Contents

From the Secretariat

Jane Morrison

The AGM this year will be held on Wednesday 23rd September at the Ambassadors Hotel, 12 Upper Woburn Place, London WC1H 0HX, starting at 6:15 p.m. full details including the Agenda etc. were sent to all paid members at the end of August.

The date was originally announced as Thursday 24th September but it has been changed and is now confirmed as Wednesday 23rd.

We hope you will be able to attend. Full details can also be found on our web site.

Our annual Spring event (Workshops & Conference) will be held on 15th, 16th & 17th March 2016.

This is the first time we have organised the event in London since 2009.

The full list of speakers and on-line booking for this event should be available early in December.

The next Newsletter will be the December issue and the copy date is 16th November. As usual any articles, letters etc. can be sent for inclusion to: `newsletter@ukuug.org`.

Chairman's Report

Kimball Johnson

Sponsor Members

I would like to take the opportunity to thank our two silver sponsors, 2ndQuadrant and SUSE. Their generous support helps us to put on the free events throughout the year.

Events

Thanks to all those that attended OpenTech in June, it was once again a successful well attended event. My thanks go out to Sam Smith for once again organising such a great event.

The first FLOSS UK Dynamic Languages conference was held in Manchester a week later. Thanks to those who spoke and attended. We hope to run it again, details will come early next year.

Barcamp Birmingham returned once again, thanks once more to those who attended, if you would like to see the talks, they are all on the FLOSS UK YouTube page: `http://ow.ly/RnFcA`

Looking forward, in November we are hosting the annual LDAPCon conference, this year it is being held in Edinburgh. Booking will open shortly, see `http://ldapcon.org/` for more infomation.

We are always looking for ideas for tutorials to run for our members, so if you have any topic you wish to hear about, please contact me at `kimball.johnson@flossuk.org` or Jane on `office@flossuk.org`.

Support for local user groups

The budget is still available for FLOSS UK to assist local user groups by helping them to obtain speakers for their events and assist with travel expenses. In addition we have a small budget for assisting with projects that would benefit the Free Software or Free Hardware communities in some way.

If you have an idea and wish some support, please contact Jane on `office@flossuk.org` and it will be discussed by Council.

From the Editor

Paul Waring

Council (and myself with my Editor's hat on) have recently had a discussion regarding the future of the FLOSS UK newsletter, which at the moment is produced quarterly in black and white. This format has been easy for me to work with – thanks in no small part to the build system I inherited from Roger Whittaker – but it has restricted the type of content, particularly when it comes to photographs, diagrams and colour.

In order to enable FLOSS UK to produce a newsletter with a wider range of content, including full coverage of the annual Spring conference, we agreed that the best way forward would be to move to a different newsletter format, which will be produced in full colour. Due to the expense and time involved, the newsletter will be produced annually, with a distribution date 4-6 weeks after the Spring conference (i.e. late April/early May).

Council and I are aware that an annual newsletter may mean that book reviews and event reports are delayed, and in order to address this they will be published on the FLOSS UK website throughout the year, with a selection reprinted in the newsletter. As well as speeding up the process, this will also allow us to be a bit more flexible about copy dates, which have been a problem in the past when books have been delayed in the post.

If you have any ideas or comments on the proposed newsletter changes, please do get in touch via email: `newsletter@flossuk.org`.

LDAPCon 2015

The fifth International Conference on LDAP and Directory Services will be held in the UK at the University of Edinburgh School of Informatics Forum.

Tutorials: Wednesday 11th November 2015

Conference: Thursday 12th & Friday 13th November 2015

The tutorial day has one track designed for LDAP newbies and one for experienced practitioners. There will be a strong emphasis on practical work for both tracks.

The conference has a full programme with 22 peer-reviewed papers across two days. Topics include service design, LDAP schema, protocol enhancements, server technology and client programming.

There will be a poster display area and ample opportunities to meet other LDAP specialists including the central figures in the development teams of many well-known server products.

The conference dinner on 12th November will be held in a spectacular city-centre venue, providing more networking opportunities.

Full details are on the website: `http://ldapcon.org/`

Booking opens in early September, and early-bird prices are available until 1st October. There are also opportunities for commercial sponsorship that will be of interest to companies working in related areas.

Book early for the best rates!

Dynamic Languages Conference Review

June saw the first Dynamic Languages conference to be run by FLOSS UK, the first to be held in England, organised in conjunction with Shadowcat Systems Limited. The first Dynamic Languages conference was held in 2012 in Edinburgh to coincide with the inaugural Turing Festival Event.

