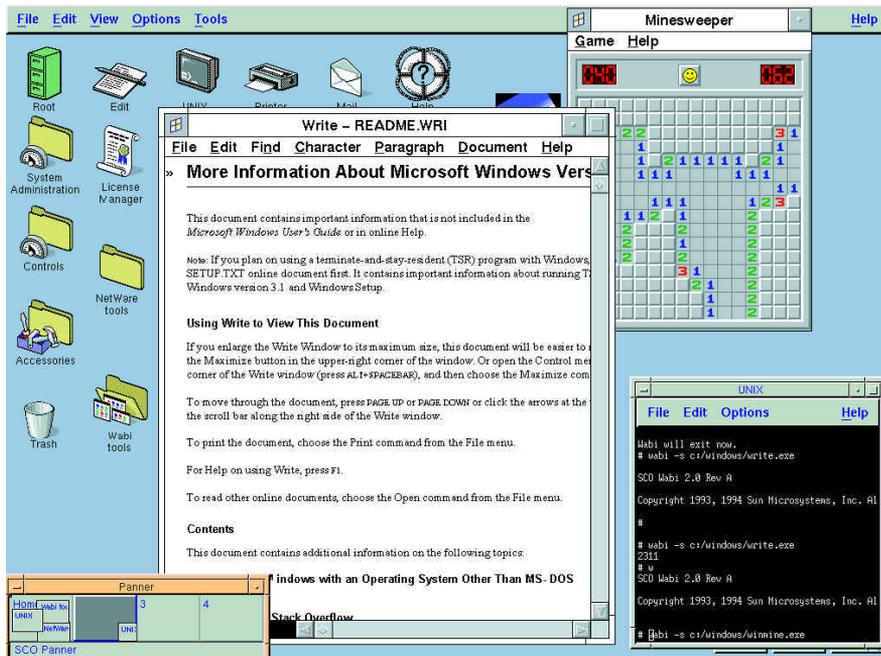


Fig 3 SCO OpenServer comes with Wabi 2.0 (as a cost extra) which I'm using here, somewhat unadventurously, to run Microsoft Write and Minesweeper. You can open Wabi as a Windows environment, or you can run individual Windows apps in their own windows, rather like "seamless" WinOS2 under IBM's Warp

Shaw told me that the two other elements you may have to fix are the ability of the application you're running to handle ASCII characters with values over 128, and the behaviour of the console you're using. Don't forget that even if you're only on the command line, you're still running an "application" — the shell.

With current versions of Linux, this shell is likely to be bash (Bourne Again Shell) by default. (You can usually check this by typing "echo \$SHELL" at the command line and seeing whether it reports "/bin/bash". If you don't get any output you probably forgot to put \$SHELL in capitals). When bash loads it consults (among other settings) files such as .profile a personal keybinding file called .inputrc, which sits in your home directory. Take a look at man bash for the full details. Shaw suggests that you include the following lines in your .inputrc file to make sure that bash can cope with European characters:

```
set meta-flag On
set convert-meta Off
set output-meta On
```

Now let's look at the console behaviour. What you know to be a character-based, memory-mapped screen is, as far as UNIX in this mode is concerned, a dumb terminal sitting at the end of a serial connection. If you enter the command "stty —all" (or "stty all" on older systems), UNIX will tell you all it

knows about the terminal, including its alleged "baud rate". You can also use stty to change the features of the virtual terminal, including its handling of ASCII chars above a value of 127. Shaw suggests that you add the following lines to your .profile script to make sure the console is 8-bit clean:

```
tty -s
if [ $? = 0 ]; then
stty cs8 -istrip -parenb >&0
fi
```

The tty command silently (-s) checks to see if it really is dealing with a terminal and returns 0, tested in the following line, if this is the case. Stty then sets the character size to eight bits (cs8), negates any tendency to clear the high bit on input characters (-istrip) and tells the terminal not to look for parity bits (-parenb). You'll notice the minus sign isn't used here to flag a parameter, as it is in many UNIX commands (like the tty command in the first line), but to toggle a parameter on or off — just one of the delightful inconsistencies which gives UNIX its charm.

Preparation for ISDN

Following our discussion about ppp and dial-up to Internet Service Providers last month, I've taken a bold step forward and installed a base rate ISDN line.

In theory this gives me higher

Linux gets Wabi

As I write, Caldera has just announced that it is shipping version 2.2 of the Linux port of Sun's Wabi, the Windows environment for UNIX. If you really must run Microsoft software such as Excel and Word on your UNIX machine, this is certainly one way of doing it.

I'm hoping to take a closer look at Caldera's offering in the near future. Meanwhile, I have the SCO OpenServer version running here (Fig 3), and it certainly does the job, provided you define the job as only running a limited subset of 16-bit Windows applications.

What worries me about all this is the price. Version 2 of Wabi requires 16-Bit Windows, which (on my SCO system at least) you have to install yourself. So add the cost of a Windows licence, to the Caldera price of nearly \$200, and you've got yourself an expensive way of running old Windows applications.

bandwidth to ISPs (provided they support ISDN, of course) although the speed of the link in practice is dependent on factors like the load on the ISP and the remote servers you are visiting.

There's an immediately discernable benefit in the speed with which you establish the connection, because ISDN dispenses with the dialling tone and the tuneful duet of line speed and communication standards negotiation that can take up to 30 seconds with a ordinary analogue phone line.

As well as the ISDN line you need a terminal adaptor (TA) through which you make the computer connection. It helps if this behaves as closely as possible to a modem, so that your software doesn't have to be completely rethought. Because base rate ISDN is effectively a pair of lines, it's also useful if your TA has an analogue input that you can treat like an ordinary phone line, either for a modem connected to a second computer, or for use as a voice line. P&L Systems, which is run by Paul Lynch whom regular readers will remember as my NeXT guru, has lent me one of its ISDN TAs. It's the ZyXel Elite 2864! "ISDN modem" and it has all these features, and more. Next month I'll tell you more about how I got this going with my UNIX network here.

• PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. He can be contacted at unix@pcw.vnu.co.uk



All together now

Chris Bidmead claims things are changing for Unix users. Then again, he said the same in 1993... Still, he remains optimistic: 1997 could finally see some Unix togetherness.



One of the things this column asked Santa Claus for last Christmas was for the Unix community to get its act together. Something along these lines — dare I hope — is actually happening. In February, we saw the coming together of X/Open and the OSF to form The Open Group, followed in September by a declaration of intent from Uniform to join the amalgamation.

My dentist tells me that amalgam is a metallic mixture that sticks in your teeth and goes hard and almost inert except for trickling out a few, probably poisonous, Hg ions that may actually help keep decay at bay. A not-too-unfair description of the massed forces behind Unix in the past, you might think (I couldn't possibly comment...). But things are changing.

One sign of this change is the way The Open Group is welcoming the Open Unix effort to its bosom. Open Unix is a development of Lasermoon's Linux-FT, now under the aegis of Caldera (more about this at www.caldera.com). And I see that CDE, The Open Group's Common Desktop Environment, has now arrived on Linux. CDE was part of the initial Unix COSE initiative that was proposed in March of 1993, but the fact that it's taken three years is hardly the fault of Linux.

I mentioned the COSE initiative in the first of these columns, written in September of that year, and was somewhat sceptical about it at the time (see panel). Indeed, it wasn't long before COSE began to look like just a knee-jerk reaction against Microsoft's

NT, with little muscle behind it. It would be ironic if Caldera's Open Unix became the focus for The Open Group's so-called Single Unix. Perhaps this threat of rivalry is one reason that SCO has decided to make its own Open Server flavour of Unix freely available to anyone not using it for commercial purposes.

So, I promised to get hold of a copy of the SCO freebie and report back to you. It took rather longer than I anticipated, and the CD and its accompanying pair of boot diskettes arrived too late for evaluation this

COSE in 1993

In September 1993 I wrote: "COSE... sounds terrific. You'll be able to sit down at any Unix workstation from any manufacturer and instantly be at home, in the same way that Windows users are today. Better still, applications will run 'right out of the box' on Unices on the same processor, and to move a COSE app across to a different processor will only need recompilation — no tweaking and twiddling with the source code..."

And porcine domesticated livestock will become airborne. Alas, the Unix world has promised various forms of togetherness in the past but has still managed to remain a mass of twisty passages, all different. Fingers crossed.

month. Next month, if all goes well...

PPP to the internet

A debate among journalists on one of the electronic forums recently discussed the relationship between information that goes out in columns like this, and information that is available on the internet. I could, I suppose, stuff this column with rejigged wisdom culled from the internet; certainly, internet FAQs and newsgroups are a very important source for me. The Truth is Out There (if you can filter it out from the junk

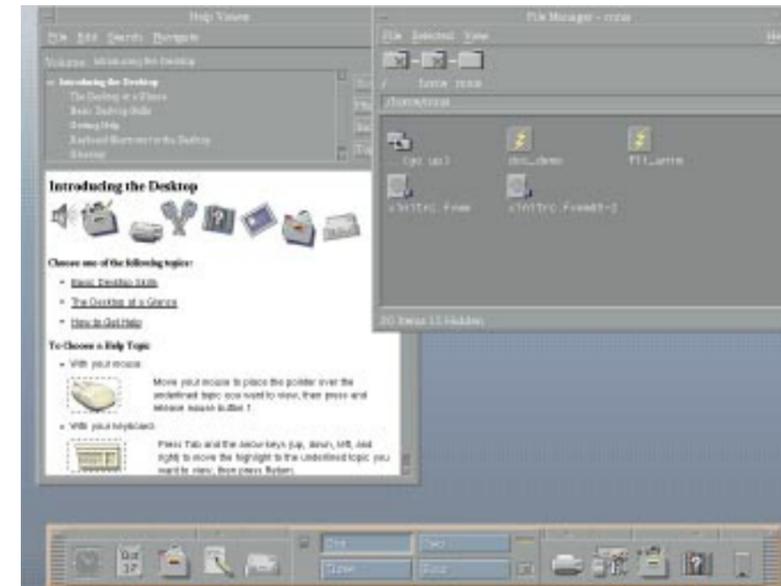
But I'm starting to assume that if I can get to it, most of you can get to it too. So my role isn't to relay woggles of publicly accessible material. What you get in this column is my own personal adventure/quest/struggle to get a few particular things done, and generally to make sense of what I call "grown-up computing". I do give pointers to what I think is good and useful information on the net, and occasionally I may summarise. But mostly I take it for granted that cyberspace is something we share.

I realise this may not be true for everybody. But if you're adventurous enough to be running a flavour of Unix, or to contemplate doing so, you're probably adventurous enough at least to be thinking about connecting to the internet. My minimim recommendation: get yourself an email address.

Unless you're one of the lucky few with a fibre cable coming in off the street, you're likely to connect through a serialised network connection called PPP (Point to Point Protocol). In October, I passed on some tips for tracking what's

happening with PPP once you're making connections through it. The month before, I explained in general what PPP is, apropos some problems I'd been having with my own internet connection. Last month, I included some screenshots of GateKeeper, the graphical front-end that drives PPP on my NeXT machine.

