



linux.conf.au 2003

at The University of Western Australia

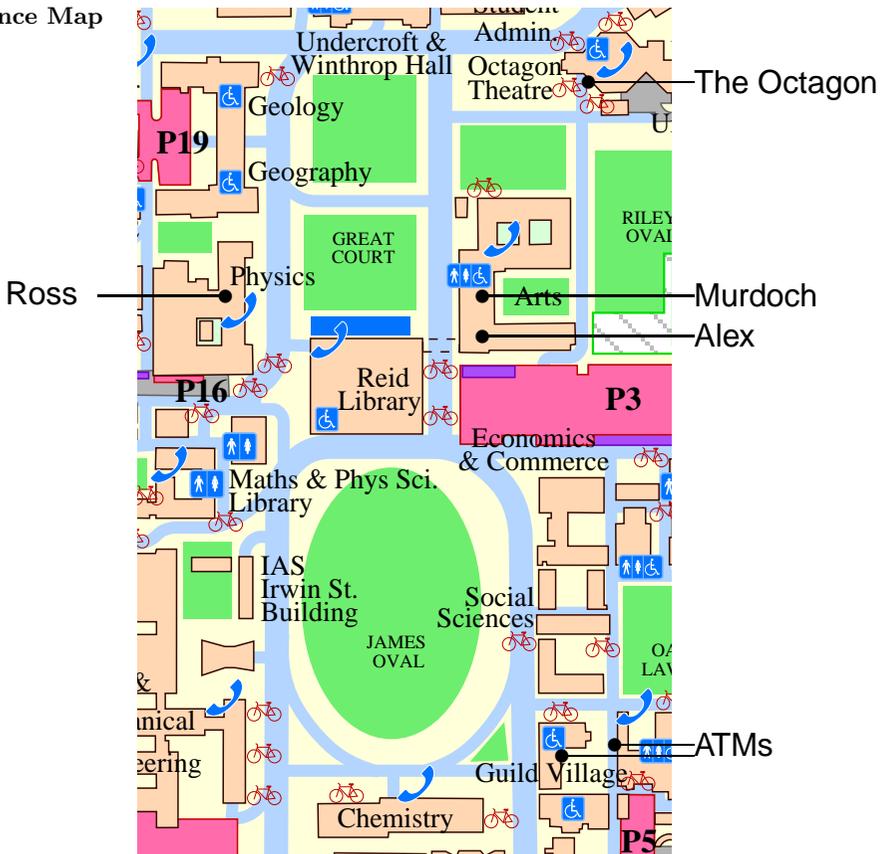
January 22nd to 25th, 2003

Perth, Western Australia

In this booklet you will find:

About Linux.conf.au 2003	3
Our Generous Sponsors	3
Programme	7
Abstracts	11
Transport	30
Eating	32
Perth & Entertainment	36
Notes	37
Our Professional Delegates	38

Reference Map



There is a map of the whole campus and surrounding area in the very middle of this booklet.



About Linux.conf.au 2003

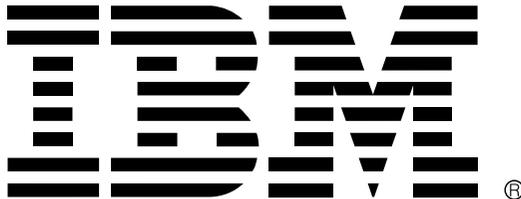
Linux.conf.au is the Australian Linux Technical Conference hosted annually by Linux user groups across Australia. The conference runs between the 22nd and 25th of January 2003 in Perth, Western Australia. It is a four day event comprised of keynote speeches, paper presentations, tutorials, a formal dinner, birds of a feather sessions and informal gatherings.

Continuing on from Melbourne, Sydney and Brisbane, 2003 brings hoards of hackers covering the width and breadth of the Open Source community. Linux.conf.au has a very high standing in the international community for having a technically focused yet relaxed conference schedule. All presenters are selected from the community and are of the highest calibre.

Our Generous Sponsors

... without their help, Linux.conf.au 2003 would not have been possible.

Penguin Sponsor



Major Sponsors



Computer Associates™



Special Sponsors



THE UNIVERSITY OF
WESTERN AUSTRALIA

Venue Sponsor



Regional Delegate Program Sponsor



Online Media Sponsor



Online Credit Card Processing

Supporters

These organisations have donated services or products to assist us. Thank you!

Boffins Bookshop

Canning College

Celebrity Enterprises

City Printing and Design

Cold Fusion User Group WA

Colourbox Digital Printing

Electrical, Electronic and Computer Engineering, UWA

Pelican Manufacturing

Perth Convention Bureau

Thawte

University Computer Club



Programme

Wednesday 22nd January 2003

Wednesday's sessions are each 3 hour tutorials.

	Alexander	Murdoch	Ross
8:00 – 9:00	Registration		
9:00 – 9:30	Welcome, Orientation - Octagon Theatre		
10:00 – 11:30	PHP Introduction <i>Rasmus Lerdorf</i>	Using autoconf, automake and friends <i>Malcolm Tredinnick</i>	Using Sweep: Fun with Scrubby <i>Conrad Parker</i>
11:30 – 11:45	Morning Tea		
11:45 – 13:00	PHP Introduction (cont.) <i>Rasmus Lerdorf</i>	Using autoconf, automake and friends (cont.) <i>Malcolm Tredinnick</i>	Using Sweep: Fun with Scrubby (cont.) <i>Conrad Parker</i>
13:00 – 14:30	Lunch		
14:30 – 16:00	PHP Advanced <i>Rasmus Lerdorf</i>	The Unreliable Guide to Kernel Hacking <i>Paul "Rusty" Russell</i>	The Ins and Outs of PostgreSQL <i>Gavin Sherry</i>
16:00 – 16:15	Afternoon Tea		
16:15 – 17:30	PHP Advanced (cont.) <i>Rasmus Lerdorf</i>	The Unreliable Guide to Kernel Hacking (cont.) <i>Paul "Rusty" Russell</i>	The Ins and Outs of PostgreSQL (cont.) <i>Gavin Sherry</i>
Evening	Professional Delegates' Networking Session		

"If you want to travel around the world and be invited to speak at a lot of different places, just write a Unix operating system."
– Linus Torvalds



Thursday 23rd January 2003

	Alexander	Murdoch	Ross
8:00 – 9:00	Registration		
9:00 – 10:30	Advanced Network Techniques <i>Andrew Tridgell - Octagon Theatre</i>		
10:30 – 11:00	Morning tea		
11:00 – 11:45	My computer is bigger than yours! <i>Martin Schwenke</i>	ARQuake - Outdoor Augmented Reality Gaming <i>Wayne Piekarski</i>	Pretend Systems Doing Real Work <i>Dan Shearer</i>
12:00 – 12:45	Making the right sort of difference: bugs and what to do with them <i>Telsa Gwynne</i>	\$100,000 OUT OF THE WINDOW? YOU CANNOT BE SERIOUS! <i>Philip Hart</i>	Dynamic Probes - Debugging by Stealth <i>Suparna Bhattacharya</i>
12:45 – 14:00	Lunch		
14:00 – 14:45	Sweep: Real-world audio editing on Linux <i>Conrad Parker</i>	Teaching Software Engineering using Open Source Software <i>David Glance</i>	HAM Radio <i>Hugh Blemings</i>
15:00 – 15:45	more, More, MORE! - Porting Hints and Horror Stories from the Debian Project <i>Bdale Garbee</i>	Dynamic Feedback in LVS <i>Jeremy Kerr</i>	DHCPv6 for Linux <i>Suresh Kodati</i>
15:45 – 16:15	Afternoon Tea		
16:15 – 17:00	Q&A Session - Octagon Theatre		
17:00 – 18:00	Linux Australia AGM (optional) - Alex		
18:15 –	LCA Keysigning ¹ (optional) - Alex		
19:00 – 20:00	ISOC-AU Meeting ² (optional) - Ross		
Evening	Unplanned: free time		

¹See <http://www.linux.conf.au/keysigning.html> for details

²See <http://www.isoc-au.org.au/TCCM/>



Friday 24th January 2003

	Alexander	Murdoch	Ross
9:00 – 10:30	Herding Wild Cats and Related Adventures - An Inside Look at the Debian Project <i>Bdale Garbee - Octagon Theatre</i>		
10:30 – 11:00	Morning tea		
11:00 – 11:45	The Boston Consulting Group/OSDN Hacker Survey <i>Jeff "Hemos" Bates</i>	Putting a filesystem into a device driver <i>Greg KH</i>	Zeroconf networking <i>Brad Hards</i>
12:00 – 12:45	initramfs, klibc and early userspace <i>H. Peter Anvin</i>	PHP Printing with PDF <i>Rasmus Lerdorf</i>	Problems in Open Source Licensing <i>Jeremy Malcolm</i>
12:45 – 14:00	Lunch		
14:00 – 14:45	Towards full NTFS semantics in Samba <i>Andrew Tridgell</i>	Biometrics and Linux <i>Alexander Reeder</i>	High Availability using Reliable Server Pooling <i>Thomas Dreibholz, Michael Tuexen</i>
15:00 – 15:45	Useful kernel audits <i>Alan Cox</i>	Perdition: A Mail Retrieval Proxy <i>Simon "Horms" Horman</i>	Who wants another filesystem <i>Neil Brown</i>
15:45 – 16:15	Afternoon Tea		
16:15 – 17:00	BOFs ³ - Arts Lecture Rooms ⁴		
18:30 – 23:00	Conference Dinner - Currie Hall		

³Birds of a Feather meetings, or BOFs, are meetings of people with similar Linux related interests. These could include favourite distributions, preferred scripting languages, who you think the coolest kernel hacker is, or whether you think Alan Cox's favourite colour is blue or yellow.