The conference is a cross-discipline event with a loose umbrella of open source dynamic languages; by which we mean high-level computer languages that perform most of their behaviours during runtime as opposed to static languages that do this during compile.

The conference hopes to bring together all the languages in an open forum to discuss and present the manner in which they approach and solve tasks at the cutting edge of development technology.

The event is held on one day, this year it was Saturday 20th June, in central Manchester. Our broad theme this year was to allow a general open-forum for discussions around dynamic languages with the idea of testing the format and approach with a particular reference to web technologies and system administration.

We were fortunate to have two keynote speakers, Lorna Mitchell and Gianni Ciolli, who bookended the events of the day. The other speakers were James Laver, David Jones and Matt S. Trout. The day was rounded with a series of lightning talks from the audience with several people creating new talks on the day of this event.

Lorna kicked off the day with a passionate talk about 'Your Open Source Journey'. This was a talk specially commissioned for the Dynamic Languages conference and it was a great initial talk to start a new lease of life for the conference. Lorna spoke passionately about being involved in open source, the challenges and the path to follow.

David Jones gave us an interesting talk on the language Go. His proposition was to ask the question of whether Go is a Dynamic Language. Well being statically typed with most behaviours controlled at compile time, it isn't. But that was part of the charm of the talk as it was in fact an introduction to Go and how it benefits from not fitting the dynamic language model. In many ways another perfect DL talk.

James Laver gave us a developers at the coalface talk about the use of Agile and Agile Processes in a production environment. This was a talk as much about how to treat developers as about processes, it also covered some of the best practices for running teams and treating your workforce with respect.

In his usual staccato style Matt S. Trout gave an impassioned monologue about the use of object orientation without any of the boilerplate in Perl. As always Matt delved into real code examples and applied usage to punctuate his discussion and highlight the points. He also took a wry look at the notions that Perl doesn't have an OO system by a comparative look at OO in other Dynamic Languages.

Our final keynote was delivered by the enigmatic, and charming, Dr Gianni Ciolli. Gianni gave a custom talk on the use of Postgres with multiple languages and how we can utilise the same processes and libraries with little change. As informed as he is entertaining, Gianni sold well the need to evaluate Postgres as a first choice for database when beginning any project.

To round the day off we had a round of lightning talks. On a personal level I love the lightning talks more than any other part of a day, giving someone just five minutes to shine or plummet to a fiery demise is part of the whole joy in attending any event. All of our lightning talk speakers did a fine job but I would like to make particular note of two people.

Tom Bloor prepared a lightning talk specially for the day. It was the very first time Tom had stood in front of a group of people and given a solo tech talk that he had devised and written. He did a great job and the start of a much longer career in technical presentations.

Paul Waring gave the audience a fun tour of why all Dynamic Languages are awful except for the one that he prefers (PHP). It was light-hearted, provocative and above all fun and placed the exact correct spin on a lightning talk.

We learned several lessons in holding the event, it wasn't as much of a success as we would have hoped with a low speaker count and limited audience. I don't want to dwell on the reasons for this. As organisers and promoters we have discussed and evaluated what happened and how we overcome that. We took everything as a lesson and in doing so, we learned.

None of this, though, should detract from the effort of the volunteers, speakers and audience. A fledgling event is bound to have a number of teething troubles and this has been valuable in many ways.

The conference will return to the FLOSS UK schedule in 2016 in the Autumn.

OpenTech Review

Quentin Wright

OpenTech 2015 held on 13th June 2015 at ULU Student Central

OpenTech has been a more-or-less annual event for UKUUG/FLOSS UK since 2005. This year it was held again at what used to be known as the University of London Union on Malet Street in London but has been re-styled as 'Student Central'.

The event is loosely described as a 'mix of technology, experience and everything else' and this year's one-day conference was no exception. The event was sponsored by the Open Data Institute and they bring their own flavour to the proceedings. The ODI quite rightly take the view that a proper data infrastructure is as important as the transportation infrastructure. As well as the ODI, there were speakers linked to the Open Rights Group, which incidentally was founded at the first OpenTech 10 years ago. Broad themes that were covered by the conference included the follow on from Snowden, Open Data in all its flavours with talks about Rail Data, Healthcare Data, Government publishing and web sites, analysing data and Democracy and the Election. There were also more technical talks about Network Addressing, Blockchains and gitfs which is a file system built in Python. There were a few wild cards too with a talk entitled 'The Fantasy of Frictionless Cities' and another 'Digital Toilets for your Convenience'.