If I'd realised I was going to turn the PPP saga into a serial, I'd have been more methodical about it. Notably missing so far is a discussion of how to set about making



Left The Common Desktop Environment. This is what you see when you first log in to Linux Pro Desktop, available from WGS (www.linuxmall.com). It's uncannily like the AIX desktop — which of course is the point of CDE. The Front Panel (long strip at the bottom) is also a relation of OS/2's Launchpad

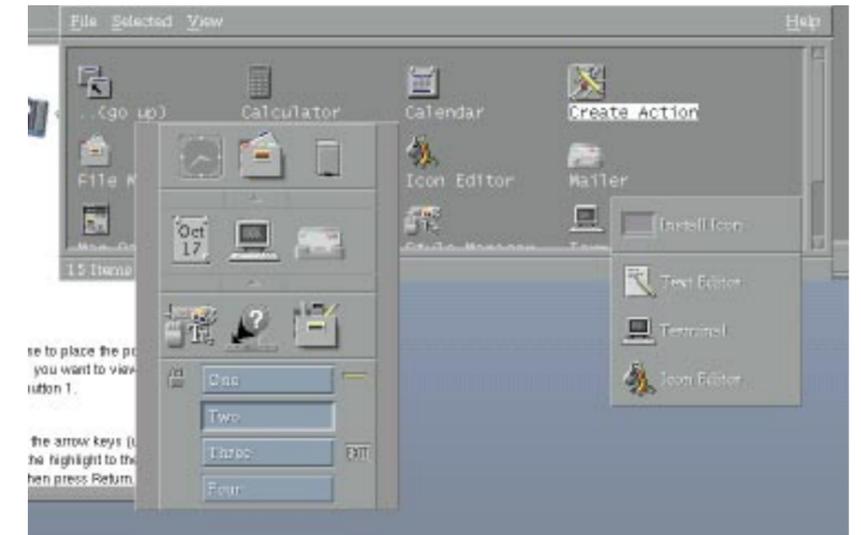
Below A closer view, with the Front Panel transformed to a more vertical shape. The panel to the right is a tear-off from the main Front Panel. As with OS/2 the icons are objects, which can be allocated behaviours with simple scripting. For full details, see the WGS home page

the connections through PPP to your internet service provider in the first place.

The arrival on my desk of a shiny new Surfer modem from Psion-Dacom has prompted me to work on PPP afresh. The Surfer comes bundled with software and trial accounts for Pipex, CompuServe and AOL and, as such, represents a real bargain for mainstream computer users. But for those of us who see the so-called mainstream as just a shallow but loud babbling brook, this isn't a lot of use.

Because all the software is for just one operating system, or maybe two — Windows and the Mac — does this mean that Unixen should just give up? Sometimes the answer, alas, is yes. I mentioned PinkSlip a couple of months ago. That's a proprietary protocol that Pipeline uses, which only works with their own (Windows) software. AOL does something similar, which again locks you into AOL's own software, and therefore into Windows. Yes, it's completely crazy, and I gather these companies have seen the error of their ways, but it's going to take a while for them to fix it. So until then, Pipeline and AOL aren't in our frame.

Generally, you'll find ISPs offering a PPP connection, but one of the biggest, the IBM Global Network, still only does the older SLIP (non-)standard. You can cope with SLIP from Unices like Linux, but I decided to put that aside for the moment and concentrate on PPP, which is fast becoming the all-embracing standard for TCP/IP down a serial line, and it will also cope with other transport



protocols like IPX.

I find the best way to tackle vendors offering PPP is just to plunge straight in with a modem and a simple utility like tip or cu that lets you talk to your modem. You could try ringing their help desk, although I hope you fare better than I did (see page 282).

Before you can get onto the remote system, you obviously have to make a physical connection by dialling up. Once you're connected, the ISP needs to know a) who you are (username), and b) whether you really are that person (password). There are some more complicated schemes that do further checks (on your hostname, for example) but I haven't come across them. So basically you need to get the name and

the password across somehow.

The "standard way" is an ASCII exchange before you bring up PPP. If this is what the ISP needs (Netcom UK works this way), you can do this manually from any terminal-type program that talks down a serial line to your modem. In what follows, I'm using tip on my NeXT machine. (I notice that the manual for dip on my version of Linux carries this succinct comment at the end: "BUGS: This program does not work very well.")

First, type ATZ and hit carriage return. You should get "OK", which means the connection to the modem is working. Then you dial:

```
<modem initialisation stuff> ATDT
<phonenumber>
```

Assuming you've got the baud rate

right, this should bring up a prompt string that says something like "Login:". Baud rate and initialisation strings used to be a major hurdle in the good old days, but modems at each end of the line seem to have become a lot more intelligent about getting this right automatically. If in doubt, keep it simple.

Then you do the authentication exchange. Typically, this goes:

```
Login: <yourname>
```

```
Password: <yourpassword>
```

where the stuff to the left of the space on each line is the incoming string (from your point of view) and the stuff to the right is your response. Netcom UK requires you respond to Login with:

```
uk, ppp, <yourname>
```

Demon adds a third prompt where it asks you for the protocol "Protocol:" and you respond PPP. One reason for doing this manually is so that you can see exactly what the prompts are.

If this dialogue is accepted, the next thing you'll see (maybe after a cheerful "HELLO", which might require you to pause for a few seconds) is a stream of garbage characters. This is the remote PPP throwing binary at you. At this point, you bring up PPP at your end and the exchange continues in binary. You're connected. Now you can go away and write a script that does this automatically.

Here's a tip: have pppd ready to run in a second shell window. This way you can bring it up quickly — sometimes the ppp at the remote end will sulk if it doesn't connect to you right away. And don't forget (as I did when I first tried exploring this) that the utility is called "pppd" with a "d" on the end (because it's a daemon).

A faster and more secure way of carrying out the connection is with PAP (Password Authentication Protocol). If the ISP is set up to do PAP, you start up PPP the moment the modem tells you you're connected, and leave it to PPP to carry out the authentication. You can still get as far as bringing up PPP manually, but from then on, you depend on PPP to do the rest.

How does PPP know what to do? The PPPs on most of the Unix boxes I've come across derive from the free version originally written by a team lead by Drew Perkins (you should see the name come up when you run pppd). These accept a command line parameter "+ua <filename>", where

<filename> points to a file that contains just two lines, <yourname> on the first line and <password> on the second line. When the ppp daemon sees the +ua parameter, it knows to carry out PAP using the stuff you've put in the filename.

Those are the basics, and at this level, if you treat PPP itself as a sort of black box, they really are that simple. The documentation supplied with your system should be enough to take you to the next stage of automating the whole process with the chat scripting utility. If, for some reason, things don't work, or you are a glutton for PPPunishment, O'Reilly's *Linux: Network Administrator's Guide* includes a whole chapter on the subject.

