



news@UK

The newsletter of FLOSS UK, the new name for the UK's oldest Open Systems User Group, UKUUG

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

Volume 20, Number 2

ISSN 0965-9412

June 2011

Contents

From the Secretariat	3
Chairman's Report	3
UKUUG rebranding as FLOSS UK	4
Spring Conference: March 2011, Leeds	4
OpenTech 2011	5
OpenTech report	7
IWEEE 2011 Free Software for e-health and education in developed countries	8
The £100 PC	10
Government Consultation on Free and Open Source Software	10
Book review: Designing Interfaces	11
Book review: Vi and Vim Editors Pocket Reference	11
Book review: Programming Python	12
Book review: HTML5 and CSS3	14
Book review: Mining the Social Web	15
Book review: REST in Practice	16
Book review: jQuery Pocket Reference	16
Book review: Learning the iOS 4 SDK for JavaScript Programmers	17
Book review: Learn You a Haskell for Great Good!	18
Contributors	19
Contacts	21

From the Secretariat

Jane Morrison

I am pleased to advise that the Spring Conference held 22nd–24th March in Leeds was very well attended and the feedback forms indicate that delegates enjoyed the event. See the write-up elsewhere in this issue.

The 4 days of Perl tutorials in February (Intermediate and Advanced) also attracted a good number of delegates and we thank Dave Cross for working with us again.

Since then we have been busy organising a full schedule of events for the rest of the year.

- PF tutorial – by Peter Hansteen – 22nd September – London
- UKUUG - AGM – 22nd September – London
- Un-conference – 8th October – Manchester
- DNSSEC tutorial – by Jan-Piet Mens – November – London

The next Spring Conference will be held in March 2012, possibly in Edinburgh and the 2013 event may be held in Newcastle.

Also next year, we have already planned the following course:

“Intensive Introduction to Google App Engine with Python” – full day tutorial/hands on practical course by Paul Barry – London – 12th April 2012

The Call for Papers for Spring 2012 will be available shortly.

We are again pleased to be working in collaboration with Josette Garcia at O’Reilly to bring you various tutorials.

Novell have renewed their Gold Sponsoring membership with UKUUG for a further 12 months. The sponsoring memberships do greatly assist us and allow us to keep event fees as low as possible.

The UKUUG Annual General Meeting will be held this year on Thursday 22nd September. Further details, Agendas etc. will be sent to you automatically during August.

Don’t forget the Inland Revenue allows UKUUG members to claim their subscription amounts for tax allowance purposes. See further details on our web site.

The next Newsletter will appear in September and has a copy date of: 19th August. Interesting articles from members will be very welcome – all submissions should be sent to newsletter@ukuug.org.

Chairman’s Report

Paul Waring

Spring Conference

Our annual Spring conference was held in Leeds towards the end of March and was a successful event as always, with a packed programme covering topics across all aspects of system administration, over a hundred delegates, and several opportunities to sample the varied cuisine on offer in the city. Thank you to all the speakers who gave up their time to discuss their latest work with us, and the numerous volunteers who helped the event run smoothly. Where speakers have provided slides we have made these available online on the conference website:

<http://www.ukuug.org/spring2011>

Plans are already underway for the 2012 and 2013 conferences, which we are hoping to hold in Edinburgh and Newcastle respectively. There are lots of ways to help out, from chairing sessions to preparing the timetable, so please let us know if you would be interested in getting involved.

Changes to Council

In a previous newsletter I reported that due to the resignation of John Collins there was now a vacant place on Council. I am pleased to announce that Martin Houston volunteered to step in, and he was co-opted onto Council at our last meeting. Martin will stand for election formally at this year's AGM, when members will be able to vote on his appointment.

UKUUG rebranding as FLOSS UK

Paul Waring

You may have noticed a 'FLOSS UK' branding on the front of this newsletter (mentioned in the previous issue), and in our regular advertisements in Linux Magazine. Input from members on UKUUG's activities and branding during recent conferences, on feedback forms and at the discussion session following the 2009 AGM, suggested that a broadening of scope beyond the "Unix" specifics implied by the UKUUG name. Council discussed these ideas at its quarterly meetings and took the decision to begin branding events as FLOSS UK, starting with the unconference in October last year, where it was well received and brought in a number of people who had never attended a UKUUG event before.

The new name is intended to reflect our broader scope, as well as the co-ordinating role we hope to play with other similar groups, a suggestion which arose from the post-AGM discussion. In particular, we are fortunate to have an office and infrastructure which we can use to handle event administration, allowing other groups to concentrate on obtaining speakers and publicising their events without having to worry about booking venues. We already have plans to run a PostgreSQL stream alongside our Spring conference next year, and hope to undertake similar events in conjunction with other groups.

To reinforce the brand, we have a new website, <http://www.flossuk.org>, which is built on the Moin-Moin free software wiki. This will allow greater interaction from members and conference delegates and speakers, and has worked well when used for previous conferences.

The UKUUG site will remain online at its current address, together with all the content which has built up over the years. Council intends to continue hosting the site as an archive of past conferences, newsletters, book reviews and other activity. Mailing lists and member email addresses will also continue at ukuug.org for the foreseeable future, and the UKUUG brand still exists if there are events where Council feels it would be appropriate. Finally, the name of the company remains as UKUUG Ltd, and invoices, purchase orders etc. will continue to be issued from UKUUG Ltd.

Council hopes that the new branding will encourage new members to join, attend events and get involved, and we will be monitoring the impact of the change over the coming months.

Spring Conference: March 2011, Leeds

Roger Whittaker

The Spring Conference this year was held in Leeds on 23rd and 24th March, with a tutorial the preceding day. The venue was the very civilised 3 Albion Place, in the centre of the city.

The tutorial this year was on Haskell, given by Edwin Brady of St Andrews University. This was an enjoyable and interesting introduction: as a functional programming language, Haskell challenges assumptions and forces different styles of thinking on those who are new to it. The morning session

consisted of an introduction to the language and some of the common functions in the standard library. The afternoon session went into the problem of interacting with a pure functional language and then used the concepts taught so far to show how a simple parsers can be built in Haskell. I found this a most interesting day.