⁴BOFs will take place in the Arts building. For more information, visit <http://www.linux.conf.au/bof/>



Saturday 25th January 2003

	Alexander	Murdoch	Ross
9:30 – 11:30	Movie <i>Revolution OS - Octagon Theatre</i>		
11:30 – 12:00	Morning Tea		
12:00 – 12:45	distcc <i>Martin Pool</i>	Sequel Syndrome: The GNOME Project Grows Up <i>Jeff Waugh</i>	Katie, open-source revision control system <i>Geoffrey D. Bennett</i>
12:45 – 14:00	Lunch		
14:00 – 14:45	I'll Do It Later: Softirqs, Tasklets, Bottom Halves, Task Queues, Work Queues and Timers <i>Matthew Wilcox</i>	Linux at the Little End of the World <i>Malcolm Tredinnick</i>	Design of the Portable.NET Interpreter <i>Rhys Weatherley</i>
15:00 – 15:45	The Linux Kernel Device Model <i>Patrick Mochel</i>	EggMenu <i>James Henstridge</i>	Developing network applications under UNIX <i>Adrian Chadd</i>
15:45 – 16:15	Afternoon Tea		
16:15 – 16:45	Best of #1	Best of #2	Best of #3
17:00 – 18:00	Close - Octagon Theatre		
Evening	Unplanned: free time		

Sunday 26th January 2003

Today is a free day to code, talk to people, relax and see the sights of Perth. This is not an official conference day, but if you hang around there are fireworks in the evening down by the river!



Abstracts

Listed below are the abstracts for the presentations, in the order of the programme.

Wed 10:00 – 13:00 **Rasmus Lerdorf** (Alexander)

PHP Introduction

A fast-moving thorough and quite pragmatic introduction to the world of PHP. Topics to be covered include:

- Language introduction
- Dynamic generation of images, PDF and flash
- HTTP and other interesting protocols
- PEAR
- Sessions
- Security

Attendees will come out of this tutorial with a good understanding of how to use PHP to solve a broad range of real-world problems. Some experience with HTML and the web in general is assumed. No prior PHP experience is assumed, but the tutorial does not dwell too long on the details of the language itself, choosing instead to focus on how to use the language. This means the tutorial tends to be interesting and useful even for quite experienced PHP developers.

Wed 10:00 – 13:00 **Malcolm Tredinnick** (Murdoch)

Using autoconf, automake and friends

A fairly common expectation amongst Open Source users is that they can just grab the latest tarball of package X, unpack it and then put on a blindfold and type `./configure && make && make install` - and it usually Just Works(tm). However, the magic that goes on behind the scenes to make this process work remains a mystery, even to a lot of programmers. This tutorial will try to draw back the curtain a little bit and reveal the elves going about their business as they create `autogen.sh`, `configure.in` and `Makefile.am` files for your later enjoyment.

Using autoconf and automake provides benefits for both programmers and end-users: they help with programming for portability and detecting pre-requisites for the build phase (autoconf) and allow the developer to focus on the programming without needing to spend too much time putting together a build infrastructure (automake). At least, that is how it is meant to work in theory. In practice, a fair bit of time is spent fine tuning the various files because things never quite seem to work as you first expected. Furthermore, it seems that people will still just prefer to cut-and-paste from another similar project and poke about wildly until the thing more or less works, rather than solving their particular problem. This part of a project is seen as an arcane area that is hard to comprehend.

So we are going to fix that!

“I hate programming.” – Rasmus Lerdorf.



These programs are not difficult to use if you approach it with the right mindset and a clear system for how you are going to use them. They are the misunderstood workhorses of freely available software. Building automatic configuration and build support into your project is fairly simple. Helping with somebody else's problems is not too difficult and usually educational. Hanging out your shingle on mailing lists or IRC and saying "I will debug your build problems" is a satisfying thing to do.

This tutorial will cover three fairly large areas (without trying to bore people to death):

- Writing Makefile.am files to describe a project's build and distribution requirements.
- Writing configure.in files to handle configure-time options and detect portability constraints.
- Isolating machine-specific code fragments (and using configure-time results to pick the right code).

No prior knowledge of autotools or m4 is assumed. No knowledge of specific programming languages is required -- many of the examples will talk about C programs, but we will discuss packaging, documentation, internationalisation issues and other programming languages.

Wed 10:00 – 13:00 **Conrad Parker** (Ross)

Using Sweep: Fun with Scrubby

This is an introductory tutorial about using Sweep for editing and experimenting with digital audio. You will find out how to make your Linux box make sounds that you have never heard before, and you might also learn something useful about editing audio files.

Although Sweep is quite powerful, it is an easy to use desktop application and its interface contains none of the esoteric "weird shit" common in audio software.

Sweep harbours a pesky little virtual stylus tool called "Scrubby" who will invade your mind and make you want to remix your entire CD collection in one day.

In this tutorial you will learn:

- basic audio editing
- how to use many free LADSPA effects plugins including compression, delays, distortion and filters
- immersive loop mode recording
- beatmixing and scrubbing technique
- how to play with digital sampling

WARNING: after attending this tutorial, you may never use your Linux box quite the same way again!

Wed 14:30 – 17:30 **Rasmus Lerdorf** (Alexander)

PHP Advanced

In this intermediate-to-advanced tutorial we will have a closer look at what it takes to build



"Whoa...I did a 'zcat /vmlinuz > /dev/audio' and I think I heard God..." – mikecd on #linux

a database-driven dynamic web site with PHP. Some of the topics to be covered include:

- Cool PHP tips and tricks
- Scaling PHP-MySQL sites
- Content caching
- Overall approaches to clean web site design
- Extending PHP

People who survive this tutorial should come out of there with a lot of ideas that they can immediately apply to current projects. People who attend this tutorial should be familiar with web development, PHP and have at least a little bit of experience with C programming.

Wed 14:30 – 17:30 **Paul “Rusty” Russell** (Murdoch)
The Unreliable Guide to Kernel Hacking

This tutorial is aimed at programmers with C experience who want to code in the Linux Kernel. This introduction will walk through a lot of code, but will also cover some of the practicalities and politics of Linux Kernel development.

Wed 14:30 – 17:30 **Gavin Sherry** (Ross)
The Ins and Outs of PostgreSQL

PostgreSQL is one of the most sophisticated open source relational databases. It provides excellent performance, a large number of features, adherence to international standards, ease of installation and maintenance and more.

This tutorial will:

1. Provide a brief introduction to the database itself
2. Survey the features of the database including: extensive support for SQL92 and SQL99, transactions and multiversion concurrency control, support for many hardware and software platforms, sub-selects, views, cursors, user-defined functions and types, and full backend extensibility
3. Show how to install, configure, start and stop the database server
4. Survey PostgreSQL’s SQL extensions
5. Look at the C client library for Postgres, with examples of the usage of all functions
6. Explain procedural languages – in particular, PL/PgSQL: the Postgres procedural language
7. Review the Server Programming Interface (SPI) for backend extensibility
8. Provide extensive examples of code where relevant

Thu 9:00 – 10:30 **Andrew Tridgell** (Octagon Theatre)
Advanced Network Techniques

It is a fact of life that many protocols and algorithms are less well documented than is needed for a programmer to implement in such a way that it is fully interoperable with

“First off, I’d suggest printing out a copy of the GNU coding standards, and NOT read it. Burn them, it’s a great symbolic gesture.” – from linux/Documentation/CodingStyle



existing implementations. While there are many reasons for this lack of documentation, the result is that a programmer is often called upon to dissect an existing protocol to work out how it works. In this talk I will discuss some of the more interesting network analysis techniques that myself and other members of the Samba Team have used over the years.

Thu 11:00 – 11:45 **Martin Schwenke** (Alexander)

My computer is bigger than yours!

Your computer runs Linux. It has 1 CPU, 256MB of RAM, 2 ethernet interfaces, 2 IDE disks, a CD-ROM drive and a CD-RW drive. You reboot this machine every week or two to try out the latest kernel or to mess with a new revision of the X server. You know you've got `/dev/hda1` and `/dev/hda2`, `eth0` and `eth1`, so the configuration is pretty simple. When something goes wrong with one of the components, it is pretty easy to notice that you have a problem (using, say, `logcheck`) and to remove the cover and point to the faulty component (using your index finger).

My computer runs Linux. It has 32 CPUs, 64GB of RAM, 16 ethernet interfaces, 540 SCSI disks, a CD-ROM drive and a CD-RW drive. The contract I have with my customer means that I'm allowed about 8 hours of unscheduled down-time per year. Luckily, everything is hot swappable. I need support for persistent device names or things are going to get ugly. I need to anticipate hardware problems, so I need some fancy software that wades through the messages that I get from all of this hardware. When I've detected that something is wrong with one of the components I'm going to need some diagnostic tools that let me figure out the exact physical location of the faulty component.

This paper looks at persistent device naming and hardware inventory requirements for Linux. There are quite a few solutions for persistent device naming but most, if not all, have disadvantages. There are several distinct stages in the hardware inventory process: you need to consider how you're going to obtain, store and use the information about the hardware. Should persistent naming use hardware inventory data or vice-versa? The discussion involves driverfs, Open Firmware device-trees, various userspace methods for retrieving hardware inventory data, some serviceability infrastructure that helps to make AIX a serious operating system... and much more.

Thu 11:00 – 11:45 **Wayne Piekarski** (Murdoch)

ARQuake - Outdoor Augmented Reality Gaming

This talk will present the ARQuake⁵ system developed at the Wearable Computer Lab of the University of South Australia. Using a head mounted display, high resolution tracking devices, a mobile backpack computer, and a modified version of the infamous Quake game, users can go outside and play games against virtual and real opponents in the physical world.

We discuss the modifications which were made to the original GPL'd Quake source code,

⁵<http://wearables.unisa.edu.au/arquake/>



“You've got to look cool, People don't want to walk around advertising that they're a cyborg dork.” – Wayne Piekarski

implementing new features such as interfaces to absolute body tracking devices, real time overlays from a head mounted video camera, and the removal of keyboards and mice since these devices are not available outdoors.

We will be showing videos of the system in action, and also plan to bring the entire Tinmith mobile augmented reality backpack to demonstrate the technology to the masses at the conference.

Thu 11:00 – 11:45 **Dan Shearer** (Ross)
Pretend Systems Doing Real Work

Simulated Systems let some things be done which are not easily possible using other means. Have you ever wanted to:

- fork an entire running system (where system = hardware + OS)
- so you can follow both branches during decisions in testing runs?
- build a simulated network starting from your existing firewall and server configurations?
- build a beowulf using just one computer?
- email your whole computer to a friend?
- set breakpoints on your hardware?
- write code for hardware that doesn't exist or you haven't got?
- boot your own personal 32-processor UltraSPARC, S/390 mainframe, Itanium or completely obsolete 16-bit architecture?
- create memory errors in your hardware and observe what the OS does?