The main pppd command will also tell you a lot about ppp. Be careful with all that stuff you can put in the command line or in

system. Some of it is nostalgic, but much of it reminds me that Pick is still a force to be reckoned with, even if it does get a pretty low profile in the mainstream computing press today.

Mark Chapman (mark@wwsLtd.demon.co.uk) runs an outfit in Tunbridge Wells that specialises in software for Homecare and Nursing Agencies. He's a Pick user — he describes it as his "favourite database". Pick Systems, he tells me, claims to have become more businesslike since the demise of its founder, "which seemed to mean shedding staff by the cartload." Mark, too, remembers a night out he had with Dick Pick. "I thought he was an incredible bloke. He was well into middle age and yet a serious thrill seeker. I had thought all his jet skiing etc was just really naff marketing, but the guy was genuinely wacky."

Chris de Vaney (chris@wsel.lu) is another computer professional who responded enthusiastically to the Pick namecheck. "Oh boy, do I remember the Pick system. I evaluated it twice for large UK installations in the early eighties, and I just didn't believe how powerful it was. The natural language query interface still hasn't got anything to match it 13 years later!" Chris's story about an estimated two-year engineering application backlog for a broadcasting organisation being covered in just over eight working days using Pick, is typical of the mail I've been getting. Thanks, everybody. Over the three years I've been doing this, the column has already spun off standalone

ISP helpline howler

Here's what happened when I rang the support line of a well-known ISP.

Me: Do you do PAP or require a pre-PPP ASCII dialogue?

Help Desk: What operating system are you using?

Me: Does that matter?

HD: We need to know so we can tell you how to set up your software.

Me: I'm not asking how to set up my software. I'm asking what your software requires so I can supply it. I'm doing it manually to start with. When I understand what's happening, I'll be able to set it up for any operating system and/or application that I want to use.

HD: (After a long pause) Did you say "manually"...?

It seems that any wisdom they have about making the connection seems to be encapsulated into particular applications and operating systems. The help desk staff I've talked to certainly aren't daft, it's just that they seem not to have thought of it in terms of what's going on underneath.

an options file. The modems (or more likely, "virtual modems" like Ascend boxes) that ISPs are using these days are very smart and seem to be used to treating you as very dumb. The simpler you can keep it at your end, the more likely you are to connect. For example, you'll discover that PAP is a two-way street, allowing you to say back to the remote system: "OK, that's who I am. Now let's check who you say you are, and who you really are?" My recommendation is not to bother with any of this, unless you have a lot of time on your hands...

Pick up on Pick

My diversion in the October issue about Dick Pick seems to have triggered off a stream of email from fans of the operating

columns about OS/2 and Windows NT. Maybe Pick will be next? Or maybe I'll legitimately be able to write more about Pick here (a few readers write in to complain if I wander away from purest Unix for a moment). Because rather like NeXT, the main thrust now for Pick, Mark Chapman tells me, is to host it as an environment on other operating systems. And the entry-level version of Pick is to be hosted on, guess what? Linux.

PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. He can be contacted at bidmead@cix.compulink.co.uk



Hot Topic

The weight of useful, and useless, data available on the web prompts Chris Bidmead to dust off his Topic text retrieval system. And, a web site which compiles your kernel to order.

Last year I mentioned the text retrieval system I use on my network here, a product called Topic, from Verity. It looks after a database of practically everything I've written since the early eighties, together with snippets of useful stuff scanned in or collected electronically from the outside world.

Five years ago, Topic was the centre of my computer writing activity. But the arrival of the web, with its comprehensive search engines, has shifted the balance somewhat. At one stage, I came round to thinking that an in-house text retrieval system risks becoming close to irrelevant under the sheer weight of information available in cyberspace. I neglected Topic, and began keeping my current output in a number of Digital Library files, a vastly more simple text retrieval mechanism built into NeXTStep.

I still had Topic as a method of searching the Bidmead legacy archives, but my access to it was through ageing character-based OS/2 client software that even in its day was somewhat clunky. In comparison with the NeXTStep user interface, it looked like something out of the ark.

But I was wrong about the value of the web. There's a ton of stuff out there and it's a valuable on-going education. But free text searching can waste an awful lot of time when the raw material can be just any old junk put up by anyone who wants to build a web page. Democracy is a fine thing but if you throw open the Opera House to anyone who fancies a sing-song, you can't expect La Traviata.

The web is no substitute for a carefully qualified and managed in-house text retrieval database. Clearly, I had to start cleaning the rust off my Topic implementation and getting it back to work.

The historical roots of Topic lie in UNIX and from this spring a couple of characteristics that I find valuable.

Firstly, unlike all the "personal text retrieval" products I've used in the past, it's died-in-the-wool client-server software. Secondly, it is configured and administered through an initially baffling collection of plain ASCII files riddled with

Unixy black magic incantations. "He thinks this is good?", you mutter. Yup. And the reason I do is the reason I write this column.

Topic, as I mentioned here last year, runs under DOS, OS/2, Unix and a number of other operating systems. What I didn't mention is that thanks to its client-server design, Topic can also run under a combination of these environments. For example, you could do the indexing under Unix, run the search engine on Windows NT and have OS/2 handling the client software.

In my time with Topic, I've used combinations of DOS, OS/2, Unix and Windows NT to prepare and serve the data, each part of the work being handled by the operating system best equipped to do the job. The appeal of the arcane text files that

control Topic is harder to explain. After three years of neglect, I had to delve back into configuration files which looked like Fig 2. Each text database, or "collection" as Verity calls it, is controlled by a directory tree stuffed with a variety of files like this. I won't go into details here, but the general principle is that the stream of your documents coming into the index system is filtered into plain ASCII, examined for particular patterns to pull out fixed fields (I use Title, Source, Date and Author in my standard collections) and then an inverted index is created of all the text in the body of each document.