Throughout the two days of the conference proper, there were two tracks most of the time. As always, that sometimes made for painful choices when both talks seemed interesting, but in general the programme was well planned, although there were occasions when one room became over-full.

On the first day I particularly enjoyed Simon Wilkinson's introduction to Git, which was very clear and well presented. Jan-Piet Mens gave three DNSSEC related talks in total, including a good introduction to the Unbound DNS server and its use as a caching validating DNSSEC resolver. Adrian Kennard's talk on the current state of IPv6 was illuminating and perhaps worrying.

The conference dinner was held at the Royal Armouries. The walk to the Armouries was just long enough for everyone to develop an appetite for dinner, and provided scope for lively (and even 'religious') discussion about whose smartphone had the best navigation app. After assembling in a room filled with armour and ancient oriental weapons, we sat down to an enjoyable meal with good conversation.

On the second day I gravitated towards the track with the talks about storage and virtualisation. John Leach spoke about ceph, and Stuart Teasdale discussed MogileFS as deployed by we7. These were both interesting talks about projects to do similar things (large scale distributed storage) but with slightly different design decisions.

Thierry Carrez spoke about OpenStack: it was interesting to see him one year on from his talk at last year's Spring Conference, now with a different employer and talking about a different cloud management solution. But there is no doubt which way the wind is blowing in this area, and his old employer has moved in the same direction.

In the afternoon, after a session on PostgreSQL, there was the usual session of lightning talks – unfortunately I could not stay for these.

This was another successful and well attended conference, with a good atmosphere and plenty of opportunities for discussion.

OpenTech 2011

Paul Thompson

This year's "informal, low cost, one-day conference on slightly different approaches to technology, transport and democracy" was held on 21 May at the University of London Union.

Whilst walking through Russell Square towards the event, I was wondering how many would-be attendees might instead decide to spend their Saturday soaking up the glorious sunshine, but was pleasantly surprised to see all the sessions very well attended. The organisation was better than I remember of previous years, and that added to a great day. Talks with similar themes were grouped together well so I didn't have to dash between seminar rooms to catch the tail-end of talks in other tracks this year.

Particularly interesting for me was Julian Tait's talk on FutureEverything's Open Data Cities programme. He recounted how in 2009 they realised that cities might evolve differently if data were made open and how that could reduce inequalities. They had then set out to try to change the way that public bodies within Greater Manchester share data. He detailed how they attempted to demonstrate to local authorities that there was demand for open data from the 2.6 million population, whilst also grappling with apparent disinterest from residents who didn't seem to care who provided services across Manchester's porous administrative boundaries. Challenges included authorities often not knowing what information they held, and being inefficient at handling that data. The breakthrough appears

to have been when they specifically targeted Trafford's transit department who agreed to make some transport data open. FutureEverything created apps around that data to build momentum, and Trafford subsequently identified 2000 datasets that could potentially be released. The hope is that Trafford will become a model for other authorities.

Also on the theme of Open Data, Jag Goraya gave an engaging presentation about how the OpenData Sheffield community had developed. His fascinating props included a "command-line time-phased histogram" showing how the demographics of the community membership had evolved, accompanied by a more familiar chromatic pie chart for traditionalists. The aims of the community include: to keep pace with the agenda of Open Data, improve data literacy amongst the wider population, and to "make cool stuff". His "lessons learned" would undoubtedly be useful for others pursuing a similar agenda with local government, and included a realisation that they'd engaged Sheffield Council too soon (should have done more of what geeks usually do: "to hack cool stuff in isolation then try to get buy-in later") and that they'd "bitten-off too big a bite of the exploding ODS pie" rather than stimulating grass-roots innovation. Interesting data-visualisation devices such as an "expressionist gant", a "sweat-equity stacked area chart" and a "heat-spectrum bubble chart" neatly illustrated that the end result of some of the community's activities didn't always justify the effort required.

The various talks around open data all carried the message that data isn't necessarily the property of institutions or companies, but of the citizens: information about when and where I swipe my bus pass is as much mine as it is the bus company's. However, changing to a more open model can't be achieved without the co-operation of those already controlling it – as they're the ones with the power, access and knowledge. Private organisations also have useful data that should be opened, but the citizens have to learn to consume (and contribute) that data and to exercise their rights to it: an illustrative example was given by Greg Hadfield of how, despite Brighton & Hove having talking bus-stops, talking buses and swipe cards, the only available app was written by someone in his spare time, obtaining the data via a circuitous route.

Other notable talks included an introduction to Open Spending: "Where Does My Money Go?" by Rufus Pollock of the Open Knowledge Foundation. An example of the reuse of open government data, it has teamed with OpenSpending.org to combine government spending datasets from other countries, so can scale-up to, for example, an EU level to see where most money is being spent – such as Common Agricultural Policy, then scale-down to the level of individual farms. They have also created an open source platform providing a simple solution for the exchange and visualisation of data. After-talk discussions became quite animated, with even suggestions that some of the "where your tax is spent" bubbles should be capable of being dragged by individuals to adjust their size, thereby giving citizens direct influence on tax policy.

Another interesting talk that I attended was about a collaborative translation project, 'Adopt a Paragraph', by UCL student Juliano Spyer. Motivated in 2009 by being frustrated at the effort involved in translating texts into his native Portuguese, he pasted a document into Google Docs and invited friends to translate one paragraph each. Within 24 hours that first document was completely translated. Juliano provided lots of advice for anyone considering launching such a project, including having Google Translate do the "donkey work" of translating any straightforward text, keeping the required investment in time of translators to an absolute minimum, giving credit to contributors, and maintaining a database of participants.

The organisers' hard work made for a very enjoyable day (and indeed evening, as the bar remained open long after the sessions ended), the quality of the speakers and the sessions was excellent – and I'm glad I didn't succumb to the fresh air and sunshine of Russell Square.

OpenTech report

Martin Houston

This was my first OpenTech and sadly I missed the first session because I got caught out not being able to change onto the Northern Line at Tottenham Court Road.