There are some interesting problems and questions:

- what about preserving state when crossing over between real and pretend systems?
- licensing... some big companies get upset when free (and non-free) simulated systems are used to run their OS or other software
- how can we know if the simulated system is a good implementation of the original?

Thu 12:00 – 12:45 **Telsa Gwynne** (Alexander)
Making the right sort of difference: bugs and what to do with them

A romp through bug-reporting adventures, what I learned from them, and whether it made a difference.

It will probably include: spotting bugs, narrowing them down, working out where and how (and whether!) to report them, what you do next, and (for developers) how to put your users off ever reporting them again; and (for users) how to ensure that no-one is ever going to look at your bug.

The best tools I have found for all of this will also make an appearance. Some of them may be surprising.

“I see once again I am hindered by some acquaintance with facts.” – Telsa Gwynne



Thu 12:00 – 12:45 **Philip Hart** (Murdoch)

\$100,000 OUT OF THE WINDOW? YOU CANNOT BE SERIOUS!

A city council in New Zealand is implementing a real-time public information system for their buses, with arrivals and departures being displayed at bus stops, and next stop information inside the buses. This project started in April 2002, with the first phase being completed in November 2002. The displays are driven by a single-board computer. The original plan was to use Windows CE, but the cost for this was 10 times over the planned budget. Additionally, the time to build a suitable version of Windows CE was 6 weeks, and this would have caused unacceptable over-run of time for the project.

A Linux expert was brought into the project. The application was originally written in C in a Windows environment, and this was ported to Linux. The Linux expert plumbed the application into the new environment. The initial reaction of the contractor to the change to Linux was one of horror, followed by amazement at what had been achieved in two weeks. The first phase of the project was successful.

The two major factors that led to the success are (i) writing the application with portability in mind, and (ii) employing people with the appropriate skills.

The major lessons learned are described. This is followed by an evaluation of the likely relative market positions of Windows and Linux from a competitive viewpoint over the longer term.

Thu 12:00 – 12:45 **Suparna Bhattacharya** (Ross)

Dynamic Probes - Debugging by Stealth

Dynamic Probes - Debugging by Stealth - Update and Work in Progress Dynamic Probes has been around in the Linux Community for a couple years. Originally it was conceived as a stealth system debugger - i.e. one that you might be more happy to use in a production environment - its capability and application has expanded way beyond these modest beginnings. Since its initial release it has undergone continual development and enhancement. During 2002 DProbes has undergone major surgery: not only have we enhancing its capability, but we have streamlined the kernel patch and making it more generic. In June 2002 we released a major usability tool - the DProbes C Compiler (dpcc). DProbes was also ported to Power and zSeries architectures in 2002. Finally but not least DProbes was made a ready candidate for the 2.5 kernel and for several major distributions and has in fact been accepted by TurboLinux and MontaVista at the time of writing. So much has changed it seems an opportune moment to report on recent developments and work in progress. In this paper we will cover the following topics:

- The essential mechanisms that allow DProbes to be a system-wide debugger (kernel and multiple user contexts).
- How global probes are implemented using breakpoint mechanisms but without impacting the physical memory footprint through SOW. The techniques used here are applicable to kernel debuggers in general and would provide a means of allowing them to become system-wide debuggers.



“Bubblesort rules! ... within a very limited domain... :-)” – Martin Schwenke

- The new patch organisation into Kernel Probes; User Probes and Watchpoint Probes will be discussed in detail. These patches provide interfaces that allow any kernel module to define a probe and probe handler. It is by this means that we have been able to separate as an entirely optional component the original Reverse Polish Notation (RPN) interpreter that was previously an integral part of DProbes.
- The new pagepoint extension: we have the option of exploiting the processor's paging mechanism to extend the number of concurrent watchpoints beyond the few that are permitted by processor debugging hardware.
- The new dpcc compiler: this allows probepoints to be defined using c-like language that references the variables and structures present in the code which is being probed. The Probe definition appears as an insert to the original source, without the need to edit or recompile the original source.
- We will also spend some time discussing how DProbes is able to be used as a general debugging engine to trigger crash dumps and core dumps from arbitrary code locations; to provide an on-the-fly tracing and logging mechanism (dynamic trace), which has found particular favour with some kernel developers whose needs was to add temporary printks for debugging; and finally we will show how DProbes can be used to exercise error code paths through fault injection.

Thu 14:00 – 14:45 **Conrad Parker** (Alexander)

Sweep: Real-world audio editing on Linux

This paper outlines my recent work on Sweep, a digital audio editor, as part of a contract with Pixar Animation Studios (where it is now used in animation production) and further work supported by the CSIRO.

Sweep is Free Software, available under the GNU General Public License.

This presentation will include:

- a brief demonstration of Sweep's capabilities, including multichannel editing, recording, and interactive scrubbing
- an overview of Sweep's internals, including the implementation of vinyl style "scrubbing" and monitoring of playback latency
- an introduction to writing Sweep and LADSPA⁶ plugins
- integration with Ogg Vorbis, Speex, MP3 and PCM formats

Thu 14:00 – 14:45 **David Glance** (Murdoch)

Teaching Software Engineering using Open Source Software

This talk outlines the tools, languages and platforms that can be used in the teaching of Software Engineering and Computer Science at a University level. This will draw on the experiences (and future plans) for using Open Source software at UWA in the department

⁶LADSPA is the Linux Audio Developer's Simple Plugin API, a standard for audio effects and synthesis plugins developed by the linux-audio-dev community.

"Oath, it is. The dress is 'outrageous bow tie', not sure if any other clothing is required." – Conrad Parker



of Computer Science and Software Engineering. Since the aims of our teaching at CSSE are to teach best practice in all aspects of the software development life cycle, it is anticipated that the skills and knowledge accumulated by the students will be directly transferable to both a corporate and Open Source development environment. In addition to using Open Source software in a corporate style development process, aspects of the true Open Source development model is also taught.

Thu 14:00 – 14:45 **Hugh Blemings** (Ross)
HAM Radio

Want to know more? Then come on down!

Thu 15:00 – 15:45 **Bdale Garbee** (Alexander)
more, More, MORE! - Porting Hints and Horror Stories from the Debian Project

Bdale Garbee is the Debian Project Leader, and over the years helped initiate the ports to five of the eleven architectures Debian supports. Along the way, it became clear to Bdale that solving portability issues improves overall software quality, and is therefore something everyone should care about! This talk will look at how the combination of source packaging, an autobuilder network, and related policies enable the Debian distribution to maintain over ten thousand packages on eleven released architectures. A few horror stories about problems encountered and what it took to fix them provide hints on how to write software that will run on many architectures.

Thu 15:00 – 15:45 **Jeremy Kerr** (Murdoch)
Dynamic Feedback in LVS

With the popularity of the Internet increasing, the demands on network servers (particularly HTTP servers) have grown to the point where a single node is not sufficient to process all incoming requests for popular services.

Two possible upgrade pathways exist - scaling up and scaling out. Scaling up involves upgrading a single computer (by way of adding more processors or memory), while scaling out involves adding other computers to run in parallel. While scaling up seems the simpler option, there are disadvantages involved:

- Scaling up does not return a linear performance improvement
- A single server must be taken offline for upgrades
- No fault tolerance is present with a single server

The advent of load balanced servers has allowed the implementation of clusters of independent servers acting as a single Internet presence, with the load shared between them. This allows larger amounts of requests to be served, as well as reduced downtime and greater upgrade flexibility, while keeping costs down. Scaling out has been shown to provide a linear performance response.



“The memory management on the PowerPC can be used to frighten small children.” – Linus Torvalds

There are many load-balancing solutions available, in both hardware and software implementations. The Linux Virtual Server (LVS) Project is an extension to the linux kernel, providing a software load-balancing solution for IP-based services. The goal of the LVS Project is to ‘Produce a high-performance and highly available server for Linux based on clustering, which provides a good scalability, reliability and serviceability.’

When requests are received by the load balancer, a decision must be made to find a server to allocate the request to; ideally, each server will be equally loaded. To achieve this, a dynamic feedback system is being developed to report the load of each server to the load balancer, so that real-time load data can be used in the allocation decisions. The system being developed is based around the Network Element Control Protocol (NECP) - a daemon is run on each server node to report the current load (a configurable metric) to a daemon running on the load balancer, which processes the load data and updates the decision tables appropriately.

By testing this in an experimental load-balancing situation, the advantage of implementing a dynamic feedback system in a load-balanced server cluster can be measured; these results will be presented at the conference.

Thu 15:00 – 15:45 **Suresh Kodati** (Ross)
DHCPv6 for Linux

Dynamic Host Configuration Protocol for IPv6 defines the communication mechanism between DHCP servers and clients in an IPv6 network, whereby servers pass network configuration parameters like network addresses, DNS information etc., to clients. DHCPv6, an evolving IETF standard, uses the stateful version of address configuration mechanism defined by IPv6 standards which provides better control over the allocation of IPv6 addresses compared to its counterpart, stateless address auto-configuration. This paper explains our work related to the implementation of a DHCPv6 client, server and relay for Linux as defined by IETF standards.

We discuss the architecture of the DHCPv6 solution including (1.a) design of state machine for DHCPv6 server, (1.b) address delegation policy, (1.c) maintenance of various addresses and states in persistent and nonpersistent memory, (1.d) server’s lease database and its scalability, (1.e) optimizing server performance and responsiveness under load (2.a) design of state machine for client (2.b) client address request policy, (2.c) storage of addresses at various phases of client (3) state machine for relay and (4) process of upgradation of the implementation to the latest drafts/protocol versions.

This work has become a part of USAGI⁷ project which aims to deliver the production quality IPv6 protocol stack for Linux, tightly collaborating with other projects and volunteers from various organizations.

⁷<http://www.linux-ipv6.org/cvsweb/usagi/src/dhccode/>

“Why do we have to hide from the police, Daddy?”
“Because we use vi, son. They use emacs.”



Fri 9:00 – 10:30 **Bdale Garbee** (Octagon Theatre)
**Herding Wild Cats and Related Adventures -
An Inside Look at the Debian Project**

As a long-time member of the Debian Project and the project's current elected leader, Bdale Garbee has a unique perspective on how Debian actually works. This talk will begin with a history of Debian, how the project is structured, and how Bdale came to be involved in Debian. It will then touch on how Debian manages to deliver more packages for more architectures than any other distribution, and other project attributes that have made it such a success. If you're thinking of joining the Debian project, or are just curious about what it is, how it works, or why it inspires such enthusiasm... this is a talk you won't want to miss!