Many simple text retrieval packages just index which words are in which document but don't bother to log exactly where each

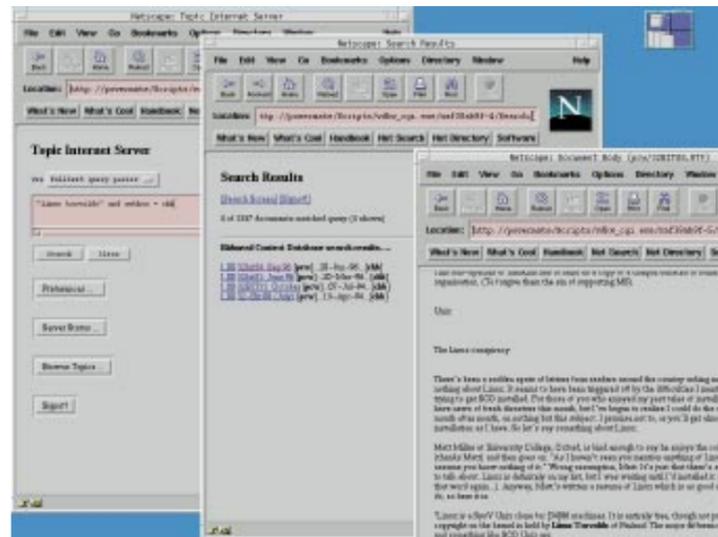


Fig 1 Topic Internet Server is the Verity search engine linked into the web server of your choice. The result is a ubiquitous text retrieval system that can be accessed across the network from a variety of different machines. This is how it looks from Caldera

Fig 2 — delving into config files

```
$control: 1
descriptor:
  /collection = yes
{
  data-table: _df
  /num-records=1
  /max-records=1
  {
    # Header information for partition management
    constant: _DBVERSION      text "vdk11"
    fixwidth: _DDDSTAMP      4 date
    varwidth: _DOCIDX        _dv
    fixwidth: _PARTDESC      32 text

    fixwidth: _SPARE1        16 text
    fixwidth: _SPARE2        4 signed-integer
  }
  data-table: _df
  /offset=64
```

(... and lots more.)

word is (the offset into the text). NeXT's own Digital Librarian works like this, which means it won't allow you to do "proximity searches" (find "relational" within ten words of "Codd"), search for whole phrases, or weigh the relevancy of a returned document on the basis of how many times a particular word or phrase occurs. Topic does all this and more — Boolean searches of course ("marsupial OR reptile") — but the name of the product derives from the way it can also search on "topics", complex clusters of words and phrases representing concepts.

The words and topics are related to one another hierarchically in family trees of topics, sub-topics, and sub-sub-topics extended as far as necessary to define the particular family of ideas on which you are trying to home in. Very useful if you regularly need to profile a sea of electronic documents into predetermined subjects in which you're interested.

My chief use for Topic has always focused on the basics, like being able to combine fixed field searches with free text searches ("Linus Torvalds and source = PCW"). Fundamental to any text retrieval system, in my humble opinion, is the ability to search on one or more date fields, a feature that's often seriously neglected.

Currently, the server end of my Topic system is running on my Windows NT box but fundamentally it's still UNIX software at heart. Windows software would use a GUI to launch and configure the indexing and

retrieval engine which would certainly be nice and simple, but would inevitably restrict the possibilities. Topic launches from the command line (obviously under Windows NT you can knock up a few icons backed by batch files if you want to make it look pretty) and uses command line parameters and this nest of plain ASCII config files to define exactly what you want to happen. How the text is broken into fields, what to do with those fields, where the main document starts and stops, what kind of filters to apply, how to tune the indexing and so on are all defined by the config files.

I haven't said anything about the client end, which is the bit the user sees. Verity has traditionally offered a choice of client-end packages to cover all the main operating systems and inevitably they've all worked slightly differently and been out of phase in their versions. The solution Verity has come up with is, as far as I'm concerned, near Nirvana and The Future of Computing. Many software and hardware manufacturers are doing it now in various ways. It works, it's simple and it's delightfully cross-platform. I'm talking about web browsers, of course.

There's at least one for every operating environment these days (NeXT has a choice of four or five but that's because browsers were invented on NeXT!). Forget the browser wars as Netscape and Microsoft haggle over advanced features. Keep it simple: stick to basic HTML 2 conventions

and the network is your oyster. What this latest implementation of Topic does is offer an extension at the server end that works alongside your regular web server. I'm using Microsoft's Internet Information Server but any server with a common gateway interface (CGI) can do the job.

You create an HTML query form which can be squirted across the network, collect the query through any browser on any operating system and return the result list as a second HTML page. The result list contains skeleton details about each hit, combined with an HREF pointer to the document itself. Like the server, these client pages are all capable of being tailored via ASCII files. The neat thing is that all the gubbins is kept together at the server end. No complicated client software or configs to distribute to each workstation. All each client needs to know is the web address that gets the initial query page started. I've gone on about this at some length, partly because I'm in the heady throes of getting it up and running (extraordinarily painlessly, as it happens), but also because I'm sure we're going to hear more about this "thin client" style of computing in the near future.

Linux kernel compilation while-u-wait

Here's another twist on the browser: use it to compile your Linux kernel! Probably the most alarming thing that migrants from off-the-peg operating systems like Windows have to face when they install Linux for the first time is the suggestion they recompile the kernel. This is because a typical Linux startup will come stuffed full of drivers for all sorts of peripherals and bus connectors you don't actually need.

