



# 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 4**

**ISSN 0965-9412**

**December 2011**

---

## **Contents**

<b>From the Secretariat</b>	<b>3</b>
<b>Chairman's Report</b>	<b>3</b>
<b>We won and we didn't notice – a conversation with Jeremy Allison of Samba</b>	<b>4</b>
<b>The Amazon Kindle</b>	<b>8</b>
<b>Thoughts on Linux User Groups</b>	<b>9</b>
<b>The Occupy Movement</b>	<b>10</b>
<b>Book review: Learning Perl</b>	<b>11</b>
<b>Book review: The Book of Ruby</b>	<b>12</b>
<b>Book review: HTML5: The Missing Manual</b>	<b>13</b>
<b>Book review: Programming Android</b>	<b>14</b>
<b>Book review: CSS Pocket Reference</b>	<b>15</b>
<b>Book review: Programming Social Applications</b>	<b>15</b>
<b>Book review: JavaScript Web Applications</b>	<b>16</b>
<b>Book review: Head First jQuery</b>	<b>16</b>
<b>Book review: Perl Pocket Reference</b>	<b>17</b>
<b>Contributors</b>	<b>18</b>
<b>Contacts</b>	<b>19</b>



## From the Secretariat

*Jane Morrison*

The UKUUG Ltd Annual General Meeting was held on 22nd September. The minutes and associated documents are available at <http://www.flossuk.org/Events/AGM2011>. Printed copies are available from the office on request.

Current Council members are: Paul Waring, Howard Thomson, Phil Hands, John Pinner, Holger Kraus, Kimball Johnson and Ian Norton. Paul Waring continues in the role of UKUUG Chairman and Howard Thomson continues as Treasurer.

We are currently putting in place a full schedule of events for 2012 – to date the following have been agreed:

- Perl Tutorials – 2 day Intermediate (including practical sessions) – 21st and 22nd February – and 2 day Advanced (including practical sessions) – 23rd and 24th February – London (see flyer inserted with this Newsletter)
- Spring 2012 Tutorial and Conference – 20th, 21st and 22nd March 2012 – Edinburgh – (see insert with this Newsletter)
- Intensive Introduction to Google App Engine with Python – tutorial by Paul Barry – 12th April 2012 – London (see flyer inserted with this Newsletter)
- Python Tutorial by John Pinner – details to follow – May 2012

We are also planning more tutorials for 2012 – if you have any particular topics that you think would be of interest please let us know.

I would like to remind you of a couple of benefits we have in place for members – we have a page on the web site <http://www.flossuk.org/Consultants> which lists members who can provide consultancy services. This is free to members: take a look and send me your details if you are a Consultant and get some advertising for free.

Also, I would like to remind you that in addition to the Reilly book discounts (40% on-line and 37.5% via the Secretariat) we also provide other discounts for members these include: 2nd Quadrant (10% off PostgreSQL courses), GBDirect (10% off various courses), Bytemark Hosting (10% discount on all virtual machines) and Opendgear (10% discount across a full range of products). See <http://www.flossuk.org/OtherDiscounts>.

The annual membership subscription invoices will be sent out in January, please look out for your invoice and as always prompt payment will be gratefully received!

We would like to thank the continued support of sponsors including our Gold Sponsor member, SUSE. Their involvement does help us to keep event fees down.

I should like to take this opportunity to wish you all a very Happy Christmas and a Peaceful New Year. The Secretariat will close on 19th December and re-open to January 3rd 2012.

The copy date for the next issue of the Newsletter (March 2012) is 17th February, and interesting submissions from members are always welcome at [newsletter@ukuug.org](mailto:newsletter@ukuug.org).

---

## Chairman's Report

*Paul Waring*

### Unconference

Our second unconference was held on 8th October at the Manchester Conference Centre, which we have used several times for our Spring conferences. Several local groups attended, including

---

Manchester Free Software and the North West England Perl Mongers, and we had speakers for every available slot on the timetable. A full review can be found later in the newsletter.

### **AGM**

Our AGM was held in London on 22nd September in London. Martin Houston, who was co-opted onto Council earlier this year, was elected, and Holger Kraus was re-elected for a second three-year term. Although there was a vacant space on Council following the AGM, I am pleased to announce that Ian Norton (who has spoken at previous conferences) volunteered to step in, and was co-opted onto Council at our last meeting. Ian will stand for election at the 2012 AGM.

### **Get Involved**

UKUUG exists to serve its members and we are always on the lookout for people who are keen to get involved in any capacity, whether that be through volunteering to help with organising events, writing newsletter articles, or entirely new activities which haven't been tried before. We are particularly keen to get members involved in running next year's Spring conference. If you would like to help out in any capacity please do get in touch via [office@ukuug.org](mailto:office@ukuug.org).

---

## **We won and we didn't notice – a conversation with Jeremy Allison of Samba**

*Richard Hillesley*

On a recent visit back to the UK, lead Samba developer Jeremy Allison met up with Richard Hillesley. Richard here relates Allison's description of the history of his involvement with open source, Linux and Samba.

Jeremy Allison is a steelworker's son from Sheffield in South Yorkshire and a Samba developer who has lived in California for twenty years or more, but who has retained much of the accent, humour and politics of his youth.

He was in the UK during the recent riots and "got to enjoy them first hand", but says he doesn't miss much about England other than the company of his family and friends and an occasional hankering for a jar of Branston Pickle or a tin of baked beans; the kind of things you don't imagine you'll miss until you are living somewhere else.