Fri 11:00 – 11:45 **Jeff "Hemos" Bates** (Alexander)
The Boston Consulting Group/OSDN Hacker Survey

Who are the hackers that create all this great free software? How much time do they spend? Why do they do it? Where are they from? Do they think its sustainable? In order to better understand the nature of this dynamic community and provide lessons to the business world about innovation, The Boston Consulting Group, in cooperation with OSDN, surveyed hackers participating on software projects on SourceForge.net, the world's largest collaborative software development Web site for the Open Source community and the Linux kernel mailing list.

Fri 11:00 – 11:45 **Greg KH** (Murdoch)
Putting a filesystem into a device driver

For a while there was a freeze on assigning new major and minor numbers in the kernel, so developers had to use a different way to have a device driver interact with userspace. One of the ways this can be done is by embedding a filesystem into the device driver. This paper will show how this can be easily done for both the 2.4 and 2.5 kernel trees.

It will cover what the basic requirements for a filesystem are, how to create the internal kernel structures and register them properly, and how to create a filesystem that is contained within a removable module. It will also highlight the differences between the 2.4 and 2.5 kernel versions of the VFS layer, and how it can help do the main work for the driver. For drivers that only need to be in 2.5 and beyond, the driverfs filesystem will be discussed as an alternative to creating a separate filesystem. The new libfs code in the 2.5 tree will also be covered, as it can handle much of the grunt work in creating a ram based filesystem.

Fri 11:00 – 11:45 **Brad Hards** (Ross)
Zeroconf networking

Zeroconf networking is an emerging field of work from the Internet Engineering Task Force, providing IP level networking without needing a network administrator. It provides methods to assign an IP address to a host, discover network services by name or characteristics, assign



**"Excusing bad programming is a shooting offence, no matter
what the circumstances." – Linus Torvalds**

multicast addresses and to translate between names and addresses.

The starting point for Zeroconf networking is to take two hosts with ethernet ports, and a cross-over cable, and have them communicate without needing to manually configure network addresses (or have one of them run an DHCP server). This is currently possible with legacy protocols such as Appletalk or NETBIOS, but Zeroconf networking provides a standardised, interoperable way to make easy networking available for any host. This is being adopted by Apple for OS X, under the marketing name of Rendezvous.

While the original scope is simple connections using ethernet cross-over cables or USB host-to-host devices, Zeroconf can be used for new applications that are simply not viable without such technology. For example, it becomes possible to move ethernet connected printers into the mass-consumer market, where the issues associated with configuring an IP address would normally have incurred excessive support costs for retailers.

This talk will mainly focus on the `zcp` tool, which uses standard Linux interfaces to establish IP addresses for a group of hosts that are on the same network (“link local”, traffic with Zeroconf source or destination addresses is never routed), which is available on kernel.org (and mirrors). This tool is (mostly) interoperable with current Windows and MacOS implementations, and aims at being a reference implementation for the Zeroconf IP assignment.

In addition to `zcp`, options for “service discovery” will be overviewed. These include using Service Location Protocol (SLPv2, currently supported by the OpenSLP tools) and use of existing DNS functions and record types (as supported by the BIND server, and other DNS daemons).

The talk will also cover DNS extensions to use multicast to identify the IP address for a particular host. This is a relatively simple idea, with two implementation options currently under consideration. There will also be a brief discussion of multicast group address assignment.

Fri 12:00 – 12:45 H. Peter Anvin (Alexander)
initramfs, klibc and early userspace

Currently the Linux kernel contains a large amount of once-off routines to handle things that would normally be done in userspace. These are in the kernel only because they are run before root is mounted. In the Linux 2.4 kernel series, the kernel core was simplified by creating a synthetic root filesystem at a very early stage, which is later overmounted by the user-visible root. In the Linux 2.5 kernel series, work is currently underway to remove these once-off routines by populating the synthetic root filesystem with small user-space programs. This will result in a more familiar programming model for most developers, less risk of bugs, and more flexibility.

“...very few phenomena can pull someone out of Deep Hack Mode, with two noted exceptions: being struck by lightning, or worse, your *computer* being struck by lightning.” – Matt Welsh



Fri 12:00 – 12:45 **Rasmus Lerdorf** (Murdoch)

PHP Printing with PDF

Ever tried to print an HTML page? Did it ever look decent? Generating a PDF document from a PHP script is easier than you think. Take two parts PHP, 1 part pdflib, a cup of TTF fonts and you have a tasty dish any printer would love.

Fri 12:00 – 12:45 **Jeremy Malcolm** (Ross)

Problems in Open Source Licensing

Despite their ubiquity, a legal cloud hangs over most OSD compliant software licences. There has never been a conclusive legal determination of the validity of the licences that we take for granted. Some of the problems with licensing free software include the fact that there are often hundreds of separate copyright owners, that licences are of an uncertain duration, and as to whether the terms of the licence can be unilaterally changed to make them more restrictive. To complicate matters further, different issues arise in different national jurisdictions, and there are distinct issues raised by each of the common licences such as the GPL, LGPL, BSD, Mozilla and Artistic licences. This presentation will examine these questions, and review some pointers from Australian legal authority on shareware licences as to the way that open source licences might be enforced.

Fri 14:00 – 14:45 **Andrew Tridgell** (Alexander)

Towards full NTFS semantics in Samba

Over the last few years Samba has become a popular platform for building scalable Network Attached Storage systems that support the CIFS protocol used in Microsoft Windows networks. While Samba has enjoyed quite a lot of success as a NAS platform, the original design considerations for Samba were aimed at exposing the Unix/Posix security and filesystem model to the CIFS protocol. More recently the emphasis for NAS vendors has been implementing the fine detailed NTFS semantics in the CIFS protocol and this has stretched the design of Samba beyond what was originally intended.

The basic problem is one of ‘semantic mapping’, particularly with filesystem semantics. In this talk I will discuss the various semantic mapping problems of implementing CIFS on Posix systems and will propose a set of modifications to both Samba and the underlying operating system to allow future versions to provide much more accurate NTFS semantics.

Fri 14:00 – 14:45 **Alexander Reeder** (Murdoch)

Biometrics and Linux

Whether it be a corporation, airport, or government agency, biometrics are becoming an extremely popular way of identifying and classifying people. There are many devices which employ biometric methods, such as fingerprint scanners, voice recognition, and retinal scanners. Many companies are producing their own hardware solutions but few are providing Linux drivers. First a brief review of available hardware will be made followed by Linux support of said hardware (which is very rare).



“Unix gives you enough rope to shoot yourself in the foot.” – H. Peter Anvin

From what was available on the market I chose a fingerprint scanner and commenced work on a Linux driver. Getting documentation to allow me to write such a driver from the parent company is a story within itself, and one I will offer advice on to future adventurers. The next challenge was to write a fingerprint recognition algorithm and comparison method. While these are works in progress they are one of the few GPL'ed fingerprint recognition/verification systems available. The basics of these algorithms and the difficulties behind them will be discussed.

Finally, how can this all be made useful? The Linux model is based on a username/password system. There are fundamental changes that need to be made to the OS for authentication based on biometrics to function seamlessly. Because the authentication daemon will involve an external device there are hooks to the kernel required which can be provided by a device driver. The necessary changes to the kernel and other programs to make a biometric based authentication system feasible on Linux, with the fingerprint solution serving as an example, will be presented.

Fri 14:00 – 14:45 **Thomas Dreibholz, Michael Tuexen** (Ross)
High Availability using Reliable Server Pooling

Providing fault tolerancy is crucial for a growing number of IP-based applications. There exist a lot of proprietary solutions for this problem, but free alternatives are rare.

Currently the IETF RSerPool Working Group⁸ is standardizing a protocol suite for Reliable Server Pooling, which copes with the challenge of providing high availability by providing redundant servers. Servers for the same service are grouped into a server pool, which is identified by a pool handle (PH). A server in a pool is called pool element (PE), a user of a pool is called pool user (PU). When a PE fails, its PUs simply select another one from the pool and initiate an application-specific failover procedure. This failover is supported by the RSerPool protocol suite. Each PE registers at a name server and is then continuously supervised by that specific name server. All name servers of an operational scope provide a redundant system for name resolution from pool handles to transport addresses of pool elements to pool users. RSerPool uses SCTP (RFC 2960) to provide network fault tolerance and address scoping functionality.

The rsplib is a prototype implementation of the RSerPool protocol suite developed under the GNU Public License in cooperation between Siemens⁹ and the Computer Networking Technology Group¹⁰ of the University of Essen. It runs under Linux, FreeBSD and Darwin and uses our SCTP implementation¹¹ or the Linux and FreeBSD kernel implementations. It is available from <http://www.sctp.de/rserpool.html>. This work is partially funded by BMBF¹².

⁸<http://www.ietf.org/html.charters/rserpool-charter.html>

⁹<http://www.siemens.com/>

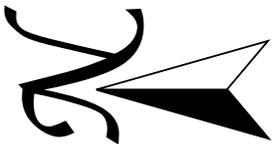
¹⁰<http://tdrwww.exp-math.uni-essen.de/>

¹¹<http://www.sctp.de/sctp.html>

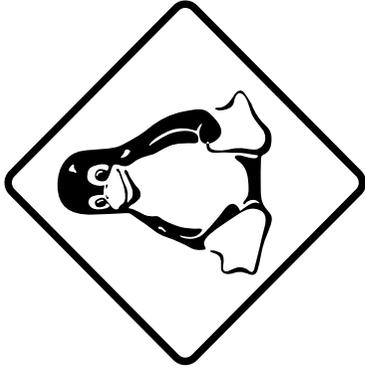
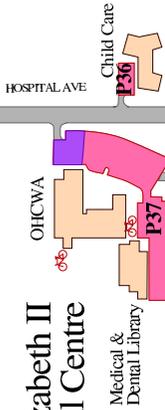
¹²<http://www.bmbf.de/>

“PHP, when it first came out, didn't really have any merits, and many claim it still doesn't, but it filled a void where a simple tool to perform a simple task was needed.” – Rasmus Lerdorf



