

Open Source *Yearbook* 2015

Opensource.com publishes stories about creating, adopting, and sharing open source solutions. Visit [Opensource.com](https://opensource.com) to learn more about how the open source way is improving technologies, education, business, government, health, law, entertainment, humanitarian efforts, and more.

Submit a story idea: <https://opensource.com/story>

Email us: open@opensource.com

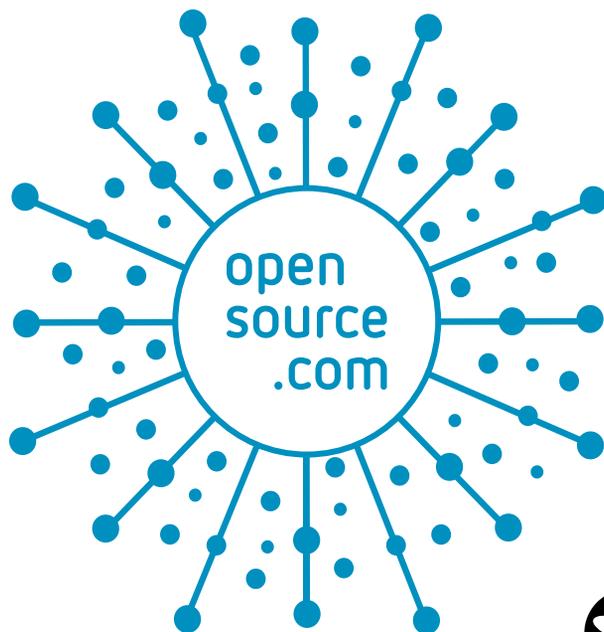
Chat with us in Freenode IRC: [#opensource.com](https://freenode.net)

Twitter @opensourceway: <https://twitter.com/opensourceway>

Google+: <https://plus.google.com/+opensourceway>

Facebook: <https://www.facebook.com/opensourceway>

Instagram: <https://www.instagram.com/opensourceway>



SUPPORTED BY RED HAT

Dear Open Source Yearbook reader,

The “open source” label was created back in 1998, not long after I got my start in tech publishing [1]. Fast forward to late 2014, when I was thinking about how much open source technologies, communities, and business models have changed since 1998. I realized that there was no easy way—like a yearbook—to thumb through tech history to get a feel for open source.

Sure, you can flip through the virtual pages of a Google search and read the “Best of” lists collected by a variety of technical publications and writers, much like you can thumb through newspapers from the 1980s to see the how big we wore our shoulder pads, neon clothing, and hair back then. But neither research method is particularly efficient, nor do they provide snapshots that show diversity within communities and moments of time.

The idea behind the Open Source Yearbook is to collaborate with open source communities to collect a diverse range of stories from the year. We let the writers pick the criteria, which means the yearbook isn't just full of the fastest, most popular, smartest, or best looking open source solutions. Instead, the yearbook offers a mix of open source solutions and projects, from a range of writers and communities, to offer a well-rounded (albeit incomplete) glimpse at what open source communities and projects looked like in 2015.

We couldn't have put this yearbook together without contributions from the following writers:

- Alicia Gibb, CEO of Lunchbox Electronics
- Becky Stern, director of wearables at Adafruit
- Ben Cotton, support engineer group leader at Cycle Computing
- Ben Nuttall, education developer advocate for the Raspberry Pi Foundation
- Christine Abernathy, Developer Advocate on the Open Source team at Facebook
- Cindy Pallares-Quezada, an Outreachy alumni
- David Both, Linux expert and enthusiast
- Drishtie Patel, GIS Analyst and Missing Maps Project Coordinator at the American Red Cross
- Harris Kenny, VP of Marketing at Aleph Objects
- Italo Vignoli, founding member of The Document Foundation
- Jeff Triplett, Frank Wiles, and Jacob Kaplan-Moss, Django contributors
- Jesus M. Gonzalez-Barahona, co-founder of Bitergia
- John Esposito, Editor-in-Chief at DZone
- Jos Poortvliet, ownCloud community manager
- Mano Marks, director of developer relations at Docker, Inc.
- Michael E. Meyers, the VP of Developer Relations at Acquia
- Robin Mulwijk, Internet and e-government advisor
- Seth Kenlon, independent multimedia artist, free culture advocate, and UNIX geek

Thank you to everyone who contributed to the 2015 Open Source Yearbook.

Best regards,

Rikki Endsley

Opensource.com community manager

If you're interested in contributing to the 2016 Open Source Yearbook, email us at open@opensource.com or submit your story idea: <https://opensource.com/story>

Links

[1] History of the OSI (Open Source Initiative): <https://opensource.org/history>



CONTENTS

WORKING

- 4 6 creative ways to use ownCloud**
Jos Poortvliet
Learn about six favorite uses for ownCloud, the open source self-hosted storage platform.
- 6 6 useful LibreOffice extensions**
Italo Vignoli
Extend the features in the already feature-packed LibreOffice office suite with these six useful add-ons.
- 8 5 favorite open source Django packages**
Jeff Triplett, Jacob Kaplan-Moss, and Frank Wiles
Django contributors share their five favorite Django packages in 2015: Cookiecutter, Whitenoise, Django REST Framework, Wagtail, and django-allauth.
- 11 Facebook's top 5 open source projects of 2015**
Christine Abernathy
We round up Facebook's top five open source projects in 2015, measured by community activity and impact.

- 13 10 cool tools from the Docker community**
Mano Marks
Check out 10 cool tools that can help expand your knowledge and use of Docker.
- 15 5 handy Drupal modules**
Michael E. Meyers
Find out about five handy Drupal modules: Views, BigPipe, Rules, Features, and Lightning.
- 17 Top 5 open source frameworks every application developer should know**
John Esposito
The DZone editor in chief picks five open source frameworks every application developer should know.
- 20 10 helpful tools for a sys admin's toolbox**
Ben Cotton
We highlight 10 sys admin tools that released new versions in 2015.

Best Couple

Best Couple of 2015: tar and ssh

..... DAVID BOTH

22 Picking one Best Couple for the yearbook was no easy task. We explain why the 2015 honor goes to an odd couple: tar and ssh.

COLLABORATING

- 24 Top 10 open source projects of 2015**
Jen Wike Huger
Our editors flipped through our 2015 archives and picked 10 open source projects that were popular with readers.
- 26 Top 5 open source community metrics to track**
Jesus M. Gonzalez-Barahona
Which project community metrics should you be tracking? We examine five metrics that provide a multi-faceted view of your community.
- 29 Diversity in open source highlights from 2015**
Cindy Pallares-Quezada
Diversity efforts received more attention in 2015. We round up a few of the highlights.
- 31 2015 was a good year for creating the world's 'missing maps' with OpenStreetMap**
Drishie Patel
The Missing Maps project launched in 2014. Learn about the progress the project made putting people on the map using OpenStreetMap in 2015.

LEARNING

- 36** **5 great Raspberry Pi projects for the classroom**
Ben Nuttall
The Raspberry Pi is perfect for the classroom. Get started with these five great projects.
- 38** **3 open hardware projects for beginners**
Alicia Gibb
The Lunchbox Electronics team picks three favorite projects and one open hardware book to keep you learning and making.
- 39** **Publisher's picks: 29 open source books for 2015**
Rikki Endsley
Six publishers share picks for top 2015 open source-related book releases.
- 47** **8 books to make you a more open leader**
Bryan Behrenshausen
If you've resolved to become a more open leader in 2016, these books can help.

Most likely to succeed

10 projects to fork in 2016

..... JASON BAKER

- 34** Our Most Likely to Succeed honors go to 10 projects with enough momentum to continue their success in the new year.

Most Playful

Best open source games of 2015

..... ROBIN MUILWIJK

- 50** Robin covered the world of open source gaming news in 2015. He picks a few favorite games to receive the year's Most Playful honors.

CREATING

- 52** **10 tools for visual effects in Linux with Kdenlive**
Seth Kenlon
We round up top tools and plugins to get the most out of video processing with Kdenlive.
- 56** **Adafruit's best open source wearables of 2015**
Becky Stern
Adafruit picks favorite open source wearable products and projects from 2015.
- 58** **5 favorite 3D printing projects of 2015**
Harris Kenny
We look at five of the top innovative 3D-printing-related projects in 2015.

.....
60 Call for Papers / Editorial Calendar

61 7 Reasons to Write For Us / Follow Us

.....
All lead images by Opensource.com or the author under CC BY-SA 4.0 unless otherwise noted.

6 creative ways to use ownCloud

BY JOS POORTVLIET

ownCloud is a self-hosted open source file sync and share server [1]. Like “big boys” Dropbox, Google Drive, Box, and others, ownCloud lets you access your files, calendar, contacts, and other data. You can synchronize everything (or part of it) between your devices and share files with others. But ownCloud can do much more than its proprietary, hosted-on-somebody-else’s-computer competitors.

Let’s look at six creative things ownCloud can do. Some of these are possible because ownCloud is open source, whereas others are just unique features it offers.

1. A scalable ownCloud Pi cluster

Because ownCloud is open source, you can choose between self-hosting on your own server or renting space from a provider you trust—no need to put your files at a big company that stores it who knows where. Find some ownCloud providers [2] or grab packages or a virtual machine for your own server from our site [3].

The most creative things we’ve seen are a Banana Pi cluster [4] and a Raspberry Pi cluster [5]. Although ownCloud’s scalability is often used to deploy to hundreds of thousands of users, some folks out there take it in a different direction, bringing multiple tiny systems together to make a super-fast ownCloud. Kudos!

2. Keep your passwords synced

To make ownCloud easier to extend, we have made it extremely modular and have an ownCloud app store [6]. There you can find things like music and video players, calendars, contacts, productivity apps, games, a sketching app, and much more.

Picking only one app from the almost 200 available is



hard, but managing passwords is certainly a unique feature. There are no less than three apps providing this functionality: Passwords [7], Secure Container [8], and Passman [9].

3. Store your files where you want

External storage allows you to hook your existing data storage into ownCloud, letting you to access files stored on FTP, WebDAV, Amazon S3, and even Dropbox and Google Drive through one interface.

(Watch *Using Federated Cloud Sharing between ownCloud servers* on YouTube to learn how to configure external storage in ownCloud Server 8.1. [10])

The “big boys” like to create their own little walled gardens—Box user can only collaborate with other Box users; and if you want to share your files from Google Drive, your mate needs a Google account or they can’t do much. With ownCloud’s external storage, you can break these barriers.

A very creative solution is adding Google Drive and Dropbox as external storage. You can work with files on both seamlessly and share them with others through a simple link—no account needed to work with you!

4. Get files uploaded

Because ownCloud is open source, people contribute interesting features without being limited by corporate requirements.

Our contributors have always cared about security and privacy, so ownCloud introduced features such as protecting a public link with a password and setting an expire date years before anybody else did [11].

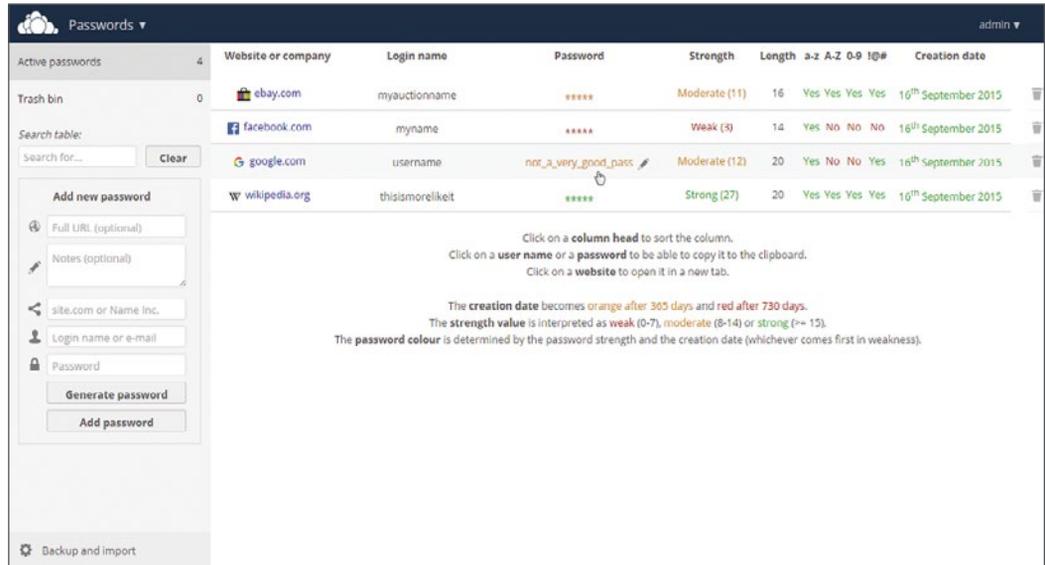
Today, ownCloud has the ability to configure a shared link as read-write, which means visitors can seamlessly edit the files you share with them (protected with a password or not) or upload new files to

*Installing ownCloud on a Banana Pi.
Photo by Jörn Friedrich Dreyer. CC BY-SA 4.0.*



your server without being forced to sign up to another web service that wants their private data. (See the *Creating a public upload folder in ownCloud 8.1* tutorial on YouTube. [12])

This is great for when people want to share a large file with you. Rather than having to upload it to a third-party site, send you a link, and make you go there and download it (often requiring a login), they can just upload it to a shared folder you provide, and you can get to work right away.



5. Get free secure storage

We already talked about how many of our contributors care about security and privacy. That's why ownCloud has an app that can encrypt and decrypt stored data.

Using ownCloud to store your files on Dropbox or Google Drive defeats the whole idea of retaking control of your data and keeping it private. The Encryption app changes that. By encrypting data before sending it to these providers and decrypting it upon retrieval, your data is safe as kittens.

6. Share your files *and* stay in control

As an open source project, ownCloud has no stake in building walled gardens. Enter Federated Cloud Sharing [13], a protocol developed and published by ownCloud that enables different file sync and share servers to talk to one another and exchange files securely. Federated Cloud Sharing has an interesting history. Twenty-two German universities [14] decided to build a huge cloud for their 500,000 students. But as each university wanted to stay in control of the data of their own students, a creative solution was needed: Federated Cloud Sharing. The solution now connects all these universities so the students can seamlessly work together. At the same time, the system administrators at each university stay in control of the files their students have created and can apply policies, such as storage restrictions, or limitations on what, with whom, and how files can be shared.

And this awesome technology isn't limited to German universities: Every ownCloud user can find their Federated Cloud ID in their user settings and share it with others.

Conclusion

So there you have it. Six ways ownCloud enables people to do special and unique things, all made possible because it is open source and designed to help you liberate your data.

Do you have other creative uses for ownCloud? Email your article proposals to open@opensource.com.

Links

- [1] ownCloud: <https://owncloud.com/>
- [2] ownCloud providers: <https://owncloud.org/providers/>
- [3] ownCloud server instructions: <https://owncloud.org/install/#instructions-server>
- [4] Installing ownCloud on a Banana Pi: <http://www.owncluster.de/2015/11/11/installing-owncloud-on-a-banana-pi/>
- [5] Host your "ownCloud" on a Raspberry Pi cluster: <https://christopherjcoleman.wordpress.com/2013/01/05/host-your-owncloud-on-a-raspberry-pi-cluster/>
- [6] ownCloud app store: <https://apps.owncloud.com/>
- [7] Passwords: <https://apps.owncloud.com/>
- [8] Secure Container: <https://apps.owncloud.com/content/show.php/Secure+Container?content=167268>
- [9] Passman: <https://apps.owncloud.com/content/show.php/Passman?content=166285>
- [10] Using Federated Cloud Sharing between ownCloud servers: <https://youtu.be/9-JEmIH2DEg>
- [11] ownCloud's Latest Community Edition Adds Video Streaming, and Easy Mounting of Third-Party Storage: <https://owncloud.com/owncloud45-community/>
- [12] Creating a public upload folder in ownCloud 8.1: <https://youtu.be/3GSppxEhmZY>
- [13] Announcing the draft Federated Cloud Sharing API: <http://karlitschek.de/2015/08/announcing-the-draft-federated-cloud-sharing-api/>
- [14] Sciebo: <https://owncloud.com/customer/sciebo/>

Author

Jos Poortvliet is a technology enthusiast and all-things-open evangelist. He's also community manager at ownCloud, and a SUSE and KDE marketing veteran. Jos loves biking through Berlin and cooking for friends and family.

6 useful LibreOffice extensions

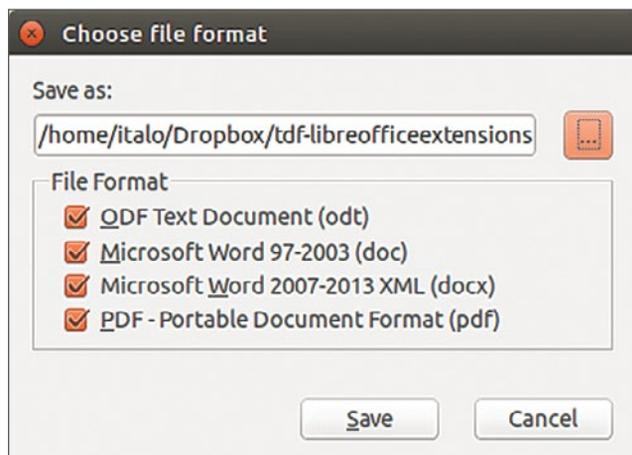
BY ITALO VIGNOLI

LibreOffice is the best free office suite around, and as such has been adopted by all major Linux distributions. Although LibreOffice is already packed with features, it can be extended by using specific add-ons, called extensions.

The main LibreOffice extensions website is extensions.libreoffice.org [1]. Extensions are tools that can be added or removed independently from the main installation, and may add new functionality or make existing functionality easier to use.

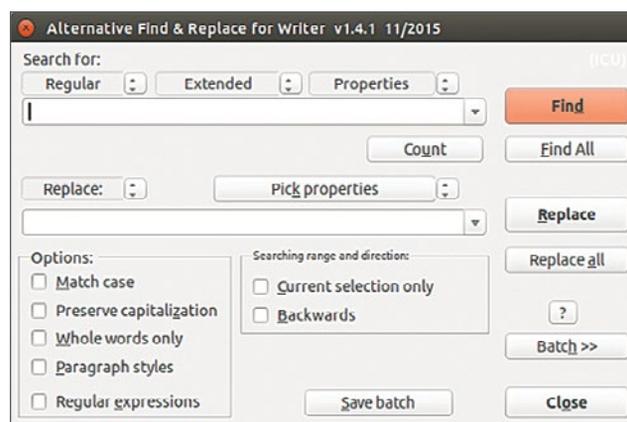
1. MultiFormatSave

MultiFormatSave [2] lets users save a document in the OpenDocument, Microsoft Office (old and new), and/or PDF formats simultaneously, according to user settings. This extension is extremely useful during the migration from Microsoft Office document formats to the Open Document Format standard [3], because it offers the option to save in both flavors: ODF for interoperability, and Microsoft Office for compatibility with all users sticking to legacy formats. This makes the migration process softer, and easier to administer.



2. Alternative dialog Find & Replace for Writer (AltSearch)

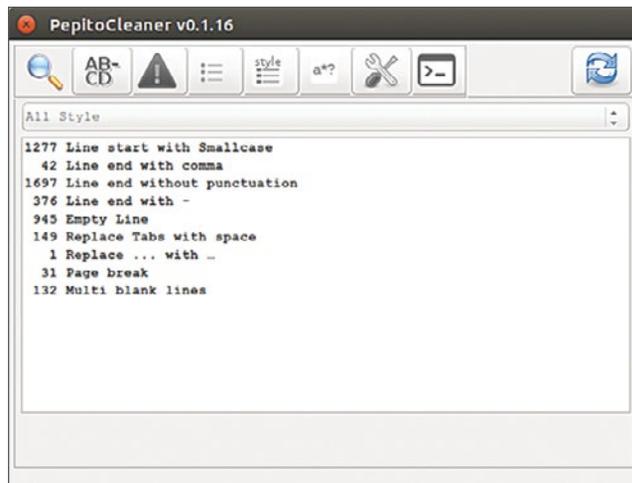
The AltSearch extension [4] adds many new features to Writer's find & replace function: searched or replaced text can contain one or more paragraphs; multiple search and replacement in one step; searching: Bookmarks, Notes, Text fields, Cross-references and Reference marks to their content, name or mark and their inserting; searching and inserting Footnote and Endnote; searching object of Table, Pictures and Text frames according to their name; searching out manual page and column break and their set up or deactivation; and searching similarly formatted text, according to cursor point. It is also possible to save and load search and replacement parameters, and execute the batch on several opened documents at the same time.



3. Pepito Cleaner

Pepito Cleaner [5] is an extension of LibreOffice created to quickly resolve the most common formatting mistakes of old scans, PDF imports, and every digital text file. By clicking the Pepito Cleaner icon on the LibreOffice toolbar, users will open a window that will analyze the document and show the results broken down by category. This is extremely useful

when converting PDF documents to ODF, as it cleans all the cruft left in place by the automatic process.

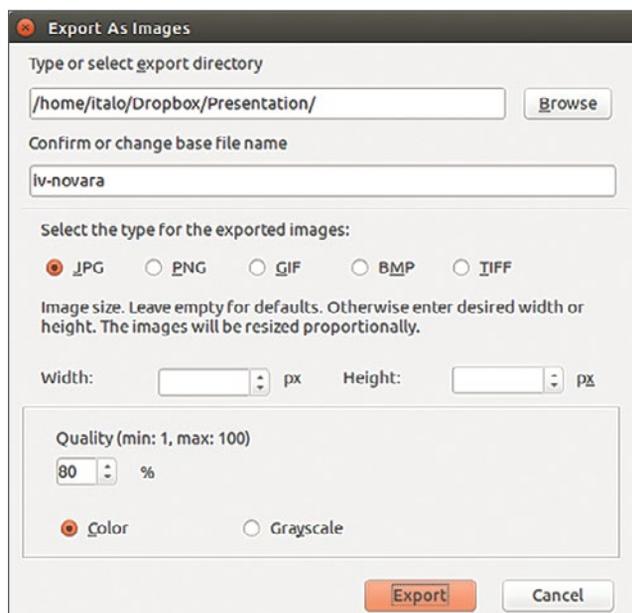


4. ImpressRunner

ImpressRunner [6] is a simple extension that transforms an Impress [7] presentation into an auto-running file. The extension adds two icons, to set and remove the autostart function, which can also be added manually by editing the *File | Properties | Custom Properties* menu, and adding the term *autostart* in one of the first four text fields. This extension is especially useful for booths at conferences and events, where the slides are supposed to run unattended.

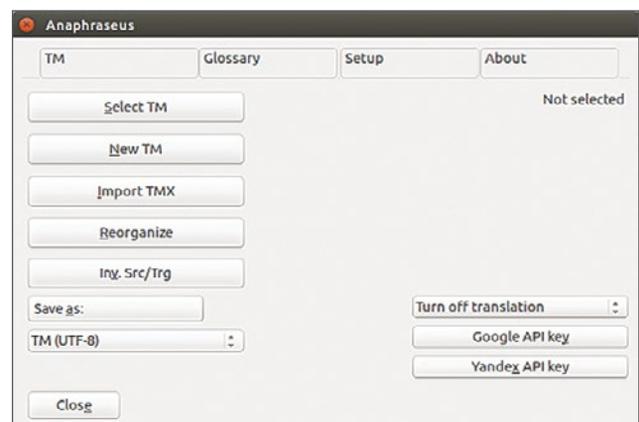
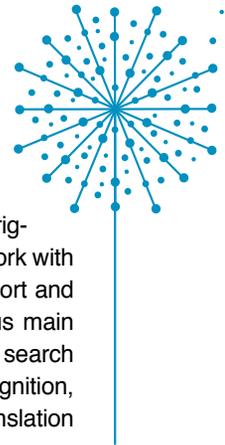
5. Export as Images

The Export as Images extension [8] adds a *File* menu entry *export as Images...* in Impress and Draw [9], to export all slides or pages as images in JPG, PNG, GIF, BMP, and TIFF format, and allows users to choose a file name for exported images, the image size, and other parameters.



6. Anaphraseus

Anaphraseus [10] is a CAT (Computer-Aided Translation) tool for creating, managing, and using bilingual Translation Memories. Anaphraseus is a LibreOffice macro set available as an extension or a standalone document. Originally, Anaphraseus was developed to work with the Wordfast format, but it can also export and import files in TMX format. Anaphraseus main features are: text segmentation, fuzzy search in Translation Memory, terminology recognition, and TMX Export/Import (OmegaT translation memory format).



Links

- [1] LibreOffice extensions: <http://extensions.libreoffice.org/>
- [2] MultiFormatSave: <http://extensions.libreoffice.org/extension-center/multisave-1>
- [3] Open Document Format: <http://www.opendocumentformat.org/>
- [4] Alternative dialog Find & Replace for Writer (AltSearch): <http://extensions.libreoffice.org/extension-center/alternative-dialog-find-replace-for-writer>
- [5] Pepito Cleaner: http://pepitoweb.altervista.org/pepito_cleaner/index.php
- [6] ImpressRunner: <http://extensions.libreoffice.org/extension-center/impressrunner>
- [7] Impress: <https://www.libreoffice.org/discover/impress/>
- [8] Export as Images: <http://extensions.libreoffice.org/extension-center/export-as-images>
- [9] Draw: <https://www.libreoffice.org/discover/draw/>
- [10] Anaphraseus: <http://anaphraseus.sourceforge.net/>

Author

Italo Vignoli is a founding member of The Document Foundation. He handles the organization's PR and media relations, coordinates its certification program, and is an international spokesman for the project. Italo has supervised some of the largest LibreOffice migration projects in Italy, and is a LibreOffice trainer.

5 favorite open source Django packages

BY JEFF TRIPLETT, JACOB KAPLAN-MOSS, AND FRANK WILES

Django [1] is built around the concept of reusable apps [2]: self-contained packages that provide re-usable features. You can build your site by composing these reusable apps, together with your own site-specific code. There's a rich and varied ecosystem of reusable apps available for your use—PyPI lists more than 8,000 Django apps [3]—but how do you know which ones are best?

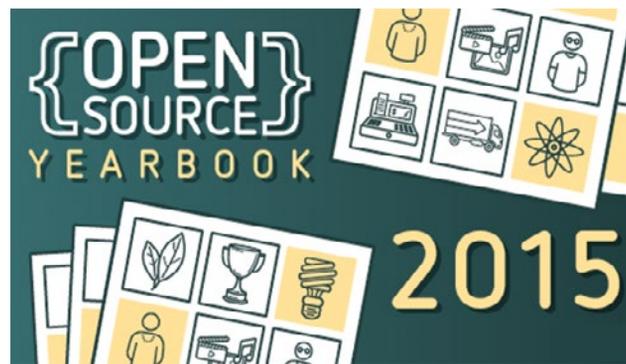
To help focus your app search, we've put together this list of our five favorites. They are:

- **Cookiecutter:** the best way to start a new Django site. [4]
- **Whitenoise:** the best static asset server. [5]
- **Django Rest Framework:** the best way to write REST APIs with Django. [6]
- **Wagtail:** the best Django-based content-management system. [7]
- **django-allauth:** the best way to provide “social login” (e.g., Twitter, Facebook, GitHub, etc). [8]

We also recommend you check out Django Packages [9], a directory of reusable Django apps. Django Packages organizes Django apps into “grids” that allow you to compare similar packages and chose between them. You can see which features are offered by each package, as well as usage statistics. (For example, the grid for REST tools [10] might help you understand why we recommend Django REST Framework.)

Why should you trust us?

We've been using Django for longer than almost anyone. Two of us (Frank and Jacob) worked at the Lawrence Journal-World [11] (birthplace of Django) before Django was released (and in fact helped make the open source release happen). We've all spent the past eight years running a consultancy that advises companies on how best to use Django.



So, we've seen the entire history of the Django project and community, and we've seen popular packages come and go. Between the three of us, we've probably tried at least half of these

8,000 apps personally, or we know someone who has. We have a strong understanding of what makes an app solid and reliable, and we have a good understanding of what gives these things staying power.

Best way to start a new Django site: Cookiecutter

Starting off a new project or app is always a bit of a pain. You can use Django's built in `startproject`` but if you're like us, you're particular in how you do things. Cookiecutter solves this by giving you a quick and easy way to define project or app templates that can be easily reused. A quick example, just `pip install cookiecutter`` and then run this from the command line:

```
$ cookiecutter https://github.com/marcofucci/
  cookiecutter-simple-django
```

You'll immediately start getting prompted for quick answers, such as the name of your project, repo, author name, email, and a few other bits of configuration. These are used to help fill out the project details. We picked the ever so original `'foo'` to be our repo name. So cookiecutter created a simple Django project in the subdirectory `'foo'`.

If you poke around in the `'foo'` project a bit, you'll see the other bits of configuration you were prompted for have been templated into the files themselves along with sub-directories as necessary. This “template” is all defined at the GitHub repo URL we used as the only argument when we called `'cookiecutter``. This example used a remote GitHub repo as the template; however, note that you can use local

file system directories as well, which is perfect for non-reusable scenarios.

We mention cookiecutter as a great Django package, but honestly it's useful for plain Python or even non-Python-related purposes. Being able to lay things out exactly as you like in an easily repeatable way makes cookiecutter a great tool for keeping your workflow DRY.

Best static asset server: Whitenoise

For many years, serving your site's static assets—images, JavaScript, CSS—was a pain. The built-in `django.views.static.serve` view [12] is, as the documentation states, “not hardened for production use and should be used only as a development aid.” But serving media from a “real” web server, such as NGINX or out of a CDN, can be difficult to set up. Whitenoise cleanly solves this problem. It's as easy to set up as the development-only static server, and is hardened and optimized for production. Setup is simple:

1. Make sure you're using Django's `contrib.staticfiles` app [13], and that you've correctly set `'STATIC_ROOT'` in your settings file.
2. Enable Whitenoise in your `'wsgi.py'` file:

```
from django.core.wsgi import get_wsgi_application
from whitenoise.django import DjangoWhiteNoise
application = get_wsgi_application()
application = DjangoWhiteNoise(application)
```

That's really all it takes! For large applications, you'll likely want to use a dedicated media server and/or a CDN, but for most small- or medium-sized Django sites, Whitenoise is more than powerful enough.

For more information on Whitenoise, check out the documentation [14].

Best Tool for REST APIs: Django REST Framework

REST APIs are quickly becoming a standard feature of modern web applications. An API is really simply talking in JSON rather than HTML, and of course you can do this with just Django. You can craft your own views that set the proper content types and return data in JSON rather than templated HTML responses. This is exactly what many people did before API frameworks such as Django Rest Framework (a.k.a., DRF) [6] were released.

Building a REST API with DRF is similar to working with Django's Class Based Views if you're familiar with them, except these are specifically designed and targeted around an API use case. Quite a bit of code is involved in your average API setup, so instead of a code sample to get you excited, we'll highlight some of DRF's features that make your life easier:

- Automatic browseable API, which makes development and manual testing a breeze. Click around in the DRF demo example [15]. You can view API responses and support POST/PUT/DELETE type operations without having to do anything yourself.

- Easy integration of authentication styles, such as OAuth, Basic Auth, or API Tokens.
- Simple permission system for fine-grained control of which users can use which API endpoints and/or actions.
- Built-in rate limiting.
- Nearly automatic API documentation when combined with `django-rest-swagger` [16].
- Extensive ecosystem of third-party libraries.

Although you could certainly build an API without DRF, we can't fathom a reason why you would start off down that path. Even if you don't use all of DRF's features, building up your own API views from their solid base view classes is a huge win in terms of safety, consistency of your API, and development velocity. If you aren't using DRF already, you should set aside some time to check it out.

Best Django-based CMS: Wagtail

Wagtail is the current darling of the Django CMS world and with good reason. Like most CMS systems, it gives you flexibility to define different types of pages and their content via simple Django models. This takes you from zero to a basically working system in hours, not days. To give you a quick example, to define a Staff page type for people at your company can be as simple as:

```
from wagtail.wagtailcore.models import Page
from wagtail.wagtailcore.fields import RichTextField
from wagtail.wagtailadmin.edit_handlers import
    FieldPanel, MultiFieldPanel
from wagtail.wagtailimages.edit_handlers import
    ImageChooserPanel
class StaffPage(Page):
    name = models.CharField(max_length=100)
    hire_date = models.DateField()
    bio = models.RichTextField()
    email = models.EmailField()
    headshot = models.ForeignKey('wagtailimages.Image',
        null=True, blank=True)
    content_panels = Page.content_panels + [
        FieldPanel('name'),
        FieldPanel('hire_date'),
        FieldPanel('email'),
        FieldPanel('bio', classname="full"),
        ImageChoosePanel('headshot'),
    ]
```

The real appeal of Wagtail, however, is in its easy-to-use modern admin interface and flexibility. You can control which types of pages are allowed in different areas of the site, add additional complex logic to your pages, and get standard moderation/approval workflows right out of the box. With most CMS systems, at some point you run into a wall you can work around. With Wagtail, we have yet to find a wall we couldn't easily find a door through to make it do exactly what we want in an easy and maintainable

way. If you're interested, we wrote up a deeper dive into Wagtail [17].

Best Social Login Tool: django-allauth

django-allauth [18] is a reusable Django application that solves your registration and authentication needs. Whether you need a local or social registration system, django-allauth has you covered.

The project supports multiple authentication schemes, such as user name or email address. Once a user has signed up, multiple strategies are supported for account verification ranging from none to email verification. Multiple social and email accounts are also supported. Pluggable signup forms are also supported which allows asking additional questions during registration.

django-allauth supports more than 20 authentication providers, including Facebook, GitHub, Google, and Twitter. If you have an account on a social website that's not supported, then it's most supported through a third-party add-on. The project supports writing custom backends, which allows custom authentication systems

to work, too. This is nice for anyone with custom authentication needs.

django-allauth is easy to setup and has extensive documentation. The project is also well tested so you know that everything actually works.

Authors

Jeff Triplett has worked at the Lawrence Journal-World in Lawrence, KS, where Django was invented. He now works at Revolution Systems (Revsys) as a developer and a consultant. Jeff is a co-founder of the Django Events Foundation North America (DEFNA) and Conference Chair for DjangoCon US 2015 and 2016.

Jacob Kaplan-Moss is a core contributor to Django and works at Heroku as Director of Security.

Frank Wiles is President and Founder of Revolution Systems, a open source consultancy specializing in performant and large scale Python, Django, PostgreSQL, and related web infrastructure. He is a frequent author and speaker on open source topics. Frank's personal homepage is frankwiles.com.

Links

- [1] Django: <https://www.djangoproject.com/>
- [2] How to write reusable apps: <https://docs.djangoproject.com/en/1.8/intro/reusable-apps/>
- [3] PyPi packages: <https://pypi.python.org/pypi?:action=browse&c=523>
- [4] Cookiecutter: <https://github.com/audreyr/cookiecutter>
- [5] Using WhiteNoise with any WSGI application: <https://github.com/audreyr/cookiecutter>
- [6] Django Rest Framework: <http://www.django-rest-framework.org/>
- [7] Wagtail: <https://wagtail.io/>
- [8] django-allauth: <http://www.intenct.nl/projects/django-allauth/>
- [9] Django Packages: <https://www.djangopackages.com/>
- [10] Grid for REST tools: <https://www.djangopackages.com/grids/g/rest/>
- [11] Lawrence-born Django celebrates 10th anniversary: <http://www2.ljworld.com/news/2015/jul/09/happy-birthday-django/>
- [12] django.views.static.serve view: <https://docs.djangoproject.com/en/1.8/ref/views/#django.views.static.serve>
- [13] staticfiles app: <https://docs.djangoproject.com/en/1.8/ref/contrib/staticfiles/>
- [14] Whitenoise documentation: <http://whitenoise.evans.io/en/latest/index.html>
- [15] DRF demo example: <http://restframework.herokuapp.com/>
- [16] Django REST Swagger: <http://django-rest-swagger.readthedocs.org/en/latest/index.html>
- [17] Get ready for Wagtail, the best Django CMS yet: <https://opensource.com/business/15/5/wagtail-cms>
- [18] django-allauth documentation: <http://django-allauth.readthedocs.org/en/latest/>

Facebook's top 5 open source projects of 2015

BY CHRISTINE ABERNATHY

Facebook believes in the power of open source. When a community gathers to work on code, there are a host of benefits. Fresh eyes point out problems and we arrive at solutions faster. Together we tackle the challenges we're facing, innovation accelerates, and the community stretches the limitations of existing technology.

Of course, a successful open source program depends on a strong, collaborative community. As the end of the year approaches, we wanted to reflect on Facebook's top five open source projects in 2015, measured by community activity and impact.

HipHop Virtual Machine