In California the sun always shines and he is paid to work on the things he would be working on anyway: coding and debugging Samba.

Samba is a free software re-implementation of the SMB/CIFS networking protocol originally developed at IBM and used by Windows. Samba is used by most Linux and Unix distributions to provide full file client and server interoperability with Windows networks, but was first developed to interact with DEC's Pathworks implementation of SMB.

Samba began life in 1992 as "a bit of a hack" by Andrew Tridgell on his university computer in Canberra, Australia, while "procrastinating the stuff [he] was supposed to be doing" for his PhD. The initial hack [1] was to use DOS to mount disks on a Sun workstation through DEC Pathworks running on Vax and Ultrix networks, and it matured rapidly as a means to share files on Windows networks with Linux and Unix servers.

### **Richard Stallman and the GPL**

Allison has been involved in free software since before Samba and the arrival of Linux, and began to work on Samba early. "I came across the GPL early in my career", he says, "and it immediately struck me as a good idea. I had the honour of having one of my patches (to add position independent code support to GCC on the Sun 386i) rejected by Richard Stallman himself. It always reminds me

of that great web-magazine headline from The Onion: ‘God answers crippled child’s prayer. “No”, says God.’ With a start like that, I never looked back.”

The rejection “wasn’t as nasty as it sounds”, he says. “The patch was crappy but worked for me. I didn’t meet Richard until much, much later. The first time was at a show in Monterey where he stood up and called John Ousterhout ‘a parasite’ because he’d released a proprietary version of the TCL ToolKit, and Tim O’Reilly ‘a parasite’ because he wasn’t shipping free documentation. . .”

“I won’t say I am friends with Richard, but that’s not because I don’t like him, but because he is a difficult person to get to know and doesn’t make casual friends. He doesn’t know me from Adam. I’ve seen Richard talk many times, and to him I am just another fanboy who comes up afterwards and grabs his hand and says ‘Oh my God, you’re such an inspiration.’”

### **Softcraft and Minix**

Allison’s first experience of Linux came when he installed Softcraft, an obscure and long forgotten distribution that “came on a bunch of floppies from a local consulting company. Softcraft was one of the first consulting companies I can remember that was run by a woman. We hired them to do some work at Vantive, and they had their own distro. I’ve still got a Softcraft Linux mug in my house.”

This was “somewhere around 1994. . . before version 1.0”. Linux had been around for a year or two, but didn’t have a working TCP stack, and until it had a TCP stack “it was a waste of space for us. We did the early Samba development on SunOS or Solaris. Back in those days there were many UNIX – ‘pay me 100 bucks and you’ll have a UNIX-like or UNIX compatible operating system’ – like systems, and Linux was just one among many, and they were all pretty much the same, and I didn’t have the time to fix anything.”

“I had played with Minix when I was younger, when I had the time to fix SCSI drivers. But by the time Linux came out I had a job. Many of the people who later became top kernel hackers were younger than me, and had the time to work on it. When I had the time there wasn’t a Linux. There was only Minix, so I cut my teeth working on Minix. . .”

My opportunity was Samba. I happily stole time from my employer at the time, Vantive, to work on Samba. Stealing is too strong a word for it, because I told them I was doing it, and they were happy because they were having to use PC-NFS [2], and PC-NFS sucked. So when I came to them and said, ‘We have this new file server that I’ve been working on. Why don’t you try it?’ – they were great. Spending ten hours a week on Samba wasn’t what they were paying me for, but they were happy with what I was doing.”

### **Meeting Linus**

“Linus gave a very early talk for the Silicon Valley Linux User Group. When he had just moved to America he gave a talk at this terrible restaurant where we used to meet. It was a horrible place and the food was abominable, and I didn’t go, and missed his big entrance. A few years later when Linux was starting to become quite important he gave another talk about the kernel to the Silicon Valley Linux User Group, and this time it was held at Cisco, and I really wanted to go, but it was the week that the very first book on Samba came out. . .”

“Computer Literacy had a great bookstore on First Street, San Jose, and I said I would give a talk on Samba. It was one of the first talks I ever gave, and the bastards scheduled me exactly the same night as Linus. And I thought ‘I don’t wanna do this. We’ll do it and no-one will turn up.’ So I turned up and there were 50 or so people, and I said:

‘Look, are you sure you don’t want to go down the road to Cisco and see Linus instead?’, and they said, ‘No, we’re here to learn about Samba.’

So I was stuck giving the talk and missed Linus’s big entry.”

“But I got to know him anyway, because he was friends with Larry Augustine, the head of VA Linux, and I’d meet him at parties and stuff. That was when I was going through my black leather phase.

At that time nobody was shipping Linux, and we had really big dreams for VA Linux. I remember saying:

‘If Dell won’t ship Linux, VA Linux will just have to become another Dell.’”

“At the same time I met a guy called Mike Masterson who worked for one of the big consulting companies and was getting into Linux. Mike decided to have a ‘Future of Linux’ talk. I still have the advertising he did for it, and he got someone from Red Hat to speak, and Linus, and asked me to be the warm-up man.”

“This was the famous event in 97 or 98 when Oracle announced that they were going to be porting to Linux. So I turned up for the talk, and arrived at the same time as Linus. They were queueing around the block to get in, and I thought ‘Wow’. Linus wasn’t well known in the Valley at the time. His name was known but his face wasn’t. So we walked in together through the speakers’ entrance, and Mike Masterson grabbed Linus and said:

‘Hey, where you going? You can’t go in there’,

and I had to tell him, ‘Mike. He’s the reason everybody’s here.’”