However the rest of the day was a great success. I am not going to just reiterate what was said as the schedule can be found at <http://www.ukuug.org/events/opentech2011/schedule/>

I really liked the format of talks in banks of three then a chance for questions. I also liked the fact that it was a 'why' conference rather than just the technical 'how' that software development conferences usually concern themselves with.

I sat through talks from three different areas: Gavin Starks on the amee project about collecting and collating data on CO2 emissions; Chris Taggart on constricting an open source database to shine a light on the interrelationships between companies worldwide (<http://www.opencorporates.com>) and Nick Bull on having unrestricted access to Case Law, including being able to index it in search engines so the general public has a chance to find things for themselves (<http://judgmental.org.uk/>).

I detected a common pattern.

Data is in a bit of a mess. Sometimes this is just through human mess-ups but often too vested interests want to keep it messy so they can charge a lot for simply finding things out that ought to just be a simple matter of a search.

We need good provenance so that particular data can be relied upon, something that the Amee Project in particular is working on.

It was a very intense day with three, sometimes four different streams of talks and a chance to meet like minded people who use computers as a tool to do something. Another common theme was requests for help with the technical side of projects such as websites and natural language processing skills. The event happened as much in twitter as in the halls so looking for the #opentech hashtag is a good way to catch up on the discussion and background material of talks you would have liked to have gone to but had inevitable timetable clashes.

The price (a suggested donation of a fiver) was no bar to going along as long as getting to Central London is not a problem for you.

The icing on the cake was that the whole conference was literally next door to a good old fashioned Computer Fair so I found myself stocking up on various hard to get bits and pieces too.

I think the large number of free software fans made an impression on the vendors of the reconditioned PCs and Laptops for sale. I had an interesting conversation with one and found that it takes them just 4 minutes to blast a minimal copy of XP onto a reconditioned business PC. With such little time involved it begs the question why second user systems not be sent out with a choice of Linux too? Even at the low end machines that have done their 3 years service in the corporate world have 40 – 80GB disks nowadays. Allocating 20% to Linux would give 8 – 16GB to play with, plenty for someone who mostly just surfs the web.

Getting Linux pre-installed on new PCs in the mass market has been a tough nut to crack but having Linux as well as XP gives the purchaser of a £100 reconditioned PC some real extra value, and at no real extra cost to the PC recyclers.

Find who does PC recycling in your area and offer to help them change their procedures to include Linux (whichever you prefer). It would be a very effective way to get many thousands of people to see that there is an alternative to just using the same old thing.

Let me know how you get on.

IWEEE 2011 Free Software for e-health and education in developed countries

Gerald Marie-Nelly

This event (International Workshop on E-health in Emerging Economies) took place in early April in Luxembourg and was organised by the GNU SOLIDARIO association during the Med-e-tel 2011 conference (Official event for the Society for Telemedicine and eHealth).

I attended the event because I am currently investigating Open Source solutions that would be relevant for non-profit Organizations, especially for education and healthcare purposes.

I gained interest in those matters after doing a 2 months charity mission in Haiti last year. I was working in an orphanage and teaching kids computer literacy with Open Source software in 4 schools, trained the charity staff in OpenOffice and setup a website for donations (<http://pwojeespwasud.org>).

That was a priceless experience but I went there without knowing what to expect. Knowing a bit more of the real problems on site, many projects could be potentially done in the area. Solutions for managing the kids/patients in local clinics with the orphans records or even setting up media labs for schools would make a tremendous difference. We see earthquake and misery through the media but there are projects down there which inspires lots of hope.

I have been looking for some OSS projects around healthcare and education online but it was hard to find any concrete feedbacks or case studies. A lack of centralised information was also an issue.

I decided to go straight to the community and found out about the IWEEE. My overall feedback about this event is that I was agreeably surprised to discover an international community and a plethora of strong OSS applications used in many Organizations. The emphasis was towards e-Health solutions in emerging countries and the coordination between the OSS ICT solutions in healthcare.

The first day was the proper IWEEE event and the two other days were FLOSS Tracks for the Med-e-tel parent event.

The IWEEE was launched by the organiser Luis Falcon (GNU SOLIDARIO) from Argentina with an inspiring presentation about the “social” aspect of Free software. Basically, seeing “Free Software” as “Social Activism” imply focusing on people instead of corporation in this context.

Mr Falcon is also responsible for the OSS project “Medical” which is based on openERP (a python based OSS ERP solution). It is a centralised Health and Hospital Information System that provides Electronic Medical Records (EMR), a Hospital Information System (HIS) and a Health Information System. It offers 30+ modules for different pathologies and also the logistics of healthcare Organizations. That project was cited many times in the different talks and is used by Hospitals and clinics around the world: <http://medical.sourceforge.net>.

Among the other talks during that day, a pertinent overview of the use of Open Source at a Governmental level in Brazil was done by Cesar Brod. Due to the ex-president Lula’s open source policies, OSS portals ran by the Government are acting as a catalyst for the national OSS community and most essentially for the country’s development. Many projects used by the Government in Healthcare and Education are created through those portals: <http://www.softwarepublico.gov.br> and <http://www.softwarelivre.gov.br>.

An excellent example is InVesalius, an application generating 3D images from Xray 2D scans: <http://svn.softwarepublico.gov.br/trac/invesalius>

The VistA (not the windows one) platform used in US hospitals was also mentioned. Several open source versions exist (WorldVista GPL, OpenVista AGPL, vxVista EPL) and have been broadly deployed in the states: <http://vistapedia.net/>

Another talk was done by the United Nation University (Institute for Global Health) from Kuala Lumpur in Malaysia. I was quite impressed by their initiatives around Open Source since I didn’t

even know that there was such a thing as a United Nations University which delivers degrees and PhDs. They implement an open CaseMix tool (<http://www.unuigh-casemixonline.org>) providing a classification of patients' treatment episodes. For instance, similar tools or services are used by the NHS or in Australia as a key funding model for reimbursement of the cost of patient care in hospitals.

They also publish a portal online with different courses around "Open Source Training in health care". There are only 4 full courses on the website but 10 of them are in progress at the moment so check out the website if you are interested: <http://unu-igh.org>

A nice project from Bika Labs in South Africa that got my attention was BIKA LIMS for Laboratory Information Management System. I was impressed by the maturity of the project and the stories of the project manager (Lemoene Smit) who have been working more than 10 years on it. The software is based on Plone/Zope and fully open source. It allows to manage a laboratory workflow from sample reception to result publication.