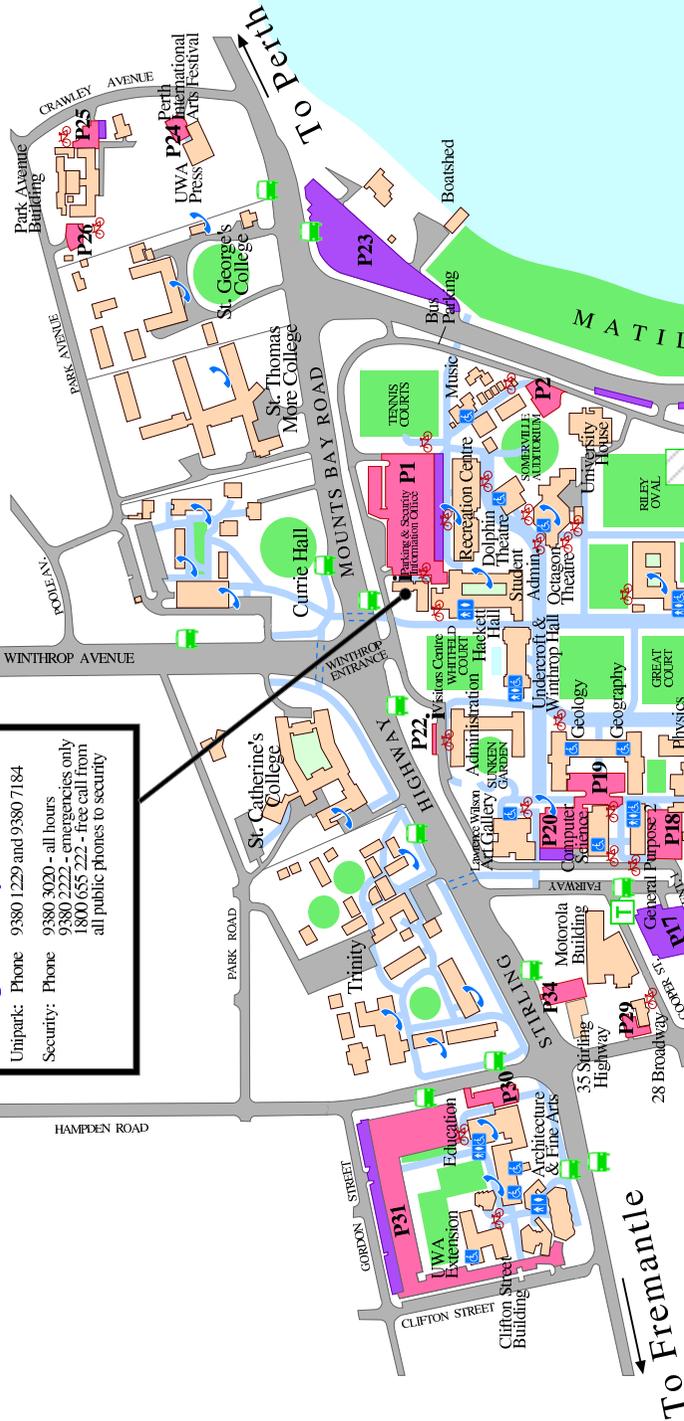


Queen Elizabeth II Medical Centre



Parking & Security Information

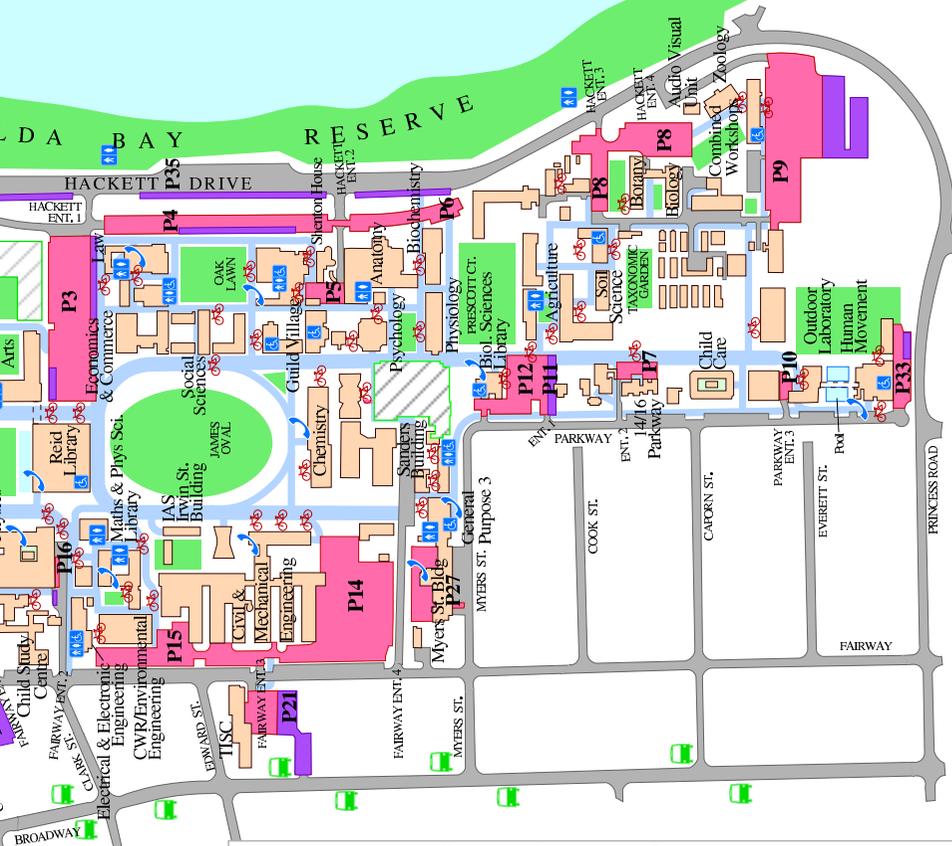
Unipark: Phone 9380 1229 and 9380 7184
Security: Phone 9380 3020 - all hours
9380 2222 - emergencies only
1800 655 222 - free call from all public phones to security



The University of Western Australia Crawley Campus

Matilda Bay

PELICAN POINT



CAR PARKS AND FACILITIES

Please read the signs before you park your vehicle!

Permit Parking

- Yellow (student) 1, 4, 8, 9, 12, 14, 31, 37
- Red (staff) 1, 3, 6-15, 18-21, 24-7, 29-31, 33-4, 36-7
- Car pooling 14 (car pooling permit required)
- University House 2 (members and guests - permit required)
- Disabled 1, 2, 3, 5, 8, 9, 11, 14-16, 18, 19, 21-2, 30-1, 34, 37 (UWA disabled or ACROD permit required)
- Reserved bays reserved at all times - including car pooling
- Service bays 1-3, 5, 6, 8, 9, 11, 12, 15, 16, 18-20, 22, 27, 30-1, 33-4, 37 (place business card or Parking Authority Permit on dashboard)

Pay Parking

- Short Term: \$1.10 per hour - 1, 3, 4, 11, 16, 20, 25, 37
- Long Term: 60 cents per hour - 9, 17, 21, 23, 31, 33, 35
- Motorcycle bays 1, 3, 4, 8, 11, 14, 15, 16, 18, 25, 31, 37 (no permit required)

Limited Bus Parking east of Music building - phone 9380 3020

Bus Routes to Campus

University service No 78, 23-4, 71-3, 9899, 103/4, 97 Shuttle bus between Subiaco Station & UWA.
Please check with Transperth for current information - 13 62 13 - <http://www.transperth.wa.gov.au>

Disabled Toilet.....	
Public Toilet.....	
Public Telephone.....	
Bicycle Rack.....	
Bus Stop.....	
Shuttle Bus Terminus.....	

Our paper covers aspects of designing and implementing highly available applications using RSerPool with our rsplib implementation. First we will give an introduction of the RSerPool protocol suite and an overview of the rsplib components. Then we describe the rsplib API especially focussing on the implementation of pool element and pool user programs to provide high reliability. All API descriptions are accompanied by code examples for better understanding. We present applications for the RSerPool framework, showing the problems of today's Internet applications and how to improve them by using RSerPool and our rsplib implementation. Therefore we cover not only new applications but also look at existing software and describe methods for improving them by using RSerPool. Finally we show how distributed computing architectures can make use of the RSerPool architecture.

Fri 15:00 – 15:45 **Alan Cox** (Alexander)
Useful kernel audits

Want to know more? Then come along!

Fri 15:00 – 15:45 **Simon “Horms” Horman** (Murdoch)
Perdition: A Mail Retrieval Proxy

Perdition is a fully featured POP3 and IMAP4 proxy server. It is able to handle both SSL and non-SSL connections and redirect users to a real server based on a database lookup. Perdition supports modular based database access. ODBC, MySQL, PostgreSQL, GDBM, POSIX Regular Expression and NIS modules ship with the distribution. The API for modules is open, allowing arbitrary modules to be written to allow access to any data store.

Perdition can be used to: Create large mail systems where a users mailbox may be stored on one of several hosts. Integrate different mail systems together. Migrate between different email infrastructure. And in firewall applications.

This presentation will examine how Perdition has evolved over the past couple of years and how it can be used to solve a variety of real world problems.

Fri 15:00 – 15:45 **Neil Brown** (Ross)
Who wants another filesystem

Q. With ext3, reiserfs, reiser4, xfs, jfs (just to name a few), who in their right mind would write another general-purpose filesystem?

A. Wrong question. They are all special purpose.

The goal of lafs is to provide optimal service as a departmental file server. Specific sub-goals include being friendly to:

- NFS - low latency writes
- backups - snapshots for on-line incremental backups
- quotas - tree based quotas



“I love the whole sound of ‘carrier grade’ though: Do you use ‘carrier grade’ Linux or just the ‘recreational boating’ version?”
– Victor Yodaiken

- admins - easy resize/migration, crash recovery
- RAID - stripe-wide writes. always.
- users - fast, reliable, stable, efficient, doesn't crash, doesn't corrupt data, stable. Did I say reliable?

lafs (pronounced "laughs") is a log structured filesystem that is designed to meet all these goals.

Sat 9:30 – 11:30 **Revolution OS** (Octagon Theatre)

Movie

Sat 12:00 – 12:45 **Martin Pool** (Alexander)

distcc

distcc¹³ is a program to speed up compilation of C or C++ code by distributing work across several machines on a network. Builds using distcc can be three or more times faster than compilation on a single machine.

distcc is designed to painlessly integrate into existing networks and software build systems. It does not require kernel patches, a shared filesystem, or even for all the machines to have the same libraries installed. distcc can even be used to distribute work across machines with different operating systems or compilers.