### **Days that used to be**

The world of Linux has been transformed during the last twenty years. In the computing industry, on the server and on embedded devices Linux has become the norm. “The people I used to call up and say: ‘This Linux thing, you should get into it. It’s really interesting. . .’. I run across them now and they tell me about their Linux credentials, and how long they’ve been in the industry working on Linux, and I think:

‘I remember you when you were working on QNX. Who are you trying to kid?’”

“Linux is now the embedded industry. It’s what you use. It’s become the establishment, like a rock band that makes it big, and starts demanding payments for its downloads. Linux has become like Metallica. . .”

“But the best way to see how Linux has changed is to think of the parties. LinuxWorld used to be an alternative event, the Woodstock of the computing world. You went to LinuxWorld, and everybody got boozed up and had a good time. The stories I could tell you. . . A lot of people had a lot of good times. LinuxWorld was the rock event of the technology world, and then slowly but surely more and more suits started turning up, and by the end of it me and a few other guys were sat there not wearing suits in a sea of PR people.”

“The last one I went to was a sad affair just after the meltdown. I remember us looking at each other and wondering what the \$\*\*? are we doing here? Nothing sadder than a bunch of old hippies in an industry that has been taken over by PR people. . .”

“The Linux Foundation event is where we all hang out now. That’s where we all ended up going. So the Linux Foundation is like a miniature version of how LinuxWorld was in the old days, but without all the booze, or rather with as much booze as we can handle these days. Linus still turns up there, and so do a lot of the other guys. But the young turks all have job titles like software development manager, and are all terribly serious.”

### **All bugs are shallow**

“The art of writing software has moved on. It isn’t materially different writing open source or proprietary software code. It really isn’t. Simply because you used to get away with absolute liberties.”

“People were sloppy. Code was sloppy, and it can’t be that any more. Again, this is the same for proprietary software. If you’re writing something simple just to test something out, you can throw it together with a minimum of fuss. But as soon as you have to write something that people depend on – and Samba is in that situation, there is a massive amount of work and infrastructure that has to be in place.”

“So we have a regression test suite, and every single commit has to pass a regression test or the code is failed. The goal is: if you want to write a new feature, you write the test for it first, and then you write the code and make sure it passes the test.”

“But I am still a believer in free and open source software. You can create proprietary software that is as good as open source software. It’s just a lot harder. The pressures to ship proprietary software are much greater, and it’s harder to find the time for the care and effort you need. Proprietary developers have begun to understand that as well. The goal is for all software to get better, because these days we depend on software in ways we didn’t used to.”

“I still maintain that it’s easier for code written as open source to hit a higher level of quality with less effort than it is to do with proprietary software, simply because you can’t sweep things under the rug, and you can’t hide things when the code is open source.”

“There have been cases where bugs have persisted in open source code for many years, and it’s been said ‘doesn’t that disprove the idea that many eyes make all bugs shallow’, but that isn’t really the case. It just shows that the code wasn’t executed very often, so no-one took a look at it. The stuff that people depend on gets looked at all the time.”

### **Where we are now**

Circumstances for Samba were transformed by the 17 Sept 2007 decision of the European Court which obliged Microsoft to publish the protocols used by Windows clients and servers under “reasonable” and “non-discriminatory” terms.

“Since MS released the documentation a bunch of new competitors have sprung up. Now it’s all documented, how hard can it be to make an SMB server? Some have tried to make a business out of it. Apple has done their own, and we’ve had bug reports because their server doesn’t implement some of the older calls, and isn’t as complete.”

“There’s more competition but it helps us to raise our game, and this is what we’ve been doing for 20 years now. The people who used to work on the SMB server at Microsoft have taken their money and their shares, and they’re all retired now and long gone. But Tridge and I and Volker are all still here working on it. So we have history with this stuff that few others do, and we do more than anyone else does.”

“No-one else, other than Microsoft, does a print server. No-one else does an Active Directory server, and Samba 4 when it’s finished will be a complete replacement for Microsoft. It’s being shipped but doesn’t do trusted domains yet and doesn’t do forests.”

“The selling feature of the latest release is SMB2, the new protocol that’s been shipping since Vista. It has lots of new and tricky features. Devising a new protocol isn’t an easy thing to get right and there are some new and tricky things that are challenging. People say we’re an old project but Tridge did some statistics on this, and our rate of change in Samba is greater than the rate of change in the Linux kernel.”

### **We won and we didn’t notice**

When Allison began to work on Samba he saw it as “a file server for me, because that’s the whole point. It still is a file server for me. I watch all my movies using Samba. I have a Sage TV box and it mounts a Samba server off my Ubuntu box, and I always run the latest version, so if that doesn’t work that is a showstopper bug. My home system has to work...”

In the beginning Linux made its mark as a file server, thanks to Samba, or as a web server, thanks to Apache, but “Samba is no longer a sexy and glamorous project. It’s fading into the background and has become part of the plumbing.” Where once Samba and Apache sold Linux to the world they are now just part of the plumbing. “But that’s OK, plumbers make good money.”

“Samba is infrastructure and essential, because without it a lot of things wouldn’t work. I don’t know where it’s used, and I don’t have to care. I’m paid to help people use it, and one of the testaments to

our software is I don't get many people asking how to use it. They just use it. The people who do get involved are the people who are pushing it to do weird things..."