The conference was run as three parallel streams which makes it difficult to give a full overview here. Much of the content can be found on the OpenTech web site either in full

video, or else audio along with links to the presenters' slides and other supporting documentation.

I spent most of my time in Session 1 and made notes about some of the presentations.

Sarah Brown kicked off one of the sessions talking about 'What it's like being a female on the internet'. In summary, it can have its unpleasant sides. Sarah encouraged all to persevere and carry on tweeting.

Vinay Gupta gave a talk about marketing entitled 'Selling ideas'. He gave examples from the Hexayert Project which has created buildings which can be rapidly assembled from standard building panels and are superior to tents in disaster situations.

There then followed a series of short talks from Privacy International introduced under the broad title of 'The Third Year of Snowdenia'. I found these to be really impressive as they were comprehensive in content and very topical.

The session started off with Caroline Sinclair, 'A Lawyer who cares about technology'. She said that laws aren't up to shape with changing technology. So for example in the past in order to follow a person's movements 24 hours a day, a team of at least 8 people would be needed. With the invention of GPS and mobile phone tracking it's possible to achieve the surveillance at a much smaller cost. Another aspect of mass surveillance used to be a concentration on the content of communication and the law has reflected this. Now as well as enabling agencies to track what you are communicating, technology provides information about who you're communicating with, who you know and when that communication is taking place. In other words the metadata. Thanks perhaps to the increasing use of encryption the authorities are building haystacks of metadata which they can then plumb at their leisure and as changing circumstances dictate. A third kind of government hacking is through what is known as 'equipment interference' or 'Computer Network Exploitation'. This came to attention when it became apparent that GCHQ had carried out an attack on Belgacom. Thanks to the revelations from Snowden an overhaul of legislation is taking place in many countries throughout the world including the UK.

This talk was followed by a presentation from Ian Brown from the Oxford Internet Institute started punningly with a slide bearing the legend 'The Grim Reaper'. He continued to give an entertaining and thoroughly detailed presentation of the current state of the RIPA - the Regulation of Investigatory Powers Act. This is particularly topical as David Anderson Q.C had produced a critical report on RIPA on June 11th entitled 'A Question of Trust'. Among other things he described the act as being 'incomprehensible' and 'absurdly complicated'. Sir Anthony May had said that it 'took a year to understand'. We look forward to a draft report in the Autumn.

In all about 250 people attended the conference and as well as the formal presentations there were opportunities to network with like-minded people. One of the advantages of holding an event in London is that there were contingents who came down from Birmingham, Manchester and even Edinburgh just for the day.

Congratulations must be made to Sam Smith and his team for putting together yet another successful event and we look forward to doing the same again next year!

BBC returns to the hardware wars

Les Pounder

We have seen many different hardware platforms in recent years but the market has been dominated by two names, Arduino and Raspberry Pi. These two names have introduced the wonders of coding and hacking to users of varying levels of ability, thanks largely to their ease of use and their open communities. From children to artists, parents to painters these boards have ultimately changed the face of computing across the world. The biggest impact of all has been made to teaching the Computing curriculum in the UK. And this impact has been made by the Raspberry Pi, which started life as a board for around ten thousand university students in 2012 but has now grown to over five million users across the globe. The Raspberry Pi education team has grown and now produces educational resources and provides outreach activities across the globe. The Pi is seen as the THE board for children to learn, hack and investigate but earlier this year the BBC announced that they were working on a new piece of hardware called Micro:bit which has been seen as a potential rival to the Raspberry Pi.

Micro:bit is a small micro controller powered board that can be programmed using a block based language called Microsoft Blocks, very similar to Blockly, or using Touch Develop, a typed language similar to C++ but geared towards children. Micro:bit is a BBC project set to distribute boards to every 11 year old child who enters secondary education in September 2015. It is a low cost platform that is akin to the Arduino range of devices so requires an external computer or tablet to code the projects which are then uploaded to the Micro:bit.