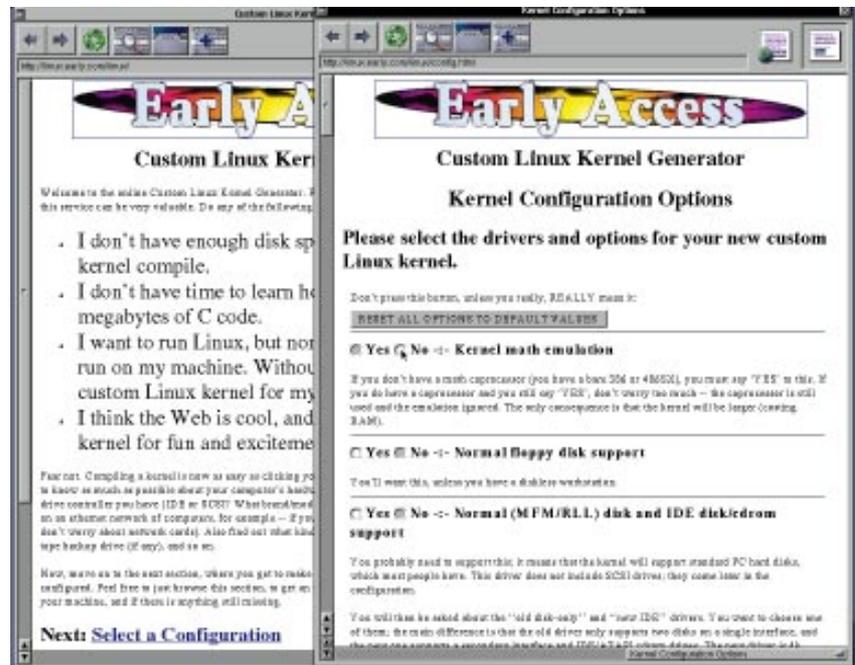
What you'll see on the console at boot time is a warning to the effect that memory is tight. You probably don't need to do anything about this straight away but eventually you'll want to slim down the kernel to only those features you need. You might also want to recompile so as to bring your kernel up to a later version. (You can check the kernel version by running "uname -a" from the command line. But if it is earlier than 1.2.13, then it's a little long in the tooth.)

I was impressed (read "terrified"!) when first presented with this kernel compiling challenge: building an entire operating system from source code isn't something you take lightly. But tens of thousands of Linux users have done it, many on a regular basis. It's actually easy because the whole

Fig 3 The Early Access Web Site at linux.early.com/linux is another cross-platform web proposition which allows you to build yourself a Linux kernel to your own specification over the internet from any machine capable of running a simple browser, in this case my NeXT machine

process is driven by a configuration file you submit to the UNIX "make" utility. Your system will arrive with the configuration file already written for you, and it will carefully trot you through a Q&A session to find out what kind of kernel you need. Fill in the answers and "make" will proceed to create your new kernel on the spot.

A couple of bright Linux hackers have taken this all a step further with a system that allows you to compile a hand-tailored kernel even though you're not running Linux. Ed Mackey and John Early have devised the "Early Custom Linux Kernel Generator", a web page that supplies you with a collection of radio buttons and tick boxes to collect details of the configuration



you want. The web site then compiles your kernel to order and delivers to your system. Look for the Early Access page at linux.early.com/linux.

Readers write...

Sevan Janiyan emailed me from Hove last month with several questions that come up often enough to air them here: "I'm 16 and very interested in Linux. I installed Linux from your cover CD a while back but I'm having problems running my Pioneer quad-speed CD-ROM drive. Is there any ftp or www site from where I could download the drivers for it? The second problem is using the `mcopy` and `mdir` commands. I can view directory listings of floppy disks in MSDOS format but how do I switch to the floppy drive? Is there any way of upgrading Linux by downloading the kernels or something like that?"

Sevan doesn't say which model of Pioneer drive he has. As far as I know, the SCSI versions of the Pioneer drives are Sony-compatible and should be catered for in the standard kernels. Anyway, the best place to look for details is in the Linux CD-ROM How To which you can pick up from www.caldera.com/LDP/HOWTO/. The LDP, or Linux Documentation Project should be the first port of call for this (or practically any) kind of advice about Linux.

The DOS drives question comes up all the time. DOS and Windows users expect to get to the floppy drive straightaway, but in Linux, as in Unix generally, you need to mount a device before you can access it, although this can be set up as an automatic mount once you know what you're doing. The mounting process can be quite complicated, depending on the device, but mounting is one of the keys to the immense flexibility of the Unix family of OSs.

Beginners will need to learn about devices and, of course, about the mount command. The floppy disk device is generally called `fd0` (or `fd1` etc., depending on how many `fd` drives you have). The place to start learning about that is the "man `fd`" command. Similarly, "man `mount`" will give you the basics of the mount command, although this is tricky stuff and you'll probably need to delve into those How To's.

Generally speaking, you won't expect to get the very latest versions of Linux on a cover-mounted CD, not because the magazine production people are trying to short-change you but because the process of making CDs and preparing them for distribution takes time. The place to look for the latest kernels on the internet is www.crynwr.com/kchanges/, which is where kernel evolution has traditionally been tracked from. But a less academic approach for beginners is www.gulf.net/~spatula/linux/kernel.html, which will lead you to full information on where the latest kernels are and what you need to do to build them.

There's a pointer to the Easy Access site from there, too. One of the reassurances that makes compiling a new kernel less than totally terrifying is the fact that you can have several different kernels lying about on your system, with a choice of which one you boot into at any one time (via a boot loader like Lilo). Provided you can always get back to a standard kernel, this can make experimentation with new versions fairly painless. My personal tip for Sevan and others is: don't get involved with the very latest experimental kernels unless you want to experiment with them. I settle for older, known kernels that work and support the hardware I use. Then I can get on with the stuff I want to do and don't lose any sleep about the code that's holding it all up.

SCO Open Server opens up

Damn, I'm out of space and I did want to say something about SCO making its Open Server version of UNIX freely available for educational purposes. This isn't quite the Free Software Foundation flavour of freeness that you get with distributions like Linux because Open Server comes with restrictions (you can't use the free version commercially) and it isn't supplied complete with source code.

Even so, it's a really big deal that this pioneering company, the first to put UNIX on Intel chips, has seen the light, or at least glimpsed the dawn. You can find all the details on the SCO web page at www.sco.com and download the software from there, or get it on CD-ROM for a (small) handful of dollars. More about this next month, by which time I hope I'll have got hold of it and installed it.