"In twenty years everything has become Linux. I have a music system that is entirely Linux-based. I have a video streaming and TV watching system that is entirely Linux-based. I wouldn't be surprised if every flat screen television I have is also Linux-based. The laptop and the desktop my wife uses to do her online banking and email is Linux. My phone is Linux, and my son's tablet PC is Linux..."

"We won and we didn't notice."

[1] <http://www.rxn.com/services/faq/smb/samba.history.txt>

[2] <http://publib.boulder.ibm.com/infocenter/pseries/v5r3/index.jsp?topic=/com.ibm.aix.commadm/doc/commadmdita/nfs.pcnfs.intro.htm>

 Reprinted by kind permission from **The H**, specialists in open source news, features and more: <http://www.h-online.com/>.

---

## The Amazon Kindle

*Martin Houston*

Amazon brought out a new Kindle with a price of just £89.

When new things come out you expect to have to wait a bit of a wait a bit for them to reach the shops.

I was thus quite surprised when the local Tesco had a plentiful supply.

I used to be an avid reader of fiction when I was younger but have not had much time to do so recently. In my current job I drive to work so don't have the reading time on public transport that may commuters do either.

Until now I have avoided the Kindle because I do not like the way that DRM is used to control the paid for content. However I was tempted to put a Kindle into the basket with the rest of my shopping. I knew it was another Linux using system to add to my collection and I thought it would be interesting to own a digital paper device. The fact that the Kindle is digital paper rather than a LCD screen make it a fundamentally different beast to a 'fondleslab'.

My first surprise was that when I got the plain brown sealed box home I found that the Tesco staff had given me the slightly more expensive older model Kindle with the keyboard.

I am not blaming Tesco as the outside of the box gives absolutely no clue as to which Kindle is inside so how were they to know? The differences are slight, the bigger Kindle is slightly heavier but has the advantages of more battery life and memory (so it can hold more books at once). It has also been out long enough that I was able to pick up a rubber protective sleeve and a screen protector for it in Poundland!

I knew the Kindle runs Linux but I was rather disturbed to find that the packaging, the getting started guide and the Kindle itself are all completely silent on this fact! There is a small Java logo on the getting started guide but that is the only clue as to the technological basis of what I have in my hand. This is not a good example for a company that wants you to respect the rights of book authors by imposing DRM on you.

The main job of a Kindle is of course as an Ebook reader. Ebooks have flexibility in the way they look and there is a reasonably good text to speech facility 'if the publisher permits'. Paid for DRM protected ebooks are of course only available from Amazon. Non DRM protected ebooks are also available from other sources, including Project Gutenberg.

The Kindle can also display PDFs – these are a bit more rigid in that text can be zoomed but does not re-flow so you will have to end up panning a page backwards and forwards unless you have eyes good enough to read really tiny text! Even so the epaper display is so wonderfully clear it is the best way to read lengthy PDFs.

The fact that the Kindle is digital paper not a power guzzling screen makes it a pretty good peripheral for your computer. If you can turn documentation into PDF or better still the proper ebook format you will have documentation you can refer to side by side with what is on the screen.

I think this is how my Kindle is going to get used the most.

The Kindle also has a reasonable web browser that is certainly good enough to read Google mail. The battery should last between one and two WEEKS between charges. This is because when the epaper screen has been drawn the Kindle can mostly just go to sleep until the next page button is pressed. This is possible because maintaining the epaper screen uses no power. Connecting the Kindle to your PC via the USB cable will also charge it so provided that you move content into the Kindle that way regularly you will probably never have to worry about charging it.

The Kindle works just like a USB memory stick so in a way could be considered as a memory stick that you can read the contents of without a computer. It is just a shame that Amazon did not consider offering a memory card slot so that the carrying capacity could not be increased. There is no special software needed to copy data to and from it so users of Linux systems are not in any way disadvantaged. Contrast this with the headache of owning just Linux computers and an iPad!

Being able to browse the Internet through Wifi is also a very handy feature if you are in a situation where you do not know when your next battery charge is going to be possible.

I would guess that using a smartphone for WiFi tethering from your Kindle is going to be a lot kinder on the battery than using the browser locally.

I guess the huge success of the Kindle has brought down the production costs of epaper. What I would really like to see now is A3 screen sized monitors in epaper. These could be used as one head in a twin head display with a conventional LCD screen.

This would allow one screen to be fully daylight visible and the other optionally powered up for videos and anything else that needs fast screen updates.

---

## Thoughts on Linux User Groups

*Martin Houston*

In the 10th November issue of Micromart there was a letter from a 68 year old lady, Lynne complaining that Derby Linux User group seems to be moribund, with no new members and nothing new on their web site since 2010!

She then ‘hit the nail on the head’ with some of the things that a LUG should be doing, such as pressuring the City Council Education department as to why they were not running any courses teaching Linux, supplying Linux CDs to local schools and colleges, going on local radio to explain to people the advantages of Linux etc.

There are some very active user groups but too many tend to fall into a routine where a few people who use Linux in their usually hectic day jobs just meet socially. There is nothing wrong with this as such but it does little to help people and dispel the myth that Linux is just for cranks and nerds.

If nobody that is motivated to go to the group has any time on their hands above that attendance is it any wonder that the projects and outreach that should be happening don’t actually happen?

What is needed is a healthy mixture of both experienced Linux users and enthusiasts with spare time on their hands who are willing to learn from and become an ‘effort amplifier’ for the people who can

---

provide the core knowledge as to why the public needs to be told about Linux and Free Software in general.