The BBC have partnered with 29 organisations to deliver Micro:bit [1] and these include names such as ARM, Barclays and Lancaster University. All of these partners along with BBC Learning have been working behind the scenes to test and refine the Micro:bit, but at this time there is some debate as to when it will be made available to schools. Originally the board was to be with teachers in time for the start of the new term in September but this deadline has now slipped to the end of October. Teachers across the UK are eager to get their hands on the board but at present only a few have been able to for testing purposes. The issue that now faces teachers is 'How can Micro:bit be integrated into Computer Science?' and herein lies the problem. By introducing another board we add another level of complexity for our teachers. With the recent changes to the Computing curriculum teachers are working even harder to learn how they can deliver the content and there are quite a number of teachers who are still trying to get to grips with learning to code and by adding another board and language their workload further increases. A clear example of this has been the rush to learn Python which has gripped the UK teaching community.

The hardware for Micro:bit is as follows:

- Nordic nRF51822 Multi-protocol Bluetooth® 4.0 low energy/2.4GHz RF
- SoC 32-bit ARM Cortex M0 processor (16MHz)
- 16kB RAM
- 256kB Flash
- Bluetooth Low Energy Master/Slave capable
- 25 LED Matrix
- Freescale MMA8652 3-axis Accelerometer
- Freescale MAG3110 3-axis Magnetometer (e-compass)
- Push Button x2

- USB and Edge connector Serial I/O
- 2/3 reconfigurable PWM outputs
- 5 x Banana/Croc-clip connectors
- 6 x Analog In
- 6-17 GPIO (configuration dependent)
- SPI
- i2c
- USB
- Micro B connector
- JST power connector (3v)

Code for Micro:bit is written using an online portal in either language and it is then compiled and downloaded to your computer. Plugging in the Micro:bit using a USB lead will mount the board as a mass storage device and you can then copy the code across before pressing reset for the code to take effect.

Freedom and choice are what FLOSS is built on but can we be sure that Micro:bit will be an open source project? A quote from the BBC media centre [2] states 'The technical specifications for the device will be open sourced, and the partnership plans to collectively develop a not-for-profit company to oversee and drive the Micro:bit legacy.' But what about the software? Will this be open sourced or will it become proprietary forcing children to learn the Microsoft way?

I remain open minded about Micro:bit after having used it for a day at a recent BBC Make It Digital event. The ultimate goal of the project is comparable to that of the Raspberry Pi and the two boards are not rivals, in fact the team behind Micro:bit are well aware that their board is a first step to physical computing and something that children will use before and with the more powerful Raspberry Pi.

The only questions we face now are 'When will the board be available?' and 'What support materials will there be for teachers and parents to follow?' These questions along with the possibility of open sourcing the software will be answered in the coming months.

[1] `http://www.bbc.co.uk/mediacentre/mediapacks/microbit/partners`

[2] `http://ow.ly/RnIAh`

Crafting the InfoSec Playbook
Jeff Bollinger, Brandon Enright, Matthew Valites
O'Reilly Media
ISBN: 978-1-4919-4940-5
276pp.
£ 20.15
Published: May 2015

reviewed by Nigel Barker

While it might not be clear to a British audience, a playbook is a collection of plays: the tactics that an American Football team will use in any particular situation. This book describes putting together an InfoSec playbook for your Computer Security Incident Response Team so that the CSIRT will know how to react to evidence of any attack or intrusion.

Jeff Bollinger is an Incident Response Technical Lead at Cisco and his co-authors both have Cisco connections but the book is vendor neutral. Brandon Enright authored a blog posting on the Cisco Security blog in November 2013 which covers some of the same material as the book but in summary form.

The approach they favour is to build a repository for your security logging and then within your plays to develop searches that generate actionable results for your analysts to work on. To do this requires you knowing what you're trying to protect, what the threats are, how you will detect them and how you will respond.

With this information you can write a play, either based around a particular threat or a particular subset of systems. As well as the search, the play will include guidance on how the analyst should work the incident and space for the analyst to record notes for the next time that play comes round.

The book provides a useful guide to the sort of logs you need to collect and discusses parsing and extraction of metadata. There is an interesting comparison of security log metadata with the Dublin Core library classification system.

I particularly liked the chapter on Advanced Querying which introduces some of the maths behind advanced data searching such as Markov chains, set operations and the false positive paradox (even if you only have a tiny percentage of false positives they will likely outnumber true positives because true positives are so rare).

Like much of the book the ideas are interesting but would take a lot of work to get them to a useful state for your own environment, but perhaps this is the point. The authors are clear that homegrown plays, which make use of local site knowledge as well as threat intelligence, are superior to an approach built around a single vendor product, but do require work to develop and maintain.