Windows wisdom

I seem to get an inordinate number of emails asking about Windows problems. Dear people, this isn't fair. I come here to get away from Windows. It's all summed up by a sig I came across recently in the comp.sys.be Usenet conference:

Customer: "I'm running Windows 95."

Tech Support: "Yes..."

Customer: "My computer isn't working now."

Tech Support: "Yes, you said that."

PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. He can be contacted at bidmead@cix.compulink.co.uk



Trouble and strife

Why can't you use Windows NT Workstation to support a third-party web server? Because of a legal limitation imposed by Bill Gates. Chris Bidmead tells the sorry tale.

As I write there's a row going on — I hope resolved by the time you read this — about the licensing limitations that Bill Gates is putting on the use of Windows NT Workstation. Initially, the beta release of Windows NT Workstation version 4.0 enforced these legal limitations with a software restriction that constrained the number of unique IP addresses that could contact a web server to ten or fewer in a ten-minute period. In response to the howl of outrage from customers and competitors, Microsoft removed this restriction from the final release version of Windows NT Workstation, but retained the legal limitation set out in the software licence.

Effectively, this means that, although it is functionally capable of doing so, you can't use Windows NT Workstation to support a web server from a third party like Netscape. If you want to do that, you have to pay three times the price (\$999 as opposed to \$290) to buy Windows NT Server. Strangely enough, if you do this, Microsoft will throw in its own web server free of charge. Which means you'll probably be reluctant to spend an extra \$300 with the likes of Netscape, buying its FastTrack web server. Netscape retaliated by sending a letter to the Justice Department complaining about what it alleged was a violation of the US anti-trust laws. The company's attorney, Gary Reback, said that Microsoft was deliberately crippling Windows NT Workstation as a way of encouraging customers to buy an all-Microsoft solution. My first reaction to this was to shrug and say, well, it's Bill's ball-game and if that's how he wants to sell Windows NT, it's entirely his affair. More power to Linux, which will not only run



Left This is GateKeeper, the PPP front end written for NeXTStep by Felipe A. Rodriguez and distributed by freeware. **Below** GateKeeper isn't just a pretty face. As you can see here, the graphical front end offers comprehensive (and comprehensible) access to the deep mysteries of PPP, and there's tons of online help.



Netscape's web server, but also in most distributions comes with its own GNU web server, Apache. No restrictions, and no charge. But the issue goes rather deeper than that. It centres on TCP/IP, the open network connection protocol on which the internet is founded. Tim O'Reilly, who heads up the O'Reilly publishing company, evidently cares about preserving open

computing. His company built its reputation by publishing highly-regarded books about Unix, and a more recent crop of books about Windows NT has been instrumental in establishing the credibility of Microsoft's new operating system in the market.

Here's what the Windows NT Workstation licence says: "...You may permit a maximum of ten computers to

connect to the Workstation Computer to access and use services of the software product, such as file and print services and peer web services. The ten connection maximum includes any indirect connections made through software or hardware which pools or aggregates connections. "My italics. Putting a licence restriction on file and print services is standard practice for LANs using a proprietary transport protocol like IPX or NetBIOS.

But the internet isn't a LAN. It's truly open territory, and it uses an open transport protocol. You don't know how many people are going to be visiting your web site, and if you had to issue client licences to everybody before they could do so, the web would never have got started. Tim O'Reilly points out that the legal restriction is even more stringent than the original software dongle. "The limitation has been expanded, from 'ten users in ten minutes' (the original limitation) to 'ten users (period)'. We believe that Microsoft's position amounts to nothing more than a 'land grab' in the uncharted territory of the internet."

He counters my comeback that Bill has a right to do what he wants with his own operating system by pointing out that TCP/IP, which is where the restriction lies, isn't his [Gates], or anybody's, to mess around with like this. "TCP/IP is not a Microsoft product, and I don't believe Microsoft has the right to tell application vendors and users what they can and can't do with it. TCP/IP is a fundamental service for internetworked systems."

To hammer home this argument he concludes: "If you accept that Microsoft has the right to tell users how many sockets their applications can have open, you must also accept that they have the right to tell users how much memory their applications can use, or how much processing power."

Netscape is angry because its business is being threatened. Tim O'Reilly is angry because a principle is being violated. I've been known to get too hot under the collar about issues like this in the past, but this time the most action it gets from me is a quick shrug. It's not that I don't believe O'Reilly is right. Of course he's right. The point is, Microsoft's behaviour over this licensing business simply confirms what I've long suspected. When it comes to the internet, despite his much-publicised 180-degree turn at the end of last year, Bill really doesn't get it. As the old Bob Dylan song says: "Something is happening, but you

don't know what it is. Do you... Mr Jones?"

At the end of 1992, Microsoft launched a product called Windows for Workgroups. It was NetBIOS joining a bunch of local machines together. The machines all ran Windows, and Bill owned Windows and he owned NetBIOS. It seems like a hundred years ago. Only a few months after that launch, the Mosaic web browser arrived and we all clamoured to get onto the internet. Now, for the first time, our desktop machines were properly connected. Globally. And it no longer mattered what operating system you were running. TCP/IP wasn't just another protocol, like NetBIOS, added to the operating system. It was the other way round. The operating system you happened to be running on the machine in front of you became simply the interface to the main action, carried out in the vast worldwide arena called TCP/IP. It was as if we'd all gone to the cinema and had been sitting in the dark for fifteen years, thinking how comfortable our seats were, or not. And then the film started.

This is what Bill doesn't get. By imposing these kind of restrictions on Windows NT Workstation, he is a tail trying to wag the dog. Either his customers will walk, across to Linux, perhaps, or a third party like Netscape will supply a TCP/IP stack not written by Microsoft that will arguably take the right to restrict clean out of Bill's hands.

Or perhaps people will just ignore the licensing issues. In which case, Bill may well send out agents across the internet to sniff out offenders, and maybe drop writs on their web servers in the form of a Word for Windows macro virus. It will all get very silly and make more lawyers rich. I return to my first thought: this has got to be good for Unix.