The Occupy Wall Street movement has much to teach us in this respect. I am not suggesting that LUGs need to camp on their local town hall steps to end Local Government waste on those Microsoft licences! However I have visited the OccupyLSX camp at St Pauls on a couple of occasions and found that the way they are organised into working parties of volunteers very effective. Everything from cooking to teaching to security to sanitation is done by 'working groups'. Having people willing to teach new skills 'in action' is giving everyone a great sense of self worth and purpose. I do not know if the Occupy movement will still be going strong 'on the ground' by the time you read this – you can check at <http://occupylsx.org>, but even if the campers have packed up the lesson is that being proactive about getting our message out is still a valid one.

If you earn a living from using Linux or other Open Source or Free Software then SHAME ON YOU if you do not give up a little of your free time in return and find (or start) a LUG. At least join FLOSSUK if you happen to be reading this but are not yet a member.

If you don't yet know much about Free Software Movement but would like to be part of its capability to help change our world then do the same. Let me know if you are not made welcome!

Hopefully that way we can have LUGs achieve critical mass of enough people both with the knowledge of what to and the time to put in the hours to make it happen.

LUGs should be feeding plenty of easy to digest information about why free software is the better way for humanity to use technology into their local media.

Cash strapped local councils should be beating a path to the door of their local LUG for help in getting and end to the hefty bills to the likes of Microsoft and Oracle that we have to pay out of our taxes.

My own LUG is ChelmerLug – <http://www.chelmerlug.org> – if you are a Linux user in the Chelmsford, Essex area you are welcome.

Southend has another well established user group – <http://www.soslug.org>. New members are most welcome at both.

Back on the subject of OccupyLSX, if you need an understanding as to why things are SO wrong that thousands of people are moved to leave their warm homes and make a public nuisance of themselves then a good resource is "The Crash Course": <http://www.chrismartenson.com/crashcourse>. You can either just watch the series of videos on YouTube or there is a DVD iso image that you may download and duplicate as much as you want (even for a fee which you get to keep – capped at \$10 or equivalent). The course takes about 3½ hours. It is US based and was produced in 2009. The fact that a few years have passed since it was made make it all the more disturbing as we can see the crisis predicted actually coming to pass day by day now.

---

## The Occupy Movement

*Martin Houstin*

GO TO WORK – FOLLOW FASHION – WATCH TV – SPEND MONEY – LOOK HAPPY – ACT NORMAL. REPEAT AFTER ME – I AM FREE.

One of the many slogans that have come out of the Occupy movement that is sweeping the world. The situation with the Economy, Energy and the Environment is now getting so serious that increasing numbers of ordinary people have got 'the wake up call' and are now going beyond just taking one day a time with their heads down 'acting normal'.

I am one of them. We have been sleepwalking for too long because our leaders keep downplaying the seriousness of the situation.

What they thought would just be the usual short lived demo has been gathering momentum for months now. On the weekend of 19th/20th November representatives of the now thirty occupations going on in the UK alone gathered in London to meet and plan. New occupations are springing up all the time all over the world including ones that the authorities had thought they had 'dealt with' such as Oakland.

How do we know this? The internet is the power behind this movement as it was behind the 'Arab Spring' with the ongoing toppling of several dictators.

Richard Stallman of the Free Software foundation came to speak at and encourage the OccupyLSX camp on November 5th. The Free Software Movement he started is coming up for 30 years old. A lot of the ideas that it started that sharing was a more intellegent way to go than just trying to claim ownership of everything. The free software economy that we now have is a testament to there being a better way to do things that movements like OccupyWS and Anonymous are now needing to use in the physical world.

Of course the free software community has an important role to play in helping this. If you want to help find the nearest Occupy group to you and ask to join their 'Tech Team'.

Streaming what is happening to the Internet is the best protection against violent supression of peoples rights to protest. Also our IT skills could be put to good use in turning plans into day to day reality.

Although it is clear from our viewpoint that the current movements had a basis in the free software movement many of the people now involved, not being geeks, are still suffering from having to use Windows.

Getting involved in the larger Occupy movement will also enable us to reach large numbers of people and give them 'something to take home' in that it makes the best use of the computers available and removes the fear of the unknown that people may have about Linux and other free and open source software. With spaces for teaching being made available like "The Ideas Bank" in London there is a golden opportunity to spread the free software message. There is also a specific need of people with the know-how to stream video so that 'Citizen Journalists' can do their job.

The two ways this seems to be done at present is either livestream or ustream services which are both flash based and are commercial operations. What is needed is a fully free software alternative to be available too that can be distributed.

Anyone feel that they want to help?

---

## **Learning Perl**

**Randal L Schwartz, brian d foy and Tom Phoenix**

**O'Reilly Media**

**ISBN: 978-1-4493-0358-7**

**390pp.**

**£ 30.99**

**Published: July 2011**

**reviewed by Paul Waring**

I originally learnt Perl using the 5th Edition of this text, so I had high expectations for this revised edition. Refreshingly for a printed book, the content is up to date, covering Perl 5.14 topics such as improved Unicode support. In fact, the text is so far advanced that you may find that your distribution's stable Perl package has not yet caught up – even the latest version of Ubuntu only had 5.12 at the time of writing.

The first few chapters cover the basics of Perl, including arrays and subroutines. Unlike some other introductory books, the authors do not waste time here discussing how to switch on syntax highlighting or which editor you should use (vim). I was pleased to see an early mention of the warnings and strict options, which are invaluable when trying to track down bugs and a good habit to get into.