It is worth mentioning that Bika Labs has also different branches for different industries such as water quality management, wine and beverage industry, geology and mining (<http://www.bikalabs.com>.)

I will not describe each talk done during the two following days (Med-e-tel FLOSS tracks) but the ones that captured my attention and also those focused on providing platforms centralising different OSS projects in healthcare.

Regarding the issue of centralising information about OSS for healthcare, Thomas Karopka and Holger Schumhl presented the medfloss.org portal during a talk about governance for FLOSS in healthcare. 230 projects were listed on the website during the talk, this is the ultimate place to seek for Open Source software in healthcare or submit a project if you are involved or interested. Holger Schumhl also referred to the Open Health Tools teamforge portal but that was definitely more oriented to the practitioners with many standards and protocols of the healthcare industry provided by those tools (<http://www.projects.openhealthtools.org>).

Another place listing many projects around this topic is DebianMed presented by Andreas Tille from Germany. A Debian repository with many projects properly packaged in DEBs since many scientists tend to just provide sources when it comes to OSS practices. The repository is currently focused on medical practice, imaging and bioinformatics. See <http://wiki.debian.org/DebianMed>.

The last day, one track was more oriented towards AAL or "Ambient Assisted Living" projects governance. I didn't know what was AAL before which is about making smart use of technology to support well-being in the preferred living environment for people who might otherwise find this difficult (e.g. infirm or very elderly people who want to continue living in their own homes). No OSS projects were described but a manifesto and an open platform were presented: <http://aaloa.org/manifesto>, <http://www.universaal.org>.

On the other track, a Spanish company called Andago made a good impression with their OSS products for eHealth and eGovernment. They offer a new mobile health platform (web and android), Open Health Assistant, that is generic and flexible enough to enable the creation of various kinds of chronic disease management solutions, tailored to both care providers as the patients themselves. The talk was more 'marketing' orientated but it looked good and promising on the slides, you can see more at <http://andago.com>.

The last session was a workshop for "Collaborative care team in open source" by Etienne Saliez. The presentation was focused on the chronology of ICT in healthcare, from the paper-based archives to the personal computer and the current distributed environments. The confidentiality and economic issues were also treated with a proposed coordination logic model. The outcome of the workshop was the creation of a Working-Group by Monika Hubler (University of Basel, Switzerland) with all the attendees to carry on the discussion around the main topic of governance for a fragmented FLOSS community in healthcare.

Overall, the IWEEE was definitely the place to be if you needed to have a broad overview of the Open Source solutions used in Healthcare and eventually Education. The talks were highly informative, the discussions pertinent and the networking also was quite enjoyable. I personally recommend this event and will follow some projects that can fit with some charity missions that I'm affiliated with.

The £100 PC

Martin Houston

The under £100 cheap PCs for the disadvantaged have just gone on sale.

<http://www.ecycleonline.co.uk/choose-your-computer---from-9200-8-c.asp>

Ubuntu 10.10 is there as a solid (and 'no strings attached') choice.

But for only £3 more so is Windows XP and Office!

This is very bad news for the disadvantaged, for the rest of the internet community who are going to have to put up with millions of new potential malware filled zombie computers to deal with.

The users also have to agree to special licensing terms as Microsoft is doing them such a BIG favour (not).

Why would Microsoft do such a thing? Is it not trying to end of life XP?

It seems that keeping several million new computer users away from the freedom of running and learning about the free software that the Internet itself tends to run on.

We in the free software community should not take this clumsy piece of vote buying lying down. Make a fuss.

Netbooks were a great chance to free people from having to run Windows. However Microsoft came in and changed ground rules, forced netbooks to be non upgradeable and carrying battery sucking hard disks because XP was not so hot about running from SSD.

We don't want the same thing to happen in the second user computer marketplace now it is becoming significant.

Five year old PCs are just fine for running Linux for most peoples use. They really don't need to be distracted by XP and weasel worded licences.

Government Consultation on Free and Open Source Software

Martin Houston

The UK Government would like our help.

Those of you at the spring conference may remember that I mentioned there that the Home Office is in the process of collecting a list of Free and Open Source software that has a good track record of being in use and of use. The purpose of this list is to give suggestions to the system integrators of technologies they could use in constructing complete systems for Government of NHS use.

Having a list of trusted technology building blocks such as Apache, Python and Linux means that systems need not have to consist of many proprietary components that cost the UK taxpayer an arm and a leg in ongoing software licence payments.

Mr Tariq Rashid the Lead Architect of the Home Office IT Technology Solutions and Assurance Team has put together a Google Docs site where you can review the current proposed list and suggest extra packages to be included on it.

See also: <http://www.osor.eu/news/eu-uk-fsfe-appeals-for-information-on-oss-deployments> where there are links to the consultation document and to the related appeal for information by the FSFE.

Designing Interfaces

Jenifer Tidwell

O'Reilly Media

ISBN: 978-1-4493-7970-4

576pp.

£ 38.50

Published: January 2011

reviewed by Lindsay Marshall

I am conflicted about this book. It is, as the blurb says, a genuinely comprehensive look at “interface patterns” and it should be useful to anyone who is building interfaces, but...

First, the title – this is not a book about designing interfaces; it’s a dictionary/catalogue/encyclopedia of approaches to solving particular interface problems, but it doesn’t tell you how to design interfaces. Second, and this is just me of course, I find the whole idea that these are “patterns” just annoying and wrong. “Prominent Done Button” is only a pattern if you stretch the definition a long, long way, way past breaking point – this doesn’t devalue the content of course, just makes it annoying for me to read. YMMV, and almost certainly will. Third, even though I’ve described this as a dictionary/catalogue etc., there isn’t really a quick way to find a particular solution to a problem you may have. Looking in the table of contents and seeing “Pyramid” doesn’t help me identify whether or not I need to go and read that item.