The book also has helpful chapters on operations and mitigation including DNS RPZ and BGP blackholing.

For the most part the book is well edited. I only spotted one typo, where a URL is printed twice, but a couple of the diagrams are unclear, which looks likely to be a result of converting from a colour presentation to a black and white book, e.g. referring to green areas and two different data paths having the same grey line style.

The only other book I am aware of in this market space is the Syngress Logging and Log Management book by Anton A Chuvakin, Kevin J Schmidt and Christopher Phillips. The Syngress book is longer, with more emphasis on log management but suffers from poor editing. As a guide for the CSIRT opertions manager I think this book is a helpful introduction.

CoffeeScript, 2nd Edition
Trevor Burnham
Pragmatic Bookshelf
ISBN: 978-1-941222-26-3
150pp.
£ 19.50
Published: February 2015

reviewed by Charlie Harvey

Recent years have seen a trend of building languages that transcompile to JavaScript. Such source-to-source languages treat JavaScript as a sort of virtual machine, a little like Sun's original vision of what the JVM was to be. The trend has been so widely embraced that nowadays no self-respecting hipster web dev would be without such a language – whether that language be CoffeeScript, Google's Dart, Microsoft's TypeScript or functional transcompilers like ClojureScript, Elm or Scala.js.

The book's author Trevor Burnham has managed to squeeze a lot of information into a relatively short book, though there were times when he left me wanting some more detail on the topics he was covering. Like many of the PragProg books, this one takes a tutorial approach, with iterated micro-projects illustrarting the features of the language.

CoffeeScript takes a lot of the inspiration for its syntactic sugar from Ruby, and Rails has shipped with it since version 3.1. There are some Pythonic elements in the language too, not least that CoffeeScript has semanticly significant whitespace, rather than semicolons or begin-end keywords. Controversial!

This and other syntactic elements of CoffeeScript are discussed in the first 4 chapters of the book. Chapter 1 is about installing CoffeeScript and getting your dev environment set up; chapter 2 moves on to functions, scope and context; chapter 3 deals with collections, iteration and destructuring and chapter 4 with classes, prototypes and inheritance.

After this well thought-out basic CoffeeScript primer, the book moves on to the fun part – illustrating how CoffeeScript is used in the wild. Chapter 5 covers building web applications with jQuery, Backbone.js, Grunt and Bower.

In chapter 6, Burnham moves on to building web backends with Node.js and Express.js, rounding out the knowledge required to get a proper application up and running. Finally, in chapter 7 we move on to testing using Intern which is a flexible, standards compliant JavaScript and web testing framework. Intern is a nice piece of testing software and supports source maps.

In those 3 chapters there are certainly a lot of non-CoffeeScript libraries are covered, but the point that I think Burnham is trying to convey is that CoffeeScript plays well with the rest of the JavaScript ecosystem. A useful side effect of covering so many JavaScript libraries is that you get a concise overview into some of the technologies that JavaScript developers are using to build sites these days.

There are a number of smaller examples and one more complex project – a task manager – which is iteratively developed from a simple core to a fully tested application to illustrate how to approach MVC, templating, persistence, testing and so on when working with CoffeeScript.

All in all this is a useful primer and introduction to the world of CoffeeScript and at the same time gives a good overview of JavaScript development practices for programmers who haven't been keeping up with the trends in that world.

Data Science from Scratch
Joel Grus
O'Reilly Media
ISBN: 978-1-4919-0142-7
330pp.
£ 24.15
Published: April 2015

reviewed by Charlie Harvey

Data science is an increasingly important topic these days. And there are, without doubt, huge numbers of books out there already which purport to introduce it, especially books which focus on R or Python. So it is a brave decision to publish yet another.

The author, Joel Grus, explains why this book is different to those already out there in his blog post about it: `http://ow.ly/RnJ9x`. He observes that most books take either a mathematically rigorous but rather dry approach, or they focus on the tools and libraries rather than the conceptual underpinnings of data science. Data Science from Scratch will instead, he hopes, be a third way between those approaches:

- understanding the behavior of the most common tools by working through a solid-but-less-than-textbook-rigorous understanding of the math behind them, and

- implementing simplified versions of them from scratch to understand exactly what it is they're doing.

An ambitious project perhaps, but I felt that Grus succeeded remarkably well.

Data Science from Scratch uses a 'motivating hypothetical' that the reader has just been hired to lead the data science efforts at a pretend company. I am not a huge fan of this style of make believe in technical books. I find myself asking what the hell the HR people were thinking hiring a non-data scientist for the role! However, the story part is not too intrusive.