The paper will describe the problem space and distcc's solution, provide a brief example of use, and provide performance results.

Sat 12:00 – 12:45 **Jeff Waugh** (Murdoch)

Sequel Syndrome: The GNOME Project Grows Up

The GNOME 2.0 Desktop & Developer Platform is now widely regarded as GNOME's best release to date. Two years in the making, it represents a massive investment in our core technology, and the maturation of the GNOME Project as a whole.

This talk will cover major software enhancements such as accessibility, the emphasis on usability and minimalism, and our new developer platform; plus broader issues such as project management, corporate involvement, and the changing face of The GNOME Project.

Jeff Waugh is the head beekeeper of the GNOME Release Team, President of the Sydney Linux User's Group, and was on the organising committee for LCA2001. In his spare time, he is an IT consultant specialising in Free Software (on servers *and* desktops). He is especially pleased to see the keen-bean Perthies take on linux.conf.au, given their drunken enthusiasm for hosting it way back in 2001.

¹³<http://distcc.samba.org/>

"Only wimps use tape backup: *real* men just upload their important stuff on ftp, and let the rest of the world mirror it ;)"
– Linus Torvalds



Sat 12:00 – 12:45 **Geoffrey D. Bennett** (Ross)

Katie, open-source revision control system

Katie is an open-source revision control system, intended to be a look-alike of the proprietary ClearCase Software Configuration Management system. The primary difference between Katie and other open-source revision control systems is that rather than copying files from a repository to a local working directory, the Katie server exports the repository as an NFS filesystem. When compared with CVS, this allows for a number of novel features:

- Any number of workspaces may be created, with no disk space penalty, as the repository contents are logically (not physically) duplicated in each workspace.
- The time taken for the equivalents of “cvs checkout module” and “cvs update -r tag” is independent of the size of the repository, again because file data does not need to be copied to the workspace.
- The renaming or deleting of files is handled in a clean manner by supporting versioning of directories. In addition, directories may be tagged or branched just like any regular file.
- Every revision of every file ever created (including deleted files) is available to standard Unix commands through the filesystem in a hidden “extended namespace”; this for example means that no equivalent to the “cvs diff” command is required.
- The branches and versions of files and directories which a developer sees in their workspace is determined by a small text file known as a configuration specification. After editing this file, the newly-selected versions are immediately visible in the workspace.

Both the Katie client and server are written almost entirely in Perl. As there is no Perl SunRPC implementation, Inline.pm is used to link in C code that is necessary for an NFS server. PostgreSQL is used to store repository metadata, and the actual repository data is handled through a replaceable storage manager.

Sat 14:00 – 14:45 **Matthew Wilcox** (Alexander)

I'll Do It Later: Softirqs, Tasklets, Bottom Halves, Task Queues, Work Queues and Timers

An interrupt is a signal to a device driver that there is work to be done. However, if the driver does too much work in the interrupt handler, system responsiveness will be degraded. The standard way to avoid this problem (until Linux 2.3.42) was to use a bottom half or a task queue to schedule some work to do later. These handlers are run with interrupts enabled and lengthy processing has less impact on system response.

The work done for softnet introduced two new facilities for deferring work until later: softirqs and tasklets. They were introduced in order to achieve better SMP scalability. The existing bottom halves were reimplemented as a special form of tasklet which preserved their semantics. In Linux 2.5.40, these bottom halves were removed; and in 2.5.41, task queues were replaced with a new abstraction: work queues.



init: Received SIGNUKE, killing all users. – H. Peter Anvin

This paper discusses the differences and relationships between softirqs, tasklets, work queues and timers. The rules for using them are laid out, along with some guidelines for choosing when to use which feature.

Converting a driver from using the older mechanisms to the new ones requires SMP auditing and understanding the interactions between the various driver entry points. Accordingly, there is a brief review of the basic locking primitives, followed by a more detailed examination of the additional locking primitives which were introduced with the softirqs and tasklets.

Sat 14:00 – 14:45 **Malcolm Tredinnick** (Murdoch)
Linux at the Little End of the World

Small Linux installations are becoming more prevalent. Embedded devices supporting Linux are popping up all over the place. Every time a new PDA is released, the race is on to get Linux installed on it -- unless the manufacturer has already done that. We are also seeing things like small firewall installations, dedicated application servers and Internet kiosk styles of setups, where a complete reinstall is often necessary and should be uncomplicated and fast.

This talk will survey the current landscape of available software for making tiny Linux installations and the difficulty level of actually putting the pieces together (a ten year old could do it). The focus is on installing minimal amounts of software to achieve real functionality, but using readily available packages (so there will be almost no poking about inside the kernel or making unauthorised modifications outside of 'make menuconfig').

Some of the hardware considerations involved will be mentioned, but mostly we will focus on the software side of the equation.

People who have wondered about

- setting up a robust Internet kiosk or information booth,
- building a Linux-based digital video recorder,
- how “they” fit Linux onto a telephone or iPAQ,
- just saving more disk space and RAM than you could ever imagine will hopefully gather some clues from this talk.

Some familiarity with using Linux and the functions it provides and a general awareness of some of the “gee whiz” gadgets in the news today are all that will be assumed in this introductory level talk.

Sat 14:00 – 14:45 **Rhys Weatherley** (Ross)
Design of the Portable.NET Interpreter

The Common Language Infrastructure (CLI) is a set of specifications that describe a bytecode-based development and runtime environment. Portable.NET¹⁴ is an implementa-

¹⁴http://www.southern-storm.com.au/portable_net.html

“Eagles may soar, but weasels don’t get sucked into jet engines.” – from the .sig of Malcolm Tredinnick



tion of the CLI, whose primary design goal is portability to as many platforms as possible.

Portable.NET achieves portability primarily through its use of interpretation rather than Just-In-Time compilation. However, the bytecode format of the CLI presents some challenges to efficient interpreter implementation.

The approach we take is very similar to a JIT: the bytecode is translated into instructions for a simpler abstract machine, dubbed CVM (for “Converted Virtual Machine”). This machine is then interpreted using a high-performance engine.

Traditionally, abstract machines have used the same bytecode representation “on the wire” as for execution (e.g. JVM, Parrot). Our work shows that there are definite performance advantages to using different bytecode representations internally and externally.

This isn’t necessarily a new result: JIT’s have been demonstrating this fact for some time. Bytecode translation has also been common in the Forth interpreter community. But it isn’t as well known in the wider developer community.

This paper will discuss the Portable.NET engine design, and its implications for designing Open Source and Free Software VM’s in the future.

Sat 15:00 – 15:45 **Patrick Mochel** (Alexander)

The Linux Kernel Device Model

The Linux kernel is getting a makeover during the 2.5 development series concerning the way it represents devices and drivers. The device model is a refinement of kernel data structures and a unification of the mechanisms to operate on these structures.

Through this unification, an incredible amount of code cleanup is possible. Duplicate data structures and interfaces can be consolidated into common code in the device model core. Global interfaces for devices and types of devices can be created to do once near-impossible things, like system-wide power management.

In fact, many complex problems are greatly reduced in size as a direct result of the device model. Besides power management, operations like persistent device attribute restoration and userspace-driven device naming policy can be implemented cleanly and efficiently.

Further, by creating common data structures throughout driver subsystems, these objects and the linkage between them is easily represented in userspace via driverfs. Attributes of devices and drivers can be easily exported in driverfs. This interface is designed to enforce a simple, one-value-per-file ASCII interface, and eliminate the need to implement buggy formatting and copying code in drivers.

The goal of this paper is to describe the attributes of the device model, including:

- The data structures describing the objects
- The interfaces exposing them to the rest of the kernel
- Driverfs and userspace representation of object attributes
- Applications of the model, and the resolution of complex problems



```
s&&!msfQ!&&s&$&tutvK&&s&(Q)&\1!sfiupoB&&s&~&reverse Ibdlfs&e&s&~&#&&s&$&&#!uoj&s&(&.)&chr(ord($1)-1)&g&e&s&(.*)&reverse $1&see
```

- How it will achieve world peace

The presentation will discuss the concepts in the paper, as well as temporal topics, including the current progress of the implementation, what is next on the agenda, and what is needed from developers.

Sat 15:00 – 15:45 **James Henstridge** (Murdoch)

EggMenu

In the GNOME 2.0 development platform, there are currently at least four different ways to build up the menus and toolbars, each of them depending on a different set of platform libraries. This is a pain for application developers, as they may find themselves needing to rewrite parts of their UI if they choose to use parts of the GNOME platform such as Bonobo.

In order to solve this issue, a new menu API that contains the best features of the existing APIs, but at the same time can be included in GTK+ so that all applications can use it. The new API is currently in prototype form, and is scheduled for inclusion in GTK+ 2.4.

Some benefits of the new menu system include:

- actions that a user can make are represented as objects that can have any number of menu items and toolbars associated with them.
- for “toggle” style actions, the appearance of all tool and menu items are kept in sync.
- Layout of menus and toolbars can be stored in an external XML file, and multiple UI descriptions can be merged together (useful for compound documents).

The new API is designed to scale from large productivity applications down to small programs, while still being easy to use. It is also easy to extend in order to add new types of actions (which will probably be required for bonobo), and easy to wrap for language bindings.

The talk will also go over the new RFP procedure that is being adopted in GNOME to handle introduction of features such as the new menu API.

Sat 15:00 – 15:45 **Adrian Chadd** (Ross)

Developing network applications under UNIX

I will cover the various aspects of developing network and long-lived applications under UNIX. This includes comparing languages (C, C++, Java), paradigms (single thread/callback, pthreads, processes/shared memory), APIs (Unix vs Windows network IO, Unix vs Windows file IO, memory allocation/ management) and system-level performance (poll() vs kqueue(), zero-copy read()/write(), network driver performance, general kernel performance.)

“panic(“esp: Aiee penguin on the SCSI-bus.”)”— The Kernel



Transport

You are probably going to want to get around, unless you don't intend to see the beautiful city, beaches, bell tower or people that Perth has to offer and instead stay in your room. Assuming that you want to go places, you may find this information useful.

Places to know

- Perth Train Station (Platform 1): 31° 57' 3.2" S, 115° 51' 31.7" E.
- Perth Domestic Airport: 31° 55' 58.8" S, 115° 57' 37.2" E.
- The University of Western Australia¹⁵: 31° 58' 52.1" S, 115° 49' 11.7" E.