I’m being picky. The content is good (I like the mobile section particularly), the production values are high: colour, glossy pages and much the heaviest O’Reilly book I have handled recently. If you work with interfaces and can afford to buy it then you should, as you will use it. It is certainly one for the interface development team library at the very least.

Vi and Vim Editors Pocket Reference

Arnold Robbins

O'Reilly Media

ISBN: 978-1-4493-9217-8

94pp.

£ 11.50

Published: February 2011

reviewed by Andy Thomas

Another gem in the O’Reilly Pocket Reference series, this is a long overdue and much needed second edition of a pocket book that originally appeared 12 years ago as a companion to Learning the vi Editor by the same author. Times change, editors evolve and become increasingly feature-laden and the rising popularity of Vim is now reflected in the title. But the book also covers elvis, nvi and vile quite extensively despite the fact that nowhere on the front cover is this mentioned, which seems a curious omission that could impact sales of this excellent book; elvis is the default editor in Slackware Linux, for example.

Omitting the formal structure seen in some other books in the series, there are no chapters or even page breaks separating the different sections – this has allowed the author to cram as much meat into this

85 page book as possible and he wastes no time in doing so, jumping straight into the basic editing functions common to all variants on page 2. Starting with screen mode commands and moving on through startup and recovery options and substitution tricks to the ex commands, explanation of vi's three modes of operation – command mode, screen mode and ex – switching between them and how they relate to each other is rather lacking but then this is not a vi tutorial. And comparing this with the full size book, it looks as if almost every command, option and parameter in the latter has somehow been squeezed in with nowt left out – it even manages to find space to cover the differences between the Solaris implementation of 'traditional' vi and others.

Moving on to the vi clones, it's perhaps no surprise that Vim gets the lion's share of the treatment and this is really extensive, opening my eyes to features of Vim I didn't know even existed. The special features of nvi, elvis and vile are then covered and the book rounds off with a page of URLs to vi-related resources, a useful page listing the authors, email addresses and source download sites for vi and its clones and finally, an index.

O'Reilly's pocket reference series are a real boon for many as they are designed as an aide memoire for those who already have a basic knowledge of the subject, be it C, Gimp, PHP or vi. For those of us who do a bit of everything and become jacks of all trades, this series is indispensable – not all of us write PHP or use Gimp day in, day out for months on end and we tend to forget the more obscure things 2 or 3 weeks after using it. But there is another aspect to these mini-references – they are very useful when used side by side with the full size reference book when learning a new package as nowhere in "the big book" are the commands tabled one after the other with such concise descriptions of what they do and this is particularly true of the Vi and Vim Editors pocket book.

And even used on their own, they remain a great learning resource as I had a lot of fun trying out all the Vim functions I didn't know about, spending several hours working my way through the command lists instead of getting on with writing this review...

I found it a very enriching experience, although I doubt I'll be able to recall all of it in 3 weeks time – which is precisely what this book is for!

Finally, like the others in the Pocket Reference series, this is a true pocket book that will fit into any shirt or jeans back pocket, slim enough not to do the the wearer an injury if he/she sits down forgetting it is there. Many moons ago when I repaired TVs for a living, Newnes used to publish a range of "pocket books" for engineers – only a little larger than the O'Reilly books of the same name but so thick they would not fit any normal garment's pocket (although I suppose they'd fit the pockets of the voluminous white lab coats all the old hands in the trade wore at the time). And I love that creature on the front cover – well done Arnold and the O'Reilly Pocket Reference team!

Programming Python

Mark Lutz

O'Reilly Media

ISBN: 978-0-596-15810-1

1632pp.

£ 49.99

Published: January 2011

reviewed by Andy Thomas

Leaning back in my chair and contemplating the bookshelves as I ponder how to start off this review, I notice the middle of the bottom shelf is sagging badly, sagging mostly under Sendmail, 3rd edition which I had always regarded as O'Reilly's weightiest (and loftiest) title to date. All 1200 pages of it. Until now that is – Mark Lutz's Programming Python, 4th edition comfortably beats that at getting on for 1600 pages and is about the heaviest book I have ever picked up.

The centrepiece of a trilogy, where it is partnered by Learning Python on the one hand and the Python Pocket Reference on the other, this book makes it clear from the outset that it is neither a tutorial nor a Python reference – the reader needs to look elsewhere for that. The emphasis is very much on application programming while the core Python fundamentals are covered in the Learning book – there is no full-sized reference book, at least not yet.

This latest edition covers Python 3.1 only – this means only those parts of earlier versions of Python that have lived on into 3.1 are covered and nothing else. The author explains this was to keep down the page count of this already large book but it seems a shame that an all-encompassing book such as this doesn't at least cover 2.5 and 2.6 which are still being bundled with many Linux distros at the time of writing. However, we need to move on at some stage even if it does make for version incompatibility and experienced programmers should have little difficulty adapting some of the concepts and code examples explored here to Python 2.x.

After a lengthy preface we finally get into the book proper and find that it is organised into six parts; part 1 is a short and gentle introduction to applications programming in which we are shown in six stages how to create a flat file database and then further augment it using the pickle module to ensure consistent record storage. Subsequent stages introduce OOP Python style, adding persistence and console interaction before moving up to a full GUI and finishing with making the application web-accessible, preparing us for the in-depth applications programming techniques covered in the remainder of this tome.

Part 2 is about systems programming, containing six chapters on Python's extensive library of modules for system file, directory and other functions. *n*x users should not be discouraged by the fact all the examples shown here (and indeed, the whole book) are for the Windows command shell with references to using Cygwin where appropriate – there are occasional references to Linux in the text but none at all in the massive index at the end of the book, although UNIX gets a few lines here. This section ends with some really quite useful complete systems utilities although some seem to be Windows specific.

Jumping in a bit deeper in Part 3 we come to GUI programming. Five very large chapters are mostly devoted to tkinter, the Python interface to the Tk library – I had some problems running some of the examples on my Linux system and had to tinker for ages with installing tkinter before anything worked. But users of mainstream Linuxes and MacOS bundled with tkinter-aware Python packages should find these will work out of the box and there are some useful image manipulation examples as well as the complete programs that close each part of the book.