Ray Noorda still going strong

Microsoft is under attack now from another quarter — a writ from Ray Noorda, the programmer turned billionaire who built Novell up from nothing all through the eighties and is now the force behind Caldera. Caldera has just bought NDOS, the Novell version of DOS that was originally developed by Digital Research Inc. DRI was the company whose 8-bit CP/M operating system helped microcomputers grow from toys to business tools before the arrival of the IBM PC.

Caldera's claims about Microsoft's "various unfair and predatory acts" by which the industry was force-fed with MSDOS to the detriment of competitive products all

p262 >

Caldera

Last month I mentioned System Commander, the handy multiple boot utility distributed in this country by POW!, a Dorset-based distribution company. It's headed up by (and I suppose named after) an old acquaintance of mine called Daniel Power. One of the best things about writing a column is that you get old chums popping up out of the blue like this, and it's doubly pleasing when they turn up with a product as good as System Commander.

But it's best of all when I hear from people who have actually been reading the column. Daniel gave the game away with a follow-up missive, which I'm going print in full here just for the hell of it.

From: dpowera@cix.compulink.co.uk (Daniel Power)

Subject: System Commander

To: bidmead@cix.compulink.co.uk

Cc: dpowera@cix.compulink.co.uk

Reply-To: dpowera@cix.compulink.co.uk

"Are you interested in Linux derivatives/Intranet servers? I am in the process of launching a reworked version of Linux with an X Windows interface, free Netscape etc. The installation is simple, the product more accessible to the end user. It will cost 99 pounds. The product is from a company called Caldera. I don't know if you know anything about them but it is a project that was dropped from Novell and is now funded by the Ray Noorda family trust fund."

I wrote back to tell him to say that, well, yes, my readers and I have been tracking Caldera for about 18 months now, and know it quite well. But a UK source is always useful, especially if you're offering some kind of support. Unfortunately, POW! isn't in a position to do that yet, so the product is being distributed through Lasermoon for the time being. Here are the details:

Lasermoon <sales@lasermoon.co.uk> Phone 01329 834944

Caldera Network Desktop £70 (plus VAT)

WordPerfect plus Motif Licence £166 (plus VAT)

Internet Office Suite £250 (plus VAT). Includes WordPerfect, Nexus spreadsheet and Zmail

Daniel tells me that the best email address for his own company is sales@pow-dist.co.uk, and they have a website at www.pow-dist.co.uk. Full details of Caldera are on the Caldera web site at www.caldera.com.

Noorda: Has issued a writ on behalf of Caldera, claiming that Microsoft, with the total domination of MSDOS, virtually hijacked the eighties



through the second half of the eighties and beyond will strike a familiar chord to anyone studying the current Windows NT Workstation licensing affair. Ray Noorda seems to have a couple of genuine business aims here, as well as punishing Microsoft for its past misdemeanours. He wants to be able to reinstate Digital Research's DOS as part of the Caldera Linux distribution, strengthening its powers as a "network desktop"; and he wants guaranteed disclosure by Microsoft of "all APIs for any operating system it produces, as well as any modifications, enhancements, updates, or new versions of such operating systems at the time that such products are released for beta testing." This fits with Noorda's plans to

enhance Linux to the point where it will be able to run Windows applications. The ramifications of this are vast, and there isn't space to rattle on about it here.

If you want to know more, the whole legal case, with a history of "Microsoft's Growth and Domination" and Caldera's "Claims and Prayers for Relief" can be read on <http://www.caldera.com/news/complaint.html>.

Readers write

Long-suffering readers will remember that back in May of this year I ran into an absurdly simple problem with Unix that may well baffle anybody coming to the environment from DOS or Windows. I'm talking about the business of batch renaming files. In Unix you don't, philosophically, rename files — you move them. The same command, mv, is used either to shift a file from one directory or another, or to change its name. This, in itself, takes a little getting used to, but it gets worse when you start looking for an equivalent of DOS shortcuts like REN *.BAT *.BAK. Because Unix doesn't handle batch renaming like this.

The solution I stumbled on was to use the FOREACH command — roughly the equivalent of the DOS FOR batch

command. It works like this:

```
> foreach f ( *.tiff )? mv $f
$f:r.tif? end
```

This works fine, but is a little long winded because the foreach command is actually a mode that you enter, throwing up its own prompt (the question mark) and requiring the keyword "end" to exit. Great for Unix buffs, but what my simple, DOS-educated spirit craved was a one-liner. As luck would have it, a Unix buff has responded to my plea. The solution, as you've probably guessed, is a shell script that takes care of the multiple renamings and can be tailored to handle the parameter you pass it to suit your taste.

I like the script that Dr Rich Artym <rartym@galacta.demon.co.uk> has sent me, because it exemplifies one of the things that appeals to me most about Unix: the ability to make it work the way you want it to work. Of course, you need to understand something of the arcane machinery below the surface to do this properly. Two O'Reilly books, *Using csh & tcsh* and *Learning the bash Shell*, have been invaluable foglamps for me as I grope my way through the murk.

The Artym solution to batch renaming can be found on this month's cover CD.

Where does UNIX begin and end?

After much discussion, we've settled on calling this column just plain "UNIX". I suppose this avoids any ambiguity (one proposal from a PCW staff member was "Hardcore Computing"), and certainly the venerable operating system is the basis of my thoughts and researches here.

But I wouldn't want you to feel that we are in any way ghetto'd by the new title. The spirit of the column reaches far, er, rambles, some might even say, over a wide range of computing issues that tend to be neglected by the Windows-centric mainstream computer press. A great deal is happening "outside the Gates", as I tend to think of it, and some of it is even happening here on my own network. Not that Windows is excluded — how to keep the really good stuff like my NeXT workstation connected to the merely useful world of Windows is an issue I'll continue to pursue. And I hope you'll stay with me as I do.

PCW Contacts

Chris Bidmead is a consultant and commentator on advanced technology. He can be contacted at bidmead@cix.compulink.co.uk