Tickets purchased on any Transperth bus, train or ferry service, costing more than \$1.50 may be used to 'transfer'. With a ticket you can board any other bus, train or ferry as many times as you like until it expires (usually 2 hours from the time of purchase). This is often useful for return trips or journeys where multiple buses and/or trains are required.

Buses

Bus fares are charged in zones and by time. The number of zones you travel depends on how many zones away your final destination is. Telling the bus driver where your final destination is (no matter how many changes you have to do) will mean your ticket will be valid for the entire journey (within the 2 hour limit). Bus tickets are also valid on trains and ferries. You do not need a ticket to travel on CAT (intracity) buses.

There are several buses available to and from The University of Western Australia. These include the No 23, 24, 71, 72, 97 (Subiaco Shuttle) and the 98/99 (Circle Route). For more information visit <http://www.transperth.wa.gov.au/> or call Transperth on 13 62 13.

- The 23 goes from the City Busport to the top of the University (Stirling Highway), then to Claremont, then back again. To get to the bus stops on the far side of Stirling Highway (the ones that go to the Perth CBD) use the pedestrian underpasses. To find one of these simply head to the top of the uni from either the back of Physics (Fairway) or from the front of Arts (near the Reid Library).
- The 24 goes from Broadway (the next road along from Fairway on the Physics side of the University) to St Georges Terrace in the Perth CBD. It passes through Sir Charles Gairdner hospital.
- The 71 and 72 go from the top of the University to the Busport and then up to St Georges Terrace (Perth CBD).
- The 97 goes from Fairway (near Computer Science, which is just north of Physics) to Subiaco train station.
- The 98/99 is a "Circle Route". The number dictates in which direction the bus is running - the 98 travels clockwise around Perth, whilst the 99 travels anticlockwise. The Circle Route runs in a circle from Fremantle past the University and around the city as far out as Morley and Victoria Park. If you catch this bus ensure you are

¹⁵Co-ordinates of the University Computer Club - <http://www.ucc.asn.au/>



"On a normal ascii line, the only safe condition to detect is a 'BREAK' - everything else having been assigned functions by Gnu EMACS." – Tarl Neustaedter

going in the right direction because although it travels in a circle it takes 3.25 hours to complete a circuit.

Taxis

There are several taxi companies in the city. While they speak a most interesting version of English, they can usually find major landmarks such as the University, the Airport and Perth CBD.

- Black & White Taxis
Ph: 13 1008
- Swan Taxis
Ph: 13 1330
- Yellow Cabs
Ph: 13 1924
- Perth Maxi Taxis
Ph: (08) 9409 4222

For those that don't know, maxi-taxis are large van type vehicles that can seat up to 7 passengers (some maxi-taxis seat more) or copious amounts of luggage. While they do cost a little more than a conventional taxi, if you have lots of luggage or people to move then you should consider getting one of these instead.

Trains

There are four train lines, all which originate from Perth train station. The lines terminate in Midland, Fremantle, Armadale and Currumbine. Trains travelling between Perth, McIver, Claisebrook and City West are free. The cost of all other trains depend on the number of zones you are travelling to reach your destination. Train tickets also work on buses. Note that no train alone will take you to the University.

Parking at The University

During the conference, **only** *Student* parking (yellow bays) are free and unrestricted. Refer to the legend on the map in the centre of this booklet and the list below for locations. There is pay parking around the campus, with 9 hour parking bays along Hackett Drive. All other parking is restricted and enforced – this includes the red Staff bays and blue Reserved bays. Disabled parking is available in most car parks.

The closest free car parks for the conference are:

- **P1** – North end of campus, outside the Recreation Centre. Enter through the Winthrop Avenue entrance, turn left and then right.
- **P4** – East side of campus, closest to Arts. Enter through Hackett Entrance 1 and turn left.
- **P14** – South-west end of campus, adjacent to Civil Engineering. It is further than the other car parks but has many more bays. Entry is through the Fairway 3 entrance, followed by an immediate right turn.

“I'd crawl over an acre of 'Visual This++' and 'Integrated Development That' to get to gcc, Emacs, and gdb. Thank you.”
– Vance Petree



Eating

This is actually a very important thing to do, and on the nights where we don't feed you, or you get extra hungry at 02:00 am¹⁶ whilst coding and/or kernel hacking you will be glad that we wrote this down for you.

At the University

- **Reid Cafe**

Located on the ground floor of the Reid Library, underneath the overpass you have to use to access the library building, the cafe offers a range of cafe-type products including drinks and light meals. To find the Cafe, enter the library through the automatic doors, turn right through the security sensors, then go down the stairs to the left of the enquiries counter. Turning left at the bottom of the stairs should reveal its location.

If you can't find it, ask someone - everyone who's a regular on campus will be able to tell you where it is with pointing and arm movements, and generally in better detail than is described above.

- **Guild Village**

In the Guild Village you will find the Village Cafe, ATMs, a pharmacy, and more vending machines.

- **Hackett Cafe**

The Hackett Cafe is located on the north end of campus in Hackett Hall.

Near the University

- **Nagano**

88 Broadway, Nedlands

A small store selling Japanese food. It is located in the Broadway Fair shopping complex (accessible from Fairway near Cook Street, just past Chemistry). They sell a number of things including sushi.

- **KK's on Broadway**

Broadway Shopping Centre, Broadway, Nedlands

Also in Broadway Fair, KK's on Broadway (formerly The Last Drop Tavern) is a microbrewery specialising in German style beers. They do a range of food items too. If you flash them your LCA badge they'll give you a 10% discount on most things.

- **Mothers Foods**

11 Broadway, Nedlands

This is the only place you can go when you're hungover. They sell a whole load of nice homecooked style food. They do a nice chicken and avocado foccacia. They are located at the top of Broadway, right next to the Caltex on Stirling Highway/Broadway.

¹⁶The Moon Cafe is open late most nights. Chelsea Pizza will take orders till 1am on Thursday nights, 2am on Friday & Saturday, midnight all other nights.



- **Happy Buddah**

131 Stirling Hwy, Nedlands

This is a little further away, but happens to be a good source of vegetarian and vegan food, it'll make the vegetarians you are partying with happy. The phone number is (08) 6389 1333.

- **Broadway Seafood**

130 Broadway, Nedlands

Located next to Domino's, this is a good source of fish and chips. The phone number is (08) 9386 2341.

- **Pavarotti's**

25 Hampden Road, Nedlands.

Traditional home-cooked Italian cuisine and a selection of home-made wood-fired pizza. They also have an extensive range of pasta, seafood, meat dishes, and tempting salads. Open 7 days for lunch. Licensed and BYO. (08)93865550.

- **Tiamo's**

57 Hampden Road, Nedlands

Italian cuisine, reasonable prices, large serves, fully licensed, BYO and they do breakfast, lunch and dinner! They're usually the last restaurant to close on the cafe strip. Call ahead, because tables are usually hard to find. (08) 9386 6611.

- **Greco's on Broadway**

161 Broadway, Nedlands

Greco's is everything you want in a cafe - friendly staff, alfresco, indoor and outdoor eating, and a great range of cakes. They do fantastic specials on Monday, Tuesday and Wednesday. Their number is (08) 9389 8410.

- **Chelsea Pizza**¹⁷

145 Stirling Highway, Nedlands

Gourmet pizzas and cuisine. (08) 9386 7833.

- **Broadway Pizza**

165 Broadway, Nedlands

As the second closest pizza store (after Domino's), Broadway Pizza is considered by some to make 'classier' pizzas. Pay more, but get a lot more! The number is (08) 9389 8500.

- **Domino's**¹⁸

128 Broadway, Nedlands

This is about the closest pizza store to the University and is about 5 minutes walk from Physics. Domino's serve the same standard schmooze to everyone everywhere, so it should be fairly evident what you're ordering. The Domino's Nedlands phone number is (08) 6389 0233.

¹⁷<http://www.chelseapizza.com.au/>

¹⁸<http://www.dominos.com.au/>

“The most important design issue... is the fact that Linux is supposed to be fun...” – Linus Torvalds



- **Pizza Hut**¹⁹
159 Onslow Road, Shenton Park
If you don't like Domino's and still want pizza, then try Pizza Hut. Pizza Hut is a little further away in Onslow Street in Shenton Park, however they do deliver at a cost. The Pizza Hut number is (08) 9381 9333. Pizza Hut also accept the discount vouchers of all the other major pizza-making chains so that could save you some dosh. Tell them Quinn recommended you...
- **Hampden Road/Broadway** have a lot more restaurants and cafes (in fact, too many to list them all here!)

In the City

A good place to find food in the city is in Northbridge. Northbridge contains numerous restaurants, cafes, bars, pubs, nightclubs and just about everything else you could want on an entertainment strip. The easiest way to get to Northbridge is to catch a bus to the Wellington St bus station (Perth train station) and walk across the overpass and down the escalator. Continuing to head in the same direction will reveal a number of exciting places to eat.

- **Han's Cafe**
245 William Street, Northbridge
Have a thing for Asian food at the moment? Han's Cafe do some really nice rice and noodle dishes and they will give you chopsticks if you ask.
- **Utopia**
Shop 14/109 James Street, Northbridge
Vegetarian/Vegan food. The phone number is (08) 9227 8588.
- **Brass Monkey**²⁰
209 William Street cnr James Street, Perth
Good food. Good beer. Great surroundings. Enough said? (08) 9227 9596
- **Moon and Sixpence**
300 Murray Street, Perth
English style pub. This is the regular meeting venue for the SAGE-AU WA Chapter. Phone number is (08) 9481 0727
- **Moon Cafe**
323 William Street, Northbridge
The Moon Cafe is a young 'hip' place to dine out. It's a late night cafe which appeals to students and locals. They're fully licensed, BYO and have a forty foot bar... (08) 9328 7474

¹⁹<http://www.pizzahut.com.au/>

²⁰<http://www.thebrassmonkey.com.au/>



Other Places (Subiaco and Fremantle)

Of course, there are other places to eat besides Crawley and the city. The University is in close proximity to both Subiaco and Fremantle. Both of which have excellent markets and cafe strips.

- **Cicerello's**

Fishermans Wharf 44 Mews Road, Fremantle

Located on the Fremantle foreshore, this is one of the most famous fish and chip restaurants in Perth. It is fairly easy to find, there is a great big sign with the name on it.

