



# 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.flossuk.org/Newsletter>

---

**Volume 22, Number 3**

**ISSN 0965-9412**

**September 2013**

---

## **Contents**

<b>From the Secretariat</b>	<b>3</b>
<b>Chairman's report</b>	<b>3</b>
<b>Vintage UKUUG shirts!</b>	<b>4</b>
<b>Call for Papers: Spring Conference 2014</b>	<b>4</b>
<b>3D Printing: the Here and Now</b>	<b>5</b>
<b>Book review: Relational Theory for Computer Professionals</b>	<b>6</b>
<b>Book review: Functional Javascript: Introducing functional programming with Underscore.js</b>	<b>7</b>
<b>Book review: Propose, Prepare, Present</b>	<b>8</b>
<b>Book review: MongoDB: The Definitive Guide</b>	<b>8</b>
<b>Book review: Vagrant: Up and Running</b>	<b>9</b>
<b>Book review: GIMP 2.8 for Photographers</b>	<b>11</b>
<b>Book review: Learning Python, 5th Edition</b>	<b>13</b>
<b>Book review: Graph Databases</b>	<b>15</b>
<b>Book review: Realm of Racket</b>	<b>16</b>
<b>Contributors</b>	<b>16</b>
<b>Contacts</b>	<b>18</b>



## From the Secretariat

*Jane Morrison*

We are currently working with Josette Garcia and various tutors to bring you some tutorials for later this year.

The AGM this year will be held on Friday 20th 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 on 16th August.

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

The un-conference being planned for Saturday 26th October in London is still in the planning stages but we shall provide you with full information as soon as we have it.

Spring 2014 event “FLOSS UK DEVOPS” – Brighton, 18th, 19th & 20th March, please see the Call for Papers printed in this issue. Why not submit a proposal? Closing date for abstracts is 15th November.

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 15th November. As usual any articles, letters etc. can be sent for inclusion to: [newsletter@ukuug.org](mailto:newsletter@ukuug.org).

---

## Chairman's report

*Kimball Johnson*

### Events

OpenTech and Barcamp Birmingham were both excellent events, thank you once again to Sam Smith for organising OpenTech, and everyone that attended the two events.

There will be an Unconference in London in October, further details of which will be announced shortly. Next March the annual Spring Conference is in Brighton, and the call for papers is open, please see the website for more details.

### 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.

The first grant has been awarded to Hackspace Manchester, a community workshop in Manchester's Northern Quarter. FLOSS UK has bought them a kit for a Mendel 90 3D printer, and in return there will be a series of articles on 3D printing in general, and on the build of the printer, and the software for operating it. This will be over the next few newsletters starting from this one with a short introduction.

If you have an idea and wish some support, please contact Jane on [office@flossuk.org](mailto:office@flossuk.org) and it will be discussed by Council.

### **Get Involved**

FLOSS UK 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. If you would like to help out in any capacity please do get in touch via [office@flossuk.org](mailto:office@flossuk.org).

---

## **Vintage UKUUG shirts!**

***Jane Morrison***

During the annual office clear out and tidy up we have come across a stock of UKUUG polo shirts.

Back in 2006 they were very popular with members and we sold a number of them at £15 inclusive of VAT and postage.

As we now want to try and shift the remaining stock we are offering them to members at the special price of £5.00 including postage and VAT.

They are all black with the UKUUG logo embroidered in silver.

Sizes available: XL and Large with just 2 small and 3 medium remaining.

You can pay via PayPal, credit card, BACS or cheque.

Contact the Secretariat now on 01763 273475 or email [office@ukuug.org](mailto:office@ukuug.org).

---

## **Call for Papers: Spring Conference 2014**

The Spring Conference 2014 will be held in Brighton between 18th March – 20st March 2014. A call for papers has been issued, as below.

### **Call for Papers**

Recent past conferences have included sessions on: Perl for Systems Administrators; IPv6; Kerberos, Samba, Scons, Haskell, Chef, Ansible, JuJu and Coffee Script. If you would like to offer a half day workshop that might interest our target audience, please submit a proposal.