Internet and network-aware applications follow in part 4 which for many potential readers will be the most interesting part of the book, as this and GUI programming (part 3) get by far the largest treatment over chapters 12 to 16 covering in turn network scripting, client-side scripting, a complete working email client (PyMailGUI) that illustrates the techniques learned in the previous chapter, server-side scripting and finishing with PyMailCGI, a working webmail-style system. Python's network support is very standards-compliant so readers using *n*x will perhaps have more fun with the code here than with the GUI section and I certainly felt much more at home here; a lot can be learned from the concepts here that experienced programmers may find useful when writing in other languages.

Tools and Techniques is the topic of part 5, a four chapter round-up of some of the techniques used in earlier chapters including information storage and SQL integration, advanced data structures using stacks, binary search trees and graphing followed by a further two chapters on text and XML parsing and integration with other languages and their libraries. This section will probably be of interest to those who wish to use Python for scientific and mathematical computation work.

Titled simply 'The End', the final part of the book is entirely non-technical and largely philosophical look at computer programming in general, what the author thinks is wrong with current programming and development methodologies and why Python comes to the rescue, galloping in as the Knight in Shining Armour and does things "the right way".

A well-written book that explains complex concepts comprehensively yet clearly, with the occasional references to well-known Monty Python catch-phrases adding a touch of humour, this book is firmly in ‘advanced’ territory as far as the intended readership goes. But given the increasing prevalence of Python scripting even in such mundane areas as operating system install scripts, there is clearly a need for a book at this level and I think this is about the only book that fulfils this need. To get the most from this book, consider also buying the companion learning and pocket reference books and if you do buy the complete set, you’d better have a strong bookshelf.

HTML5 and CSS3

Brian P Hogan

O’Reilly Media

ISBN: 978-1-934356-68-5

280pp.

£ 25.50

Published: January 2011

reviewed by Daren Chandisingh

According to the prerequisites, “HTML5 and CSS3: Developing with Tomorrow’s Standards Today” is aimed at developers with a good understanding of HTML and CSS, as well as those having a basic understanding of Javascript in general, and jQuery in particular. I would echo these sentiments; it’s not a book for the complete beginner. The book leans quite heavily on jQuery, and while there’s a primer in the appendices, a reader who does not have appreciation of Javascript – and its potentially odd-looking syntax – is as likely to be left bewildered as he or she is enlightened.

The book begins with an overview of new additions to HTML5 and CSS3, and deprecated tags that are commonly encountered, along with a caveat that these are specifications that are in development, so could change at any time.

“HTML5 and CSS3” is presented in three parts. Broadly speaking, the first part covers new semantic and structural elements and attributes; e.g. header, nav and section elements. Part two covers more presentational aspects such as the canvas, audio and video elements, and new capabilities of CSS3 such as rounded corners, gradients and drop shadows. Part 3 covers aspects of “HTML5” which are neither markup nor presentation, and have either been spun off from the HTML5 specification, or which were never part of the specification in the first place; these include local storage, web sockets and geolocation.

Each item is introduced in the context of a realistic example, covering not only the “what” of new features, but also the “why”. I found the style quite abrupt, and occasionally lacking enough introduction to blocks of code, be it the HTML, CSS or Javascript that peppers the pages. Similarly, I sometimes read comments within the text that didn’t seem to follow or seem obviously connected to the preceding block of code. As noted before, this book is intended for intermediate developers at least, and I think a degree of inference and internal linking of dots is expected.

Most features are accompanied with a fallback proposition for browsers that don’t yet support the item under discussion; these fallbacks are largely Javascript dependent, and are accompanied with examples of how to detect whether the feature is supported. This was one of the most useful aspects of the book, I felt, as I’ve found it easier to find out what “HTML5” can do, than I have found material that enables me to use the features with confidence that my pages won’t be unusable for a significant number of my audience.

The author also tries to keep the accessibility aspects of new features in the reader’s mind, both offering examples of how the new elements and attributes can help disabled users, as well as highlighting any downsides of the fallback propositions.

This book contains a good amount of useful information for the developer who wants to get stuck in to using HTML5 and CSS3 right now. However, each feature's coverage is short, before moving swiftly along to the next item, so this is something of a whistle-stop tour. Some work is required on the reader's part in order to make the most of what's presented, but it's certainly a sound launchpad for exploring the new frontier.

Mining the Social Web

Matthew A Russell

O'Reilly Media

ISBN: 978-1-4493-8834-8

360pp.

£ 30.99

Published: February 2011

reviewed by Paul Waring

Many people are a member of at least one online social network, whether it be Twitter, Facebook, LinkedIn or even a collection of emails and mailing lists. This book aims to show you how to pull together, filter and analyse the rapidly growing corpus of data available and turn it into something useful.

The introduction chapter is pleasingly short on flowery prose, and by page 4 we're already connected to the Twitter API, followed a few pages later by an example of frequency analysis and lexical diversity. The authors have sensibly decided to re-use existing open source tools rather than writing their own, for example graph data is output as DOT language, which can easily be visualised. As a result, by the end of the first chapter we are already writing scripts to produce graphs showing the connectivity of retweets.

The next few chapters go into more depth in data collection and analysis. There is an entire chapter dedicated to email, which contains a trove of useful information about interactions but is probably not the first thing most people think of when discussing 'social data'. Broad coverage is also given to the topic of using CouchDB to store and process social data, particularly its map/reduce functionality.

Whilst most of the first half of the book focuses on Twitter, chapters covering LinkedIn and Google Buzz (which feels slightly wasted, given that this network failed to take off) bring some variety. Part way through the Google Buzz chapter the mathematical symbols and formulae rear their heads, taking me back to my time spent in academia where I brushed against the natural language processing research area. Facebook also gets a chapter, although the speed at which its platform changes means that some of the information may be out of date, but the principles remain the same.

Overall, this is a good starting point for anyone looking to take large chunks of social data and produce summaries and graphs. You do need to have a good grasp of natural language processing to make the most of the text beyond the first half of the book, and probably a reasonable understanding of Python too, for which Natural Language Processing with Python may prove useful.

REST in Practice

Jim Webber, Savas Parastatidis and Ian Robinson

O'Reilly Media

ISBN: 978-0-596-80582-1

448pp.