- **Sail & Anchor**²¹

64 South Terrace, Fremantle

Pub and brewery rolled into one. Sail & Anchor is a Fremantle landmark, famous for its alfresco Brewer's Courtyard. (08) 9335 8433

- **Norfolk Hotel**

47 South Terrace, Fremantle

With 14 types of beer on tap, a cellar full of WA wines, a constantly buzzing atmosphere, and centered in the heart of Fremantle, this hotel makes an enjoyable stop on a night out. (08) 9335 5405

- **Little Creatures Brewing**

40 Mews Road, Fremantle

Located on the Fremantle Fishing Boat Harbour, this superb spot marries a brewery with a delightful bar and eatery. (08) 9430 5155

- **Pizza Bellaroma**

14 South Terrace, Fremantle

Traditional woodfired oven pizzas. The tables are never empty - you may need to sell your soul to get a seat. (08) 9335 1554

- **Joe's Fish Shack**²²

Mews Road, Fremantle

Located right on the Fishing Boat Harbour, the fish doesn't get any fresher than this. Joe's offers spectacular views out to the ocean. (08) 9336 7161

²¹<http://www.sailandanchor.com.au/>

²²<http://www.joesfishshack.com.au/>

"Emacs is for people who desperately want to get drunk, but feel guilty doing so without a reason." – Miles O'Neal



Perth & Entertainment

Perth is located at 31 degrees south, 115 degrees east, on the west coast of Australia. It is built around the water system known as the Swan River. Perth is best known for it's miles and miles of beautiful beaches and friendly people. Perth has heaps to see including Fremantle, Subiaco, Northbridge and Penguin Island. Here are just a few we thought important:

- **Somerville Theatre**²³ - This is an outdoor theatre located on campus, defined by a “cathedral of Norfolk pines”, just north-east of the Octagon Theatre. During the summer, the Festival of Perth show films on the big screen.
- **Sunset Cinema**²⁴ is also an outdoor cinema, located in Kings Park. Within walking distance of Currie Hall and the other colleges, it forms part of the Western Power Parkland. The gates open at 6:00pm, with screenings beginning at 8:00pm. For more information on current movies and getting there visit the website.
- **Northbridge** is considered by most to be the centre of nightlife in Perth. With so many nightclubs and pubs, the only way to experience it all is start exploring!
- **Fremantle** can be thought of as a satellite city of its own. A 15 minute drive down Stirling Highway brings you to a heart of bubbling life. Fremantle offers many markets, clubs and pubs, amongst restaurants and shopping centres.
- **Subiaco** is a 15 minute bus trip away and offers a range of stores, restaurants and fast food chains. The 97 “Subiaco Shuttle” runs between the University and Subiaco every 15 minutes, departing from Fairway on the west side of Campus.
- **Beaches** are a main attraction of Perth - there are so many to choose from! The 72 bus will take you from the University to Cottesloe beach. Some other beaches worth visiting are Scarborough, Trigg and Swanbourne.

More information on places to see is contained within your conference bag, and also at <http://www.whatson.com.au/>

²³<http://www.theatres.uwa.edu.au/whatson.html>

²⁴<http://www.sunsetevents.com.au/cinema/perth/>



“A passport is not sufficient ID to get a library card here.”
– Telsa Gwynne

Notes







This survival guide was written entirely in L^AT_EX, POSTSCRIPT[®], Perl and a Makefile, by Bernard Blackham, Davyd Madeley, Mark Tearle, James Henstridge and countless others. Front cover artwork by Mark Tearle, a long drive to Geraldton and The GIMP. The core organisers of linux.conf.au 2003 are James Andrewartha, Nick Bannon, Bernard Blackham, James Bromberger, James Henstridge, Anand Kumria, Tony Breeds-Taurima, and Mark Tearle. More people who have helped make linux.conf.au 2003 possible are Leon Brooks, Mark Gaynor, Trent 'Lathiat' Lloyd, Davyd 'proXy' Madeley, Harry McNally, Jacqueline McNally, Konrad Rubin, Adrian Woodley, the neglected parents, partners and pets of the organisers, and many many more people that I apologise to if I've left them out! Hats off to the previous organisers from CALU, LCA2001 and LCA2002 who made us think this was a good idea. Thanks to Quinn for ordering pizza for us at our organisational meetings all year. Linux[™] is a trademark of Linus Torvalds. All other trademarks are the property of their respective owners.

Our Professional Delegates

Simon Uduma Onyebuchi	ABC NIG
Jimoh Tunde S. Smith	Adebolcomputers Nigeria Ltd
Janet Olomadun Adebajo	Adelad Agro-Allied Ltd.
Pastor Adebajo Adeyanju	Adelad Agro-Allied Ltd.
Michael Haarsma	Adelaide Institute of TAFE
Stephen Donaldson	Adelaide Institute of TAFE
Anthony David	Anthony David & Associates Pty Limited
Jeremy Nelson	Apposite IT
Dr Peter Williams	Aurema Pty Limited
Robert Edwards	Australian National University
Jamie Birse	BAE systems
Andrew Buckeridge	BGC (Australia) Pty Ltd
Ben Kelly	BGC (Australia) Pty Ltd
Andrew Carson	BMIS Pty Ltd
Mike Schuman	Broadcast Engineering Services
Andrew Donaldson	Bureau of Meteorology
Russell Steicke	Bureau of Meteorology
James Hamilton	Canberra Institute of Technology
Judy Adams	Canberra Institute of Technology
Tony Breeds	CanTech Online
Christopher Brittain	Centrelink
Peter King Cheung Chan	Centrelink
Niall Young	Chime Communications
David Creelman	Cochlear Ltd
Simon Bowden	Computer Sci & Engineering, UNSW
Slade Matthews	Computer Sci & Engineering, UNSW
Trent Swift	Computer Sci & Engineering, UNSW
Greg Lehmann	CSIRO
Greg Wogan-Browne	Data Analysis Australia
Adam Crow	DCNUT Pty Ltd
Ashley McKenzie	Deakin University
Dominic Sgro	Deakin University
Renton Chappell	Deakin University
Jacqueline McNally	Decisions and Designs Pty Ltd
Steven Hanley	Department of Computer Science, ANU
Kathryn Moyle	Department of Education and Children's Services
Steve Landers	Digital Smarties
Daniel Smith	Dreamcraft Pty Ltd
Simon Fowler	Dreamcraft Pty Ltd
Desmond R Elliott	Dunhess Pty Ltd
Mike O'Connor	E-consortium P/L



Andrew Jackson	EverythingLinux
Anthony Rumble	EverythingLinux
Grant Parnell	EverythingLinux
James Stanley Akinola	F&K Wilson and sons Ent.
Paul Schulz	Foursticks
Mark Schaschke	Functional Software
Olalekan Sanyaolu	G.I.G Ventures Ltd
Ian Wienand	Gelato.org
Marc Merlin	Google Inc
Peter Duff	Google Inc
Chris Jankowski	Hewlett-Packard
Saliya Wimalaratne	Highlands Internet
Indulis Bernsteins	IBM Australia
David Gibson	IBM Linux Technology Centre, OzLabs
Hugh Blemings	IBM Linux Technology Centre, OzLabs
Stephen Rothwell	IBM Linux Technology Centre, OzLabs
James Watson	IDEXX Laboratories
Andrew Howell	Informed Technology Pty Ltd
Brian Danilko	Inspiration Technology
James Morris	Intercode
Jonathan Oxer	Internet Vision Technologies
Timothy Churchward	Internet Vision Technologies
Greg Lehey	LEMIS (SA) Pty Ltd
Anand Kumria	Linux Australia
Matt Kemner	Linux Information Technology
David Buddrige	Logica
Russell Stuart	Lube Mobile
John Usher	Maptek Pty Limited
David Lloyd	Microbits
Troy Vodopivec	Microbits
Mark Suter	Miju Systems
Anibal Monsalve Salazar	Monash University
Kim Oldfield	Monash University
Chris Quinn	Murdoch University
Michael Deegan	Murdoch University
Simon Owell	Murdoch University
Trevor Phillips	Murdoch University
Phil Hutton	NetCraft Australia
Andrew Shugg	Network 4
Peter Howard	Northern Ridge P/L
Mark Bradbury	Northern Territory University
Darryl Bond	NRG Gladstone Operating Services
Cameron Rich	Nulec Industries



Talluru Kalyan Chakravarthi	Numerectech .sdn.bhd.
Olatubosun Afolabi	Olafad Ventures Ltd
Andrew Monteiro	OpeNeT Pty Ltd
Bruce Badger	OpenSkills Pty. Ltd.
Andrew Cowie	Operational Dynamics
Adam Ashley	Optimiser Pty Ltd
Duncan Sargeant	Optus
Brendan O'Dea	Optus Internet
Dale Hutton	Penrhos College
Paul Jankowski	ProLinux
Ian Ralph	SCEGGS Darlinghurst
Darryl Ross	Service to Youth Council Inc
Rhys Weatherley	Southern Storm Software, Pty Ltd
Kim Perkins	Strathcona BGGs
Aaron Smith	Sun Microsystems Australia
Duncan Bennet	Sun Microsystems Australia
Hans Humboldt	Sun Microsystems Australia
Ian Dagger	Sun Microsystems Australia
Michael Bukva	Sun Microsystems Australia
Scott Miller	Sun Microsystems Australia
Tung U	Sun Microsystems Australia
Nathan Alberti	Swiftel Communications
Matthew David Sanderson	TCG Information Systems Pty Ltd
Dan Shearer	Tellurian
Alex Dawson	The University of Western Australia
Grahame Bowland	The University of Western Australia
Jenny Vuong	The University of Western Australia
Toivo Pedaste	The University of Western Australia
Nicholas Tan	The Walter and Eliza Hall Institute of Medical Research
Travers Nicholas	TMP Worldwide
Tim Bell	Trinity College
Andrew Hendry	Tusc
Peter Silver	UltraFast Communications
Mike Beattie	Universtiy of Otago
Byron Hammond	Vircom Pty Ltd
Andreas Koepke	WA Business News
Luke Dudney	WestNet
Anthony Pocock	WestNet Consultants
Andrew Hunter	WNI Oceanographers & Meteorologists
Craig Stephenson	WNI Oceanographers & Meteorologists
Zhenggen Xiao	Zhenggen
Marco Ostini	
Tony Grace	