- backup and restore
  - data de-duplication
  - security and audit
  - authentication and authorisation
-

- virtualisation and system deployment using ‘cloud’ technology
- operating systems
- network/cluster file systems
- ethics and legislative compliance
- storage solutions
- databases and directory services
- nomadic and wireless computing
- cluster management
- configuration management
- scripting and task automation
- benchmarking and performance tuning

If you have a novel solution to a problem, experience of a particular application or hardware platform, tips and tricks for fellow systems administrators, or a favourite tool you could talk about, please submit a paper for consideration by the programme committee.

There will be a prize for the best talk.

We are also seeking lightning talks for the afternoon of Wednesday 19th March. If you have not spoken before, or if you don't have enough for a longer talk this is a great opportunity for a less formal talk format.

**Significant Dates:**

Initial closing date for abstracts: 15th November 2013

Speakers notified by: 29th November 2013

The call for papers is open, this year we are asking speakers to submit online rather than email. The site to do so is:

<http://conference.flossuk.org/en/spring2014/cfp/session/new>

---

## **3D Printing: the Here and Now**

***Bob Clough***

You've probably heard a lot about 3D Printing in the press, and the majority of it is probably excited speculative articles about "home factories" and "the future of manufacturing". While the picture these articles paints is a fantastically rosy one, as a Maker I'm more interested in the uses the technology can be put to today.

Just around the Hackspace, 3D printing has come in very useful. When a 30 year old bandsaw was donated with one of the guides missing, I was able to design and print a complete replacement in under 4 hours [1] using just a set of calipers, some free software, and a 3D Printer. When an overzealous member broke the handle of our PCB guillotine, we could replace it the same day with an upgraded model using about 50p worth of plastic.

---

Of course, most people don't need replacement parts for bandsaws, but 3D printing is also useful around the home. When a friend broke his shower holder, he was able to replace it with a printed model quicker than cycling to B&Q [2]. Teenage Engineering, the manufacturers of a popular synthesizer, have recently released all the design files for replacement knobs and buttons for their synths in a 3d printable format, allowing their customers worldwide to print out replacement parts in mere hours [3].

Replacement knob for a washing machine the manufacturer no longer supports? Download it from the internet [4], print it out. Want some cheap handles for a cupboard in your garage? 3D print them [5]. For those of us who have access, 3D printing isn't the future, it's here and now.

Over the next few articles, Hackspace Manchester will be documenting the build of a popular open 3D printer, the Mendel 90 kindly donated by FLOSS UK. To be covered will be the build process, the software stack, that is entirely open source, and some practical uses of the printer.

[1] <http://thinkl33t.co.uk/3d-printing-bandsaw-fence>

[2] <http://www.thingiverse.com/make:38183>

[3] <http://www.wired.co.uk/news/archive/2012-10/01/3d-print-synth-parts>

[4] <http://www.thingiverse.com/thing:10833>

[5] <http://www.thingiverse.com/thing:26991>

---

## **Relational Theory for Computer Professionals**

**CJ Date**

**O'Reilly Media**

**ISBN: 978-1-4493-6943-9**

**284pp.**

**£ 30.99**

**Published: May 2013**

**reviewed by Lindsay Marshall**

I've been around long enough that when I was a student, databases as we know them now had not really been perfected and so I have never actually been taught anything about databases or database theory. (Strangely, though, I have taught a course on databases, but not with no theory).

So, I have to confess that there's a lot of techniques that I use when writing SQL that I know work, but that I couldn't honestly tell you why they work. Unfortunately, the books that introduce you to this kind of thing tend to be quite large and a bit daunting and easy to put off reading, so this book seems to have several advantages. First off it is written by Date who is a genuine database guru who writes readably and well, and second it's quite slim. It is also aimed squarely at people with experience of programming. On the downside there's quite a lot of mathematical notation in there and he introduces abbreviations for concepts

---

that mean that you can't just dip into the book – you really do have to start at the beginning and read through it.

I have started at the beginning and I am reading through it. It's not exactly light reading but neither is it impossibly heavy going and it's telling me stuff I want (and probably need) to know in an accessible way. If you need an introduction like this, then I reckon this is definitely worth your while.