The book proper starts with an excellent introduction to the Python language. The tutorial covers a large amount of ground quickly. It would be a useful reference in itself. Even as an inveterate Perl programmer, I found the content accessible and the pace well judged. One Python gripe is that the book's examples are all written in Python 2.7 rather than the more current Python 3.4.

Next up is a quick chapter of visualization – one topic that is not covered from scratch – using the matplotlib library. Next follows a chapter introducing linear algebra. All the concepts are explained with Python code. Necessarily this chapter doesn't cover everything in as much depth as one might wish but, as with other chapters a further reading section is provided.

The book moves on with a chapter on basic stats (measures of central tendancy, correlation, causation) then a good treatment of probability followed by a chapter on hypothesis and inference which introduces Bayes. Then gradient descent is covered. The next two chapters deal with obtaining and working with data – reading files, APIs, data munging, and so on. Chapters 11 to 13 focus on machine learning, k-nearest neighbour techniques and naive bayesian spam filtering. Then the next 3 chapters cover regression, simple linear, multiple and logistic. After this, some more ML topics – decision trees, neural networks, clustering, natural language processing, network analysis and recommended systems. Next a chapter on storing data in databases, then a chapter on MapReduce. The last chapter 'Go Forth and do data science' wraps everything up and points to further resources.

As you can see the amount of ground coered is huge, so sometimes there is not as much depth as some may like, however Grus always points to further resources. And his writing style is engaging and accessible. Overall I found this to be one of the best technical books I have read in some time both because of its scope and because it did succeed in treading the line between dry mathematical rigour and handwavy tool-centrism.

Docker: Up & Running
Karl Matthias, Sean P. Kane
O'Reilly Media
ISBN: 978-1-4919-1757-2
224pp.
£ 23.50
Published: June 2015

reviewed by Ian Hopkinson

This last week I have been reading dockerDocker Up & Running by Karl Matthias and Sean P. Kane, a newly published book on Docker - a container technology which is designed to simplify the process of application testing and deployment.

Docker is a very new product, first announced in March 2013, although it is based on older technologies. It has seen rapid uptake by a number of major web-based companies who have open-sourced their tooling for using Docker. We have been using Docker at ScraperWiki for some time, and our most recent projects use it in production. It addresses a common problem for which we have tried a number of technologies in search of a solution.

For a long time I have thought of Docker as providing some sort of cut down virtual machine, from this book I realise this is the wrong mindset - it is better to think of it as a 'process wrapper'. The 'Advanced Topics' chapter of this book explains how this is achieved technically. This makes Docker a much lighter weight, faster proposition than a virtual machine.

Docker is delivered as a single binary containing both client and server components. The client gives you the power to build Docker images and query the server which hosts the running Docker images. The client part of this system will run on Windows, Mac and Linux systems. The server will only run on Linux due to the specific Linux features that Docker utilises in doing its stuff. Mac and Windows users can use boot2docker to run a Docker server, boot2docker uses a minimal Linux virtual machine to run the server which removes some of the performance advantages of Docker but allows you to develop anywhere.

The problem Docker and containerisation are attempting to address is that of capturing the dependencies of an application and delivering them in a convenient package. It allows developers to produce an artefact, the Docker Image, which can be handed over to an operations team for deployment without to and froing to get all the dependencies and system requirements fixed.

Docker can also address the problem of a development team onboarding a new member who needs to get the application up and running on their own system in order to develop it. Previously such problems were addressed with a flotilla of technologies with varying strengths and weaknesses, things like Chef, Puppet, Salt, Juju, virtual machines. Working at ScraperWiki I saw each of these technologies causing some sort of pain. Docker may or may not take all this pain away but it certainly looks promising.

The Docker image is compiled from instructions in a Dockerfile which has directives to pull down a base operating system image from a registry, add files, run commands and set configuration. The 'image' language is probably where my false impression of Docker as virtualisation comes from. Once we have made the Docker image there are commands to deploy and run it on a server, inspect any logging and do debugging of a running container.

Docker is not a 'total' solution, it has nothing to say about triggering builds, or bringing up hardware or managing clusters of servers. At ScraperWiki we've been developing our own systems to do this which is clearly the approach that many others are taking.