£ 34.50

Published: September 2010

reviewed by Paul Waring

Having already purchased and read the excellent RESTful Web Services (O'Reilly 2007), I was pleased to see a new and up to date book covering this topic. For anyone who hasn't come across REST (Representational State Transfer) before, the basic concept is to use an existing protocol's (usually HTTP) capabilities instead of implementing this functionality for each application which requires it. For example, HTTP already supports a variety of authentication methods which can be used instead of having to reinvent the wheel.

The book begins with an introduction which succinctly covers all the main concepts and jargon related to REST, with some excellent diagrams for those who prefer graphical explanations to textual ones. This is followed up by two chapters demonstrating the basic operations involved in REST, centred around the example of a coffee shop. The chapter covering an example of an ordering service is helpful, but unfortunately uses .NET for all the example code, and in my experience the HTTP and web client libraries provided do not work well across different platforms.

The chapter on Scaling Out is particularly useful, covering not only how to cache content, but just as importantly how to make content cacheable. Two chapters dedicated to Atom seem slightly wasted though, and the authors could perhaps have culled one in exchange for detailed coverage of another topic.

The topic of security gets a dedicated chapter, covering not just authentication and authorisation but also transport-level confidentiality and integrity. The performance aspects of security are also covered, including the overhead of cryptographic operations and the lack of caching which comes with HTTPS. Finally, the book is rounded off with a chapter of how REST fits in and compares with other well known concepts associated with web services, such as SOAP and WSDL.

Overall, this is an excellent overview of REST, with plenty of clear, crisp diagrams and a broad coverage which takes in everything needed to get started.

jQuery Pocket Reference

David Flanagan

O'Reilly Media

ISBN: 978-1-4493-9722-7

160pp.

£ 9.99

Published: January 2011

reviewed by Mike Fowler

This book serves as an introduction to jQuery and provides very thorough coverage of all the aspects of the library. Though a reference for the API is provided at the end of the book, the book feels more like a pocket guide than a pocket reference. In fact the whole book is the jQuery chapter from the author's much larger Javascript: The Definitive Guide. This has the advantage of making a very readable little book however the sidebars are preserved and look quite odd as many overflow to a subsequent page. Given the format of the book I would have preferred to see the sidebars integrated into the main text for improved readability.

Each chapter of the book gradually introduces you to more and more features of the library. The author has assumed that you know JavaScript but not jQuery which helps keep things to the point. The examples are simple and clear and not contrived as examples often are. The author takes the time to explain why the library works in the way it does and what browser limitations the library helps you to work around. Amusingly the author also provides some critique of the library, particularly where the naming is confusing. For example the `jQuery.ajax()` method has an option called `type` which the author rightly points out would be better named `method`.

jQuery experts probably won't find this book of much use as the library moves so quickly the API reference is already somewhat out of date. The book covers version 1.4 and we are already on version 1.6.1. However for people who only dip into jQuery occasionally or who have no previous knowledge of jQuery, this book will serve as an excellent introduction and refresher.

Learning the iOS 4 SDK for JavaScript Programmers

Danny Goodman

O'Reilly Media

ISBN: 978-1-4493-8845-4

318pp.

£ 26.99

Published: December 2010

reviewed by Mike Smith

A brand new first-edition book, this one (well, printed at the turn of the year) but the very first thing I notice is the diagonal flash across the top right hand corner – “Covers iOS SDK 4.2”! I presume this is to demonstrate how new and up-to-date the book is, but I'm not sure of the wisdom of this technique. It is of course already out of date.

I've been doing a little tinkering with iOS recently (well, on and off for a few years now) and so this title grabbed my attention. Another worry I have for O'Reilly and the author, which I'll share with you before getting into the content, is that there is a danger that the way the book is positioned limits the readership base. How many potential iOS developers are there just starting out (yes, probably quite a lot); but more importantly how many of those have a JavaScript programming background? I just wonder why buy this book when there are other more general iOS texts about. (I think a better title to draw people in would have been “iOS development for web developers”, or something similar.) It reminds me of the new near field communication phones coming out – Orange are making a splash with one at the moment, but you have to be on their network, have to have a Barclay card to top it up, and have to have the specific model of phone – all limiting the potential audience. anyway, I digress, onto the review.

There's an initial positioning chapter that discusses the relative benefits and pitfalls of native application development versus web-based apps. The author talks about his early experiences of the iPhone – life before the SDK, so I think that's one reason for the inclusion of this material. This book seems to be drawn from much personal experience.

The next chapter walks through the SDK installation and loading up a sample project. Unfortunately, again the world is changing here. I think as a registered developer you can still download the 4.3 version of the SDK from the web, but the latest (and nicely integrated version with Xcode and Interface Builder all together) is now only available through the Mac app store (at a cost too, the meanies).

The next few chapters gently introduce us to developing our first application, discuss iOS frameworks and object-oriented coding. Actually this is one area where a direct comparison is made between creating classes in JavaScript and performing the same activity in Objective C. It's not until page 92 that model-view-controller is mentioned. I'm jumping ahead here, but it's drummed into you (at least in many other texts I've read) at an early stage that this is the way it's done in the Apple/Next/iOS

world. Excluding the appendices this book has 165 pages, so we're over a third of the way through by now. A mate of mine has written a PHP MVC framework which is quite cool (some great wizardly Apache rewrite rules make it all work automatically: impressive.)

Data types and memory management up next, and discussions on various topics – with mention of C and C# too. Things like structures, variable scope – a whole mix in chapter 7. But not the special classes in Cocoa – this is saved for chapter 8. Actually we cover similar subject areas – C arrays in chapter 7 followed by NSArray, NSMutableArray etc in chapter 8.

The purpose of Chapter 9, and in fact the final chapter of the book, goes to the core of the title of the book. Performing common JavaScript tasks in cocoa touch. Or that's what I thought – but actually it's just a collection of, yes, useful routines that actually you want to perform in any language. I didn't spot any actual JavaScript code here – just a collection of Cocoa/Cocoa Touch recipes. Useful nonetheless.