---

## **Functional Javascript: Introducing functional programming with Underscore.js**

**Michael Fogus**

**O'Reilly Media**

**ISBN: 978-1-4493-6072-6**

**260pp.**

**£ 22.99**

**Published: June 2013**

**reviewed by Lindsay Marshall**

Sometimes you look at a book, and think “yes, but...” and this is definitely one of them. I use both JavaScript and Underscore.js and I am not shy about functional programming either when it is appropriate, but I get nervous when people start being purist about things. And this book definitely gives off a definite whiff of purism from the start (though that may just be me being cynical of course). One of the Forewords (there are two for some reason) starts with the line “this is a terribly exciting book”. Unfortunately, it really isn't, in fact, I started to wilt quite quickly as I got in to it. As I just said, I am quite familiar with functional programming (and have even taught it) and the concepts and theory behind it, but I found the explanations in the first chapter over technical and clumsy and most certainly not for beginners. Though to be fair it is unlikely that any beginner would pick up the book, unless they thought the title meant “JavaScript that works”.

However, let me say straight away that the ideas presented are entirely sound and the code examples are frequently very clever indeed, and the coverage of functional techniques is complete, but it all feels a little like a religious exercise, rather than a sensible way of doing things. Very much, “Look what I can do with JavaScript!” “Very nice dear, now eat your porridge”.

If you are a functional programming person and you like JavaScript then this book is definitely for you, otherwise I would tread carefully. There are certainly some good ideas in here that are worth picking up on, but whether you want to dive wholeheartedly into doing all your JavaScript functionally, that's another matter entirely.

---

**Propose, Prepare, Present****Alistair Croll****O'Reilly Media****ISBN: 978-1-4493-6637-7****80pp.****£ 11.50****Published: June 2013****reviewed by Lindsay Marshall**

The subtitle for this book is “how to become a successful, effective, and popular speaker at industry conferences”. It is 56 pages long. I thought it was going to be a book about presenting – something I have a particular interest in – but no.

The chapter on presenting is 9 pages long and there is almost nothing about actual presentation in it, though to be fair there are some interesting tips like not wearing finely patterned clothing as it tends to produce moiré effects on video. The rest of the book is (briefly) about what to do and not do when pitching a talk to a conference (including the tip that all other things being equal it's better if you are a member of a minority). The blurb claims that tens of thousands of companies apply to speak at conferences every year. If this is true then perhaps there is indeed a market for this book, but the content does seem to boil down to: be relevant, be interesting and don't oversell – sound advice but pretty obvious. Overall this feels like material that would make a useful website but really not much more than that.

I suppose if you are wanting to get a start in the business of presenting at industry conferences (not academic ones of course, where peer review is the first hurdle), then you might find this slightly useful, but overall I just don't see the point in it.

---

**MongoDB: The Definitive Guide****Kristina Chodorow****O'Reilly Media****ISBN: 978-1-4493-4468-9****432pp.****£ 30.99****Published: May 2013****reviewed by Paul Waring**

MongoDB is one of the more well-known ‘NoSQL’ (non-relational) databases which have received a lot of attention over the last few years. As a relatively new piece of software – at least compared to its relational cousins – it has developed at a rapid pace, and this book has been updated after just three years to take account of the changes.

The first chapter is just a couple of pages about why you might want to use MongoDB – although if you're picking up a book at this level, you probably don't need convincing. This is followed by a chapter outlining all the basic concepts in MongoDB, and how they differ from relational databases (e.g. you have collections of documents, rather than tables with

---

columns and rows). The MongoDB shell, a command line interface, is also covered in depth across the next two chapters, with demonstrations of all the standard query operations.

The second part of the book begins with two chapters covering indexing. These start off with an explanation of how to use the `explain()` and `hint()` functions to show how a query is processed, including whether it uses any indexes, followed by the different types of indexes and where and when you should (and, just as importantly, should not) use them. The rest of this part of the book covers aggregating data and application design, and a short section on when not to use MongoDB (e.g. if you need transactions).

Parts three and four cover replication and sharding respectively – important topics if a database is expected to scale reliably. The one-minute test setup serves as a useful introduction to the subject, as well as an environment to work through some of the concepts without having to install multiple MongoDB instances. Whilst the implementation details are specific to MongoDB, much of the advice is sufficiently generic to be of interest to users of other database software.

Part five discusses application administration, though the majority of this is dedicated to dealing with problems such as phantom operations, out of memory errors and repairing data files. Definitely worth a read, ideally before you run into problems!

The sixth and final part is dedicated to system administration. As you might expect, this goes into a lot of technical detail, covering the standard topics which any administrator wants to know, including memory use, monitoring, backups and performance tuning.

Overall, this is a comprehensive guide to MongoDB. It is perhaps a little too advanced for complete beginners – despite the claims on the back cover – but otherwise does a good job of covering most of the topics which you are likely to need.

---

## **Vagrant: Up and Running**

**Mitchell Hashimoto**

**O'Reilly Media**

**ISBN: 978-1-4493-3583-0**

**156pp.**

**£ 22.99**

**Published: June 2013**

**reviewed by Paul Waring**

As a developer, one of the biggest challenges I find when testing or debugging code is building a sandboxed environment which mirrors production, yet can easily be created and destroyed. Vagrant is a piece of software to support the automated creation and provisioning of virtual machines for exactly this purpose. I've been looking forward to the release of this book for some time, as whilst the online documentation is excellent, I find a well-written book is often easier to follow when first learning about a topic.

One important myth that is refuted in the first chapter is that Vagrant only works with VirtualBox. Whilst this was the case prior to v1.1, Vagrant now supports all the major virtualisation layers, as well as Amazon EC2, although the book does use VirtualBox for all of

its examples. The standard topics of an introductory chapter are also covered – installation, why you might want to use Vagrant, and where to find support and help.

The second chapter covers the basics of setting up your first Vagrant machine, including a sample Ubuntu Precise VM which is pre-configured. Each part of the sample configuration file is examined, from basic networking to how the box frees up resources on destruction. This dovetails nicely into the following chapter, which shows how to extend a bare bones machine with automated provisioning, beginning with shell scripts and moving on to cover Chef and Puppet (albeit briefly).

Chapter four covers networking, and at only 10 pages I felt this was a little light on detail, given how complex various network options and their interactions with each other can be. The same applies to the next chapter, which covers the modelling of clusters containing multiple machines, but only briefly, and with a sole real life example (MySQL) which wasn't actually a cluster.

Despite being referred to throughout the text, boxes – the base images upon which Vagrant environments are built – do not get full coverage until the sixth chapter. All the basic aspects of boxes are discussed, as well as some useful tips on specific areas such as removing udev rules which can cause boot failures.

The final chapter of the book is by far the longest and explains how to extend Vagrant using plug-ins. This is perhaps a little advanced for an 'up and running' text, and I would have preferred greater detail on the topics in previous chapters. Anyone who is already familiar with RubyGems will find this chapter easier to work through, as Vagrant plug-ins are packaged as gems.

As with many technical books, there are a couple of appendices which in this case cover environment variables and configuration options. However, given Vagrant's rapid development this is one of the areas where the online documentation is likely to be more up to date and useful.

Overall, I felt this book provided a good broad overview of Vagrant, and would be sufficient for a developer who wants to create simple sandboxed environments for testing code. It's also an ideal starting point from which you can jump off into the more comprehensive online reference documentation.

---

## **GIMP 2.8 for Photographers**

**Klaus Goelker**

**O'Reilly Media**

**ISBN: 978-1-937538-26-2**

**400pp.**

**£ 30.99**

**Published: May 2013**

**reviewed by Andy Thomas**

At last, a book on GIMP that is aimed squarely at serious photographers. While a number of books on GIMP are already available, many of these concentrate on its image creation and graphic design features, with little coverage of using this powerful tool for anything more than the most basic photographic tasks such as cropping and resizing images. But written by a photographer for photographers, this book finally meets the need for a book that will show photographers ranging from keen amateurs to professionals how to use much more advanced digital correction, editing and enhancement techniques such as colour curves, layers and the exciting new facilities made possible by the recent integration of the GEGL libraries into GIMP 2.8.

Organised as just 5 large chapters plus appendices the book is accompanied by a comprehensive DVD containing not only the GIMP 2.8 installables for Windows, Mac and Linux but also a large number of sample photos for use in the hands-ons exercises described in the book, a selection of useful free programs (mostly for Windows and Mac), a list of useful links to further resources and the book itself as a PDF e-book.

Easing into the book, chapter one entitled 'Basics' starts with an overview of GIMP followed by a useful refresher on digital image essentials before moving on to the installation of GIMP on Windows, Mac and Linux computers. GIMP nowadays is platform and operating system agnostic and a nice thing about this book is it adopts this same approach; although lavishly illustrated with screenshots from a Windows PC, the author treats all platforms equally which helps widen the book's appeal to users of just about all computing systems (yes, GIMP runs fine on FreeBSD and there are efforts in progress to port this to Android devices too).

The main meat of the book is in chapters 2, 3, and 4 which in turn cover the topics of correcting and touching up images, using masks and layers along with the colour tools and working with black and white (grayscale) images respectively. Throughout these chapters the emphasis is on learning by doing – organised as a series of 'mini-workshops' the common and even some more unusual tasks are introduced one by one; each operation first discussed in detail and then the reader is invited to access a sample image from the DVD and try out tasks such as correcting the rotation of a scanned image where the original was not positioned very accurately on the scanner, for example. And while on a particular task, the author often mentions other common issues and problems related to this, such as the moire effects that sometimes mar scanned images, for example, and the tutorial branches off to offer solutions for these issues.

It is in these chapters that the excellent illustrations and screenshots throughout the book really come into their own – with the image being worked on surrounded by the relevant

GIMP windows helpfully annotated with comments. And almost as if the author Klaus is able to read a curious reader's mind, he has the knack of pointing out other features, buttons and controls that appear in the GIMP windows, explaining clearly what they do and how they can be used to modify the result of the task currently in progress. Alternative methods of achieving the same result, along with their pros and cons, are presented throughout the book and to help the reader who has somehow become lost doing a particular exercise, the DVD offers a helping hand by including samples of each stage of a given task as they are being worked on as well as the finished result, so the reader can see what is supposed to happen in an exercise and very often can reverse or reset changes.

Chapter 4 is particularly welcome as it is all too easy to forget that you can't simply pop a black and white film into a digital camera! Although black and white-only digital cameras do exist they are very expensive and most of us will have to make do with converting colour images to grayscale. There is a lot of useful information on achieving a successful conversion as well as on colouring exiting black and white images, decomposing images and the use of channels to extract particular areas of interest.

Chapter 5 covers working with camera RAW files such as DNG (Digital Negative Format) and camera manufacturer's proprietary RAW formats and we are treated to a detailed tutorial on using UFRaw both as a GIMP plug-in and as a stand alone program to 'develop' digital negatives before further processing using other software. Here the author makes the point that to get the best out of any digital camera, images should always be saved in the camera's native RAW format rather than the popular JPEG format – it seems a shame this advice is imparted towards the end of the book although to be fair, the book deals with working with images that have already been taken in one way or another when it's already too late to go back and re-shoot in RAW format! Admittedly I know very few digital photographers who use RAW even occasionally, never mind on a regular basis.

In contrast to the main chapters, the appendices are very short indeed but nevertheless sweet, mentioning the IWarp filter by way of some light relief, a description of what's on the DVD, a table of the file formats GIMP can use along with the many additional formats supported by the UFRaw plug-in and rounding off with the usual index.

I have thoroughly enjoyed reviewing this not only because it has been a complete and compelling distraction for me over the last few weeks, lifting me out of the grey IT infrastructure world I normally inhabit, but being a keen photographer myself for many years of the "old school" kind, using film until fairly recently and striving to get original shots right every time, I have never done any digital image post-processing myself beyond cropping and brightening up the odd scanned picture with ImageMagick. This book is changing that – I admit I haven't done all the exercises yet (but I will) but it has given me the confidence to worry a bit less about getting the original photo absolutely right and instead concentrate on the picture first and then tidying things up afterwards using GIMP. It has also opened my eyes to the incredible range of features within GIMP.

Well-written in a pleasant conversational style, this is a quality book – colourful, well laid out and printed on thicker, heavier grade paper than most, with 386 pages and about 3.5 GB on the companion DVD. I can recommend this book without hesitation for any intermediate to advanced photographer who really wants to get stuck into digital image processing.

---

## Learning Python, 5th Edition

Mark Lutz

O'Reilly Media

ISBN: 978-1-4493-5573-9

1600pp.

£ 49.99

Published: July 2013

reviewed by Andrew Richards

At a daunting 5lbs / 2.25 kg for the paper edition, your postie will not be best pleased with you when delivering this book, and if you're just exploring Python at the moment, this is not the right book – such a large and complex book is thoroughly intimidating (for an excellent digestible intro to Python programming I'd recommend instead Toby Donaldson's *Python: Visual Quickstart Guide*, or digging around the web for free learning resources, especially YouTube videos like the 'Google Python Class').

If however you have already gained enough Python to know that you want to learn the language in depth, then this book provides an excellent way of doing so. The author takes considerable trouble to winkle out the detail in the language, including careful consideration of Python 2.x and 3.x, which is very helpful in understanding the differences between the two (e.g. a good description of why both `sort()` and `sorted()` exist – they serve similar purposes).

Despite clearly being a strong advocate of Python he nevertheless explores the warts in Python, especially overlapping features. The book in its 5th edition so it does have the feel of having matured over many years with each new edition; I was impressed by there being relatively few typos etc, and those that I have spotted seem pretty minor.

The book has almost 1600 pages; for this review I've read about 400 so far (I'd miss the review deadline if I read more). Some basic computing and programming concepts are assumed, for example ASCII, variable declaration (even though Python doesn't do this), boolean bitwise logic, and some programming terms like scope.

It describes itself as a tutorial rather than a reference book, and I am reading it with that in mind. I think however that it will prove useful as a reference too, with its short explanatory code examples likely to prove very helpful when I'm teasing out details in my own programming.

The book focuses on the language itself rather than its libraries, although the most fundamental libraries do receive some coverage. The author has in fact written a further book *Programming Python* – with even more pages; *Learning Python* and *Programming Python* are described as forming a two-volume set.

The author also offers Python training courses. My experience of training courses in general is that the first hour or so is given over to administrivia before you actually get stuck into the course. This seems to have carried over into this book: It takes a very long time to get going: The first 100 pages are arguably very wordy without actually getting you properly started in the language, as well as having some very technical coverage of how Python executes code that is given far too early – it would have been better to get the 'Hello world'

and '2 + 2' type stuff out of the way before scaring off readers with elaborate detail about how modules are imported/exec-ed/reloaded. Significant chunks of this part of the text is given over to pre-empting possible mistakes that new learners to Python may encounter, resulting in excessive detail, for example Chapter 3 *How You Run Programs* has almost 50 pages. This level of detail does sometimes obfuscate the often straightforward concepts being described.

I'd have much preferred a far more lightweight intro to give the bare minimum to start entering Python code and seeing results, saving the detail for a later chapter. For anyone who's got enough Python to know how to edit and run a program in Python, skipping the start of the book to page 93 is probably a good idea. I don't accept that 93 pages (plus further roman-numeral pages) are needed to get you to that point.

The style of the book varies, but the early chapters tend to cover a topic very thoroughly in each chapter; explaining what the language does in depth and only then showing you the corresponding Python code. This is fine, but it's possible that by allowing the code to have 'driven' the text a bit more, it might have been given a bit more pace and clarity. There is some repetition in the text, partly needed by virtue of the amount of detail being given. The author sometimes makes reference to other languages, esp. C. On balance I like these references since it prevents Python being presented in a vacuum.

One section I particularly liked was the patient approach to explaining Object Oriented Programming (OOP). In contrast to many of the prior chapters, the author allows the code to 'drive' the text more at this point, resulting in a very accessible exploration of OOP, including the gentle introduction of various OOP concepts. I was very taken by the clarity of the books' comparison of the inheritance and composition approaches to structuring an OOP project, and appreciated the way that concepts were introduced carefully, one at a time, and that code was there to demonstrate what was being described. The author's eye for detail was used to elaborate on some of the more detailed aspects of OOP in Python, making for a greater depth of understanding than I've achieved from other sources – and that's just the book's overview chapters of OOP (75 pages) – the really detailed nitty-gritty of OOP is then covered in subsequent chapters.

Important subjects are covered in depth that are sometimes lacking in other texts – exceptions, Unicode and metaclasses all get thorough coverage (note that I've not actually read those sections yet). There is also some more 'political' discussion of Python, esp. the tendency for features to be added to Python, resulting in Python programmers needing to know a larger Python vocabulary / range of idioms so that they will understand whatever Python code gets thrown at them – particularly in a multi-person project, where the use of a feature by one member effectively mandates that all other project members need to be familiar with that feature.

Overall, this is not a book for the complete beginner or the merely curious unless they have a lot of stamina – better to start prospective Pythonistas on a more lightweight text. However, once past the beginner or curiosity stage, this book is excellent for providing truly in-depth coverage of Python, exploring in considerable detail the various features of Python, and signposting clearly the differences between Python 2.x and 3.x. It unfortunately loses its way right at the start with an avalanche of detail before actually getting down to business, making it hard work at first – but perseverance brings rewards here, with the rest of the book proving very rewarding and detailed, and likely to be a lasting and trusty companion when learning the language.

---

## **Graph Databases**

**Ian Robinson, Jim Webber and Emil Eifrem**

**O'Reilly Media**

**ISBN: 978-1-4493-5626-2**

**224pp.**

**£ 22.99**

**Published: June 2013**

**reviewed by Roger Whittaker**

Not so long ago the term “graph” to most people suggested drawing parabolae on squared paper at school. I well remember years ago explaining patiently that the “graph theory” that I was reading about in mathematics was something quite other than that.

But now everyone has heard of “six degrees of separation” and the “social graph”, and much of the data that we constantly access is most conveniently thought of in the form of nodes and relationships between those nodes, and is hence representable in the form of a graph.

This book is a short introduction to the concepts of graph databases: databases designed for holding information represented in this way. The authors explain how “relational” databases and SQL, though superb in their own traditional niche, fail badly if you try to push them into this sphere. Extracting a list of all the friends of the friends of the friends of an individual where the friendship relationship is modelled in traditional database tables is not surprisingly computationally very expensive. So “relational databases” are not actually very good for modelling relationships. Nor are most “NoSQL” databases (the authors explore this in an appendix on NoSQL design).

Where the book uses concrete examples, it uses Neo4j and its Cypher query language. Neo4j is an open source graph database written in java, specifically to solve the problem of storing graph oriented data. The book is mostly about strategies and patterns for dealing with graphical data rather than detailed code.

I found this an interesting introduction that I had heard about but never explored, which would be a useful jumping off point if one wanted to implement such a system for the first time.

---

**Realm of Racket****Matthias Felleisen et al****No Starch Press****ISBN: 978-1-59327-491-7****294pp.****£ 27.49****Published: June 2013****reviewed by Roger Whittaker**

Racket is the language formerly known as PLT Scheme, a variant that Matthias Felleisen, the main author of this book began work on in the mid 1990s. It is probably fair to say that it is mainly a research and educational language, with a particular emphasis on its use to teach programming. There is a nice IDE and programming environment called DrRacket which is used in a number of universities and colleges (particularly in the US) for introductory programming courses.

This book seems to be aiming very much at that introductory market, and I found it unsatisfying in terms of content. There is a good introduction which nicely details the history of Lisp and Scheme and explains what is good about Lisp-like languages. The book then follows the path of teaching through a number of extended examples, all of which are simple games. Personally I find this style far less helpful than teaching small snippets of code each of which shows how to do one particular thing. Although the book introduces basic concepts well, if one wanted to write one's own program in Racket for the first time, this book would not be a great help.

Like all No Starch Press publications, the book is nicely produced which very clear and readable text and code. Unfortunately there is an emphasis on "fun" with quite a few pages of dreadful cartoons, which add nothing to the text.

---

**Contributors**

**Bob Clough** is a Founder Member of Hackspace Manchester ([hacman.org.uk](http://hacman.org.uk)) and Manchester's 3D Printer User Group ([3dpug.co.uk](http://3dpug.co.uk)).

**Kimball Johnson** is a Systems Developer for Lancashire County Council He 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. He is always wanting to learn and has recently started to take on embedded programming on a variety of devices, on everything from a Arduino to a Nintendo DS.

**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 of-  
fice.

**Andrew Richards** is an IT consultant specialising in Linux and Unix based email servers, general sysadmin and networking. He dabbles in C and Python and has written a number of open source add-ons and patches to qmail. In his free time he enjoys arty foreign films, badminton and cycling. Find him at [www.acrconsulting.co.uk](http://www.acrconsulting.co.uk)

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

**Paul Waring** is a Council member of FLOSS UK 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

Kimball Johnson  
UKUUG Chairman  
Preston

Paul Waring  
Treasurer  
Manchester

Holger Kraus  
Council member  
Leicester

Ian Norton  
Council member  
Rochdale

Stephen Quinney  
Council member  
Edinburgh

Quentin Wright  
Council member  
Warwick

Roger Whittaker  
Newsletter Editor  
London

Alain Williams  
UKUUG System Administrator  
Watford

Sam Smith  
Events and Website  
Cambridge

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