Docker Up & Running is pretty good at laying out what it is you should do with Docker, rather than what you can do with Docker. For example the book makes clear that Docker is best suited to hosting applications which have no state. You can copy files into a Docker container to store data but then you'd need to work out how to preserve those files between instances. Docker containers are expected to be volatile - here today gone tomorrow or even here now, gone in a minute. The expectation is that you should preserve state outside of a container using environment variables, Amazon's S3 service or a externally hosted database etc - depending on the size of the data. The material in the 'Advanced Topics' chapter highlights the possible Docker runtime options (and then advises you not to use them unless you have very specific use cases). There are a couple of whole chapters on Docker in production systems.

If my intention was to use Docker 'live and in anger' then I probably wouldn't learn how to do so from this book since the the landscape is changing so fast. I might use it to identify what it is that I should do with Docker, rather than what I can do with Docker. For the application side of ScraperWiki's business the use of Docker is obvious, for the data science side it is not so clear. For our data science work we make heavy use of Python's virtualenv system which captures most of our dependencies without being opinionated about data (state).

The book has information in it up until at least the beginning of 2015. It is well worth reading as an introduction and overview of Docker.

This article originally appeared on the ScraperWiki blog and is reproduced with the kind permission of the author: http://ow.ly/RnJI8

The Annotated Build-It-Yourself Science Laboratory
Windell Oskay
Maker Media
ISBN: 978-1-4571-8689-9
334pp.
£ 16.50
Published: May 2015

reviewed by James Roberts

I was looking forward to seeing this book, but my enthusiasm was short lived: this is not at all the book I was expecting to see or hoping for.

It is a republishing of a classic US book for school children of the 1950's and 60's. It is aimed at, I would estimate, 10 to 14 year olds, and was an adjunct to the School Science Fair movement of the time which still is very strong in the US but largely absent in the UK.

It is essentially a sentimental journey; the original 1959 text is presented largely as-it-was with annotations by the 2014 editor. Many of these annotations are safety oriented, as many practices of the 1950's are not acceptable today, including for example the cutting and use of white asbestos boards, mouth pipetting, and the use of mercury: there are hundreds of other similar warnings.

The book is organised into three sections addressing Chemistry (46 pages), Physics (77 pages), and Biology (25 pages), each section consisting of an investigation into equipment and techniques, detailing the construction of measuring instruments, and then investigation of phenomena using those instruments and techniques.

After the sections there is a 34 page Appendix which attempts to deal with the difficulties in the US of obtaining the originally specified equipment and supplies today, over 50 years after the original publication – and this, again, even in the US.

I would not recommend that this book be given to a child in the UK. There are three main reasons for this:

1. The materials and equipment specified is entirely 100% US oriented. For example, we don't and never did have 'gallon' glass jars in the UK, and the same applies to I'd guess 85% of everything that is mentioned in the book. Every single item is a US (territories!) specific product that would have to be translated into an equivalent for the UK or other English-speaking countries. This is not straightforward.

2. Units! Units units units! The US is still, almost uniquely, using non-metric units in science. This book was written in the late 1950s, and uses US units including gallons, quarts, Fahrenheit temperature. . . (not UK gallons and quarts, or Celsius). Where it is using metric units they are some unholy mix of CGS and US imperial, it's no wonder they missed Mars.

3. Electrical Safety! The original text recommends practices that were borderline-safe in the 1950s using US mains electricity, sometimes rectified by selenium rectifiers (which I trust most readers of this have never had to encounter). While there are copious warnings about electrical safety, these still address US mains of 110V AC. UK mains is 240V AC or so, and as such more than four times more dangerous ($I = V$ squared over R); selenium rectifiers were dreadful in every way but also limited the current which could be drawn due to their high internal resistance, whereas the suggested silicon diode replacement in the annotated version does not current limit in the same way at all. In short, I regard all the mains-based electrical

experiments in this book as being potentially lethal in the UK and if the arrangements described were built as such in the UK I believe the builder would be liable for offences against IEC wiring and safety regulations and any fatalities resulting. It really is that dangerous. Your reviewer is not an electrical wuss; he has worked extensively with kV level high voltages (which just want to kill you) and regards domestic mains as 'low voltage'; your reviewer has designed electronic equipment for production runs. The mains-based electrical experiments in this book are in my opinion potentially instantly lethal in the UK and I would not let a kid near them.