Chapter five introduces basic input/output and file access. Simple error handling with `die` and `warn` is covered – the text could benefit from slightly more detail in this area, but arguably a module should be used for anything more than basic I/O. The whole of chapter six is devoted to hashes, which is sensible given that these are an important data type in Perl but can be difficult for beginners to grasp.

The next three chapters are dedicated to regular expressions, which at first seems too much for a book aimed at programmers who are new to Perl. However, regular expressions can be a difficult topic for anyone who has not come across them before, and the authors do an excellent job of gradually building up from basic matches to complex substitutions. Following these are chapters on advanced control structures (`unless`, `until`, `redo` etc.) and an introduction to Perl modules.

Chapters twelve and thirteen cover file and directory operations, building on the previous input and output work and covering a variety of tests such as checking that a file exists before attempting to read from it. The next chapter covers manipulating and sorting strings – a key part of Perl which perhaps could benefit from being presented a little earlier in the book.

Chapter fifteen covers smart matching, which personally I feel is a bit too advanced for an introductory text. The final chapter covers some advanced Perl techniques, which again are probably beyond what is required in an introduction to Perl, but the error trapping and handling section is likely to be of use in all but the most simple of scripts.

Overall, if you want to learn Perl for the first time, or refresh your memory after an absence from regular coding, this is the book to read. All the basics are covered, sample exercises are provided to check your knowledge and there is a helpful appendix demonstrating where to go next. The text could do with a bit of trimming to reduce the number of footnotes, and some of the chapters would benefit from reordering, but otherwise this is an excellent book and well worth a read.

---

## **The Book of Ruby**

**Huw Collingbourne**

**No Starch Press**

**ISBN: 978-1-59327-294-4**

**373pp.**

**£ 31.49**

**Published: July 2011**

**reviewed by Nick Booker**

“The Book of Ruby” is an excellent guide to the Ruby programming language. The book is a thorough introduction for beginners, yet it uncovers some surprising features and quirks for intermediate Ruby programmers too. The book also serves as a good reference for features you might not use so frequently, with useful in-depth sections for the more inquisitive.

By covering the basics of object-oriented programming in the first couple of chapters, at the same time as strings and numbers, the author provides a solid foundation for the rest of the book even if you’re new to OO concepts.

More advanced string operations, and container data structures are then explained, before an extensive treatment of loops, iterators and conditional statements.

Chapters 7 to 9 build on the OOP basics learned earlier by talking in more depth about how methods operate and return values, and how to use exceptions to good effect. In my opinion the “block” is Ruby’s most distinguishing feature, and Chapter 10 provides the coverage it deserves.

Symbols and mixins are discussed in detail, and then the really meaty things like file handling, data persistence, regular expressions and threading features are discussed thoroughly.

The book finishes off with an introduction to debugging and testing Ruby programs, and a brief introduction to Ruby on Rails. Both Rails 2 and 3 are covered, with differences between them pointed out clearly by notes in the left margin.

Throughout the book, the author guides you through what he calls the “pitfalls” and “choppy waters” of certain parts of Ruby. In particular the author is careful to point out where things work differently in Ruby 1.8 and 1.9.

The “Digging Deeper” boxes are useful if you need a concept explaining further, are just interested in how things work underneath, or are an intermediate Ruby programmer using them as a reference, but I don’t think skipping them does any harm if you just want a leg-up into Ruby programming – save them for a second read.

There are downloadable, runnable code examples which are referred to throughout the book. I managed to make sense of the book without them, preferring to get away from the screen for a bit while I read it, however if you’re new to programming or to Ruby I suggest you download and use them.

I think this book is a good guide to Ruby for beginner programmers and programmers branching out into Ruby, while filling gaps in the knowledge and providing a handy reference for intermediate Ruby programmers for some of the language’s less used and well-documented features and quirks.

---

## **HTML5: The Missing Manual**

**Matthew MacDonald**

**O’Reilly Media**

**ISBN: 978-1-4493-0239-9**

**448pp.**

**£ 30.99**

**Published: August 2011**

**reviewed by John Collins**

HTML5 is the latest iteration of HTML, the language used for building web pages. The history of HTML, as the book explains in the early pages, is not one of relentless progress without diversions and blind alleys.

The main benefits of HTML5 are that many common features of websites such as graphics and navigation bars are provided directly in HTML5 without having to resort to endless constructs using `<div>` etc. Also browsers which support HTML5 will automatically support things like audio and video without having to use endless, sometimes proprietary, plugins. Of particular interest to many users is the ability to adapt web pages for mobile devices.

The downside, as ever, is that if your website is to be viewed by everyone in the world all of whom will not be running recent browsers, you will not be able to use all of the features. Hopefully that will be a diminishing number, but the user must be aware of the problem.

This book is aimed at being a “how to” user guide rather than “definition and specification of” reference manual. It actually goes a fair bit further than just discussing HTML5, it talks about corresponding extensions to CSS and JavaScript. These are clearly totally necessary for the user to be able to understand the full picture of what is going on.

What is very good about the book is that at each stage the user is presented with what works and what doesn’t work in various browsers. It also provides ways of working which will give a reasonable

result on browsers which do not offer given features. Each feature is presented with a table of which release of all the browsers support that feature.

I'm not convinced that the appendices, explaining what CSS and JavaScript are, is quite necessary. The reader will be at sea if he/she doesn't know what they are when they pick up the book.

However I am sure that this is a very helpful book and one I'm sure web designers will want to have readily to hand.