A couple of appendices are included. One on using the Xcode documentation, and a brief one on common compiler errors. The first one is useful to point you in the right direction; I have my doubts about the latter.

So I have quite a few reservations about this book. It's not tremendously long, the content is a bit mixed up, I'm not convinced of the whole premise and I didn't see that many direct comparisons between the JavaScript and Objective C environments. Don't let me put you off if you do fall into the target demographic; have a flick through in O'Reilly's Safari or in Waterstones, but I'm afraid I'm not convinced this has a place on many peoples' bookshelves.

Learn You a Haskell for Great Good!

Miran Lipovača

No Starch Press

ISBN: 978-1-59327-283-8

400pp.

£ 35.49

Published: April 2011

reviewed by Roger Whittaker

I have been dabbling just a little with Haskell in the past year. As an (ex-)mathematician, I could not help being intrigued by the idea of a functional language in which you can define infinite lists, in which valid code can sometimes look pretty much like mathematical notation, and in which type classes have rules that look like (and are) mathematical axioms. So I was delighted when the tutorial at the Spring Conference this year turned out to be an introduction to Haskell. And likewise it is good to have the opportunity to comment on this book.

I certainly cannot claim to be any kind of expert on Haskell. But comparing this book with the other introductory Haskell books that I have looked at, I am very impressed with the quality of the explanations and examples. There's a slight air of zany wackiness running through the book, from its title to the illustrations to some of the examples. This atmosphere, combined with the author's infectious enthusiasm for the subject matter, makes the book a lot of fun to read. Everything is explained alongside short snippets of code which serve as direct and self contained examples. Reading the book while testing out the examples in `ghci` is easy to do, and the author doesn't make the mistake of putting down chunks of code on the page that are too big and tedious for the reader to test in practice.

The book starts off at a fairly gentle pace and the first few (of the fifteen) chapters are easy going. Then the learning curve gets steeper. I spent quite a lot of time working through the examples in the chapters on Applicative Functors and Monads, and the clarity of the examples and explanations staved off the "head explodes" experience that I was expecting to happen at any moment. I particularly liked the example in the Monads chapter of the tightrope walker carrying a pole on which birds may land on either side, sometimes causing him to fall.

There were a few places where it was not clear which modules needed to be imported to make the code examples work (but experimenting to discover the answer was instructive in itself), and there is at least one small proof reading error in a code example.

The book started out as a set of tutorials on line at <http://learnyouahaskell.com/> which is all still available under a Creative Commons licence. The online content differs slightly from the book, but not greatly. Even the slightly odd pictures (which seem to be by the author) have been reproduced from the on-line version.

If you are interested in finding out what Haskell is all about, this book can be strongly recommended.

Contributors

Daren Chandisingh wrote his first HTML page in 1995, an event that served to steer him away from Windows application development, and towards the land of GNU/Linux and FOSS. He currently works as a Web developer at Lancashire County Council.

Mike Fowler is a Senior Software Engineer at Dotted Eyes. He is a firm open-source advocate and has helped companies move from Microsoft based solutions to complete Linux/PostgreSQL stacks. He is a major contributor to YAWL, an open source BPM/Workflow system. When not melded with a keyboard, he spends his time playing as many different stringed instruments as he can.

Martin Houston has been using Unix since 1980 starting with a PDP11/70 and Linux since 1994. He was the founder of the original UKUUG Linux SIG and is currently a freelance Linux consultant. He lives in Essex with wife Teresa and 2 cats, 2 dogs and too many computers.

Gerald Marie-Nelly is a freelance Linux Consultant from Martinique (French Caribbean) working in London via his company Nawflaw Ltd (<http://nawflaw.net/>). He has worked for Red Hat in the past and is currently contracting in the City and studying management during evening classes. Apart from geeking around, he likes broadcasting Caribbean music online and takes part in various charity projects.

Lindsay Marshall developed the Newcastle Connection distributed UNIX software and created the first Internet cemetery. He is a Senior Lecturer in the School of Computing Science at the Newcastle University. He also runs the RISKS digest website and the Bifurcated Rivets weblog.

Jane Morrison is Company Secretary and Administrator for UKUUG, and manages the UKUUG office at the Manor House in Buntingford. She has been involved with UKUUG administration since 1987. In addition to UKUUG, Jane is Company Secretary for a trade association (Fibreoptic Industry Association) that she also runs from the Manor House office.

Mike Smith works in the Chief Technology Office of a major European listed outsourcing company, setting technical strategy and working with hardware and software vendors to bring innovative solutions to its clients. He has 20 years experience in the industry, including mid-range technical support roles and has experience with AIX, Dynix/ptx, HP-UX, Irix, Reliant UNIX, Solaris and of course Linux.

Andy Thomas is a UNIX/Linux systems administrator working for Dijit New Media, Imperial College London and a number of small companies. Having started with Linux when it first appeared in the early 1990's, he now enjoys working with a variety of UNIX and Linux distributions and has a particular interest in high availability systems and parallel compute clusters.

Paul Thompson is based in London and works for Novell as a Linux consultant specialising in the financial sector.

Paul Waring is chairman of UKUUG and a director of a wholesale insurance broker. Outside of work he can usually be found filing documentation bugs against various open source and free software projects.

Roger Whittaker works for Novell Technical Services at Bracknell supporting major Linux accounts in the UK. He is also the UKUUG Newsletter Editor, and co-author of three successive versions of a SUSE book published by Wiley.

Contacts

Paul Waring
UKUUG Chairman
Manchester

Phil Hands
Council member
London

Martin Houston
Council member
Chelmsford

Kimball Johnson
Council member
Lancaster

Holger Kraus
Council member
Leicester

Niall Mansfield
Council member
Cambridge

John Pinner
Council member
Sutton Coldfield

Howard Thomson
Treasurer; Council member
Ashford, Middlesex

Jane Morrison
UKUUG Secretariat
PO Box 37
Buntingford
Herts
SG9 9UQ
Tel: 01763 273475
Fax: 01763 273255
office@ukuug.org

Roger Whittaker
Newsletter Editor
London

Alain Williams
UKUUG System Administrator
Watford

Sam Smith
Events and Website
Manchester