While competent to comment on the electrical safety aspects, I only have O level chemistry and biology. I suspect that some of the chemistry work is equally doubtful as far as safety in the UK is concerned, and suspect many of the biological experiments suggested would be blocked by a modern ethics committee. There are warnings about all this, but the UK has a very different legislative environment to the US.

The book does have some real strengths: each section of the book ends with a lot (hundreds) of questions, many of them very good questions, which if pursued by a child (or adult!) would greatly enhance their understanding of laboratory work and processes and the general grind of scientific endeavour, as well as introducing expanded knowledge of the general principles introduced earlier in each section and extending the principles of enquiry.

However, and despite this, I would not recommend this book ever be given to a child unsupervised in the UK, and it was never intended by the original author to be used in such a way.

A genuinely informed and competent adult could find it to be a source of really good questions and lots of good techniques for working with children, but at the cost of translating almost every step from US practice into different UK-available equipment, supplies and application, and the consequent large added expenditure of time.

An adult with interest for themselves, on the other hand, would find much more of interest in a book published by Scientific American, 'the Amateur Scientist'. This is hard to find, but the entire series as well as the book contents is available on CD on the 'net, if you look around. If you want to build your own Van de Graaf generator and linear accelerator, that's where to go (I never did build the VdG, maybe when I retire). Of course it's still full of unholy units and unavailable US supplies.

I cannot recommend the present book for the UK, but do not take this as it being a bad book – it was a very good book in its place and time and context. What's needed for the UK is a completely different rewrite-from-scratch with many of the same sorts of experiments and questions, but translated into UK equipment and available supplies, with inherently UK-safe electrical practice, and for goodness sake, using SI units!

Contributors

Nigel Barker lives in East Lothian and works at Dell SecureWorks. His first exposure to Unix was when the comic distributor he worked for bought a Texas Instruments TI-1500 mini computer in 1989.

Charlie Harvey is IT Director of New Internationalist workers co-op, who produce an award-winning magazine, books about global justice and run ethical online shops. He has been working with free software since the late 90s.He enjoys cider and blogs at http://charlieharvey.org.uk.

Ian Hopkinson is Senior Data Scientist at ScraperWiki following work as an academic and a research scientist at Unilever. Throughout this time he has rummaged through data with a wide variety of open source tools.

Kimball Johnson has been programming since a very early age, starting with BBC Micros, then MS DOS and Windows Systems, however was enlightened with a copy of Debian GNU/Linux Woody at university. From this developed an affinity to Systems Administration, but like all good admins he is lazy and so tries to automate as much of his job as possible, and now follows the way of the Infrastructure as Code.

Mark Keating is Managing Director at Shadowcat Systems, a member of FLOSSUK, prominent in the Perl community and organiser of events such as London Perl Workshop. Follow him on Twitter: `@shadowcat_mdk`.

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

Les Pounder works closely with North Western Linux and Free Software groups to promote the use of Open Source software as opposed to proprietary software. He is also the organiser of UCubed, a free Linux and open source event in Manchester, and an organiser of Barcamp Blackpool and Blackpool Geekup, and has been head of crew at Oggcamp. He writes for Linux Format magazine and contributes to Linux podcasts including Fullcircle, Ubuntu UK Podcast and Linux Outlaws.

James Roberts was one of the last people in the UK to learn to program on 80-column cards at Control Data, and ran into Linux SLS in 1992. He runs a small company/consultancy that sneaks Linux and *BSD into SME Windows shops wherever it fits. He has inter alia taught micro-computing and MIDI and audio recording to young adults.

Paul Waring works as a freelance IT consultant. Outside of work he can usually be found filing documentation bugs against various open source and free software projects. He also edits the FLOSS UK newsletter.

Quentin Wright is a Director of Sunfield Technology working with Linux and Ruby and system integration projects. In his spare time in between playing with reluctant motor vehicles he is pre-occupied with Web and Javascript programming.

Contacts

Kimball Johnson
Chairman
Preston

Gavin Atkinson
Council member
York

Tim Fletcher
Council member
Manchester

Ian Norton
Council member
Morecambe

Stephen Quinney
Council member
Edinburgh

Quentin Wright
Treasurer
Warwick

Paul Waring
Newsletter Editor
Manchester

Alain Williams
System Administrator
Watford

Holger Kraus
Website
Leicester

Sam Smith
Events and Website
Cambridge

Jane Morrison
FLOSS UK Secretariat
The Manor House
Buntingford
Herts
SG9 9AB
Tel: 01763 273475
Fax: 01763 273255
`office@flossuk.org`