---

## **Programming Android**

**Zigurd Mednieks, Laird Dornin, G Blake Meike and Masumi Nakamura**

**O'Reilly Media**

**ISBN: 978-1-4493-8969-7**

**502pp.**

**£ 34.50**

**Published: August 2011**

**reviewed by John Collins**

This book is aimed at helping the user program applications for Android, the currently most popular operating system on smartphones and similar (and the subject of countless lawsuits!).

The coverage is extremely comprehensive, telling you how to install the SDK and the AVD (Android Virtual Device), how to develop and test applications, how to add it to the market and how even to get paid.

The Eclipse IDE is recommended and its use assumed throughout.

Obviously with much of Android being based on Java, you will have to be familiar with Java to do anything serious. The book does give an overview of Java in the early chapters, but you will probably need to learn Java from other texts to do very much with it.

The book is structured in four parts.

Part I, "Tools and Basics" is a general overview of the Android development environment.

Part II, "About the Android Framework" goes into the Android framework in some detail. Initially an example is used to explain how Model/View/Controller applies in the Android Framework. Then notably the Fragment class is explained. This class was introduced with version 3 of the SDK to cater for the requirements of devices with larger screens like tablets and is a major change to the framework. Additional chapters cover graphics and the use of the sqlite database.

Part III, "A Skeleton Application for Android" draws together the work done in the earlier parts to build a "well-behaved application". There then follows a discussion of "content provider" APIs. These APIs allows integration with local data and synchronisation with the Google cloud. An example is developed using the Finch Video Content Provider as a starting point. The final chapter in this section of the book goes on to implement REST data operation.

Part IV, "Advanced Topics" introduces additional APIs for multimedia, mapping, external sensors, near field communication, speech and gestures. There is a chapter on communication, synchronisation and social media. The final chapter briefly covers the Native Developer Kit (NDK).

Some of the code in the book are snippets rather than full working examples. That apart, this book is about as detailed, thorough and comprehensive as it gets. The book is up-to-date with code that applies to tablets and the section on Near Field Communication. I haven't had time to actually get as far as writing an Android application using it, but I intend to do so soon.

If you want to develop for Android you must get this book.

---

## **CSS Pocket Reference**

**Eric A. Meyer**

**O'Reilly Media**

**ISBN: 978-1-4493-9903-0**

**248pp.**

**£ 11.50**

**Published: July 2011**

**reviewed by Lindsay Marshall**

There is really not a lot to say about this book. It is a comprehensive reference for CSS and includes CSS3. It is written by Eric Meyer. Small in size and concise in content, though the price is a little high for such a small book.

If you work with CSS even a little, you almost certainly need to have this on your desk – it is ensconced on mine now: much easier to use than a web search that inevitably ends up at w3schools. The introductory sections summarising the syntax, units and rules are excellent. I only have two niggles. The first is that the book recommends using `::` for pseudo-elements rather than `:` and then consistently fails to do so in any of the examples that I have found. The second is an entirely unreasonable complaint: there is no indication of what the levels of browser support for the various properties are. How could there be without it becoming out of date instantly? I still want it though.

Well worth buying.

---

## **Programming Social Applications**

**Jonathon LeBlanc**

**O'Reilly Media**

**ISBN: 978-1-4493-9491-2**

**544pp.**

**£ 34.50**

**Published: September 2011**

**reviewed by Lindsay Marshall**

This book is all about using various non-proprietary tools to construct what it terms social application, though it is not entirely clear to me what they exactly mean by a social application. I think that, underneath all the social blather, it is about making sure that your users get what the experience they (think they) want without them having to do too much, that is, you connect to the various incarnations of their profile that already exist and use that to tailor what they find. Though why you would want these applications to be “highly viral” as suggested in the blurb for the book, I am not sure.

It's a bit of a beast of a book, quite thick, expensive and with lots of content on things like versions of OAuth, OpenID, Caja, AdSafe etc. More code to read through but it doesn't feel oppressive here for some reason. There are useful walkthroughs for setting up connections and getting data from them, which in my experience is the most tedious part of using any of these kinds of systems. I have to say that I would be tempted to cut out the middleman and just download the code for the ideas in the book from github and just use that without getting to grips with the horrible engine room details underneath (which the book tells you about). That being said there is a lot of useful stuff about the social graph in here and this is a developing area that people are trying to get up to speed on and this provides a comprehensive summary of a lot of ideas and issues that you need to know about. I would certainly recommend it to anyone starting to work on “socialising” their systems.

Of course, what a lot of / most / some people do is just copy the code from the Facebook developer page and then just use FB logins and data...

---

## JavaScript Web Applications

**Alex MacCraw**

**O'Reilly Media**

**ISBN: 978-1-4493-0351-8**

**280pp.**

**£ 26.99**

**Published: August 2011**

**reviewed by Lindsay Marshall**

I think I must just be getting old as I am starting to find books filled with code very hard to get along with. This book sets out to develop an MVC approach to building JavaScript applications, and looks at various ways to approach doing this, and how to make use of some of the new HTML5 features that are appearing in modern browsers. Now, let me state straight away that the content in the book is completely technically sound, and the concepts being developed are important (if a little leaning towards the religion of the pattern), but...

Reading through code examples in books is just a grim experience. This is a book that, unlike the CCS3 quick reference, would make be much better as a set of webpages: the content is high quality and useful but is not really that well suited to a linear presentation. This is partly why I often felt overloaded – the author recommends loads of different services that do very similar things to each other and this just felt a little confusing at times. (But let me stress there are services mentioned which I didn't know about that I am going off to investigate.)

The book has the strapline “jQuery Developer's Guide to Moving State to the Client” which somehow seems to miss the point of the book, though clearly that is what some of it is about. A book full of excellent, useful content, but pricey, and I find it hard to say that you need to buy it. Not one to buy without having first had a look at it.

---

## Head First jQuery

**Ryan Benedetti and Ronan Cranley**

**O'Reilly Media**

**ISBN: 978-1-4493-9321-2**

**544pp.**

**£ 30.99**

**Published: September 2011**

**reviewed by Mike Fowler**

O'Reilly's brain friendly guide to jQuery is a very different way to learn about about the technology. The book is more diagrams and examples than text and what text there is is often annotated as much as the worked examples. This presentation style takes some getting used to but it's claimed that this structure helps your brain to absorb what would otherwise be dull information. Perhaps a better way to describe this book is as a textbook and workbook merged into a single volume.

The technical pace is good for a beginner, starting with an overview of what the document object model is and how jQuery's powerful selector syntax works. Events, animations, AJAX and JSON are all gradually introduced and each chapter works through an example to fully illustrate and teach the concept being covered. The examples take a number of forms, from fill in the blank spaces in code snippets to getting the example written and running on your own machine.

Once you've finished the book I fear it will remain parked on a bookcase until you find someone else who is a jQuery beginner. With its presentation style thumbing through the book for reference is very difficult as you almost need to work through the chapter again to see where the examples are going. I also expect that people who are technically proficient in other languages and even other JavaScript libraries will find this book much to slow paced. This really is a book for complete beginners and for them it is a perfect introduction.

---

## **Perl Pocket Reference**

**Johan Vormans**

**O'Reilly Media**

**ISBN: 978-1-4493-0370-9**

**104pp.**

**£ 9.99**

**Published: July 2011**

**reviewed by Martin Houston**

As you would guess from the title this is only a small book, 93 pages including the index and a size that will actually fit into a jacket pocket so you can have it with you.

The book has been around for quite a while now, in fact I owned a previous edition myself several years ago, sadly no longer with me as it must have been lent but never returned at some point!

So I was pleased to get a copy of the new version I am reviewing. This is now the 5th edition which is new enough to cover a very modern version of Perl – 5.14.

Perl 6 has been in development for quite a while now, and many of the features that it promises tend to get back-ported into new incarnations of the '5 series' so that they can be put to work powering the World Wide Web and the much else besides that Perl is the powerhouse for.

The book is NOT intended to teach Perl (that would be some feat in 93 small pages), its role is as a mind jogging reference to the bits of Perl you have used before but cannot remember off by heart, things like what are the options to `pack` and `unpack` for getting Perl to deal with arbitrary binary data formats, or what the various regular expression modifiers or file tests are. Concise summary of subjects together with heavy indexing make it tool specific for this role.

As Perl is now such an established language much of what the book covers is useful even to legacy versions of the language. However I think the main reason for buying this book is to let you find out if the traditional way you were going to use can maybe be done cleaner with one of the newer Perl language features.

There is no point of a language like Perl evolving if us programmers are too timid to use the new features. Features that are there not because some marketing hack thought they may help sales (Perl being free avoids that conflict of interest), but because fellow programmers have carefully decided that they are useful and labour saving enough to be worth the disruption of changing the language for. Human languages evolve in this way all the time so why should their computer equivalents not also enjoy that innovation? While reading through the book on several occasions I found myself thinking "Wow can Perl do that now?" One such innovation is the 'smart match' operator (page 35) – the inoffensive looking `~~`. Some of the things smartmatch does are just 'do the right thing' common sense such as being able to deep compare two hashes without cumbersome library calls. The Pocket Reference however gives a full table of what `~~` will do when given operands of different types. One unexpected (to me) one is matching an array or hash against a coderef. The truth of such a thing is the truth of the coderef function applied to each array element or each hash key.

This is a bit like the traditional `grep` operator, which iterates a function over a list returning only those elements for which that function is true. If all we wanted to know is if the function is true for

any element then the new approach avoids having to collect would could be a large amount of data then to just to see if `scalar()` of that data is greater than zero which is the truth value that the smartmatch will deliver directly. New features are generally there to enable code that is faster, easier to maintain and hopefully both at once.

If this book piques your interest in using new features of Perl in ways that make your code both faster and easier to use then it is well worth the meagre price. Online documentation is all very well if you know what you are looking for. The summaries in this book, if they don't answer what you need to know directly will at least give you the hook to find it out online in short order.

Do I need to mention that FLOSSUK members get a generous discount on all O'Reilly titles?

---

## Contributors

**Nick Booker** is a Ruby and Python programmer, and is the technical director of Sunfield Technology Ltd. He writes mostly web applications, and dabbles in photography, swimming and running in his spare time.

**John Collins** is a former UKUUG Council member and runs Xi Software Ltd which specialises in System Management tools for UNIX and Linux systems. Xi Software is based in Welwyn Garden City and has been in existence for about 25 years.

**Mike Fowler** is a Senior Software Engineer working in the Midlands. 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.

In addition to writing articles on Free and Open Source Software for a variety of prestigious publications, **Richard Hillesley** splits his time between his home in Totnes and a communal permaculture project in the West Country...

**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.

**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.

**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 SUSE supporting SUSE Linux Enterprise Server for major customers 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

Ian Norton  
Council member  
Rochdale

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](mailto:office@ukuug.org)

Roger Whittaker  
Newsletter Editor  
London

Alain Williams  
UKUUG System Administrator  
Watford

Sam Smith  
Events and Website  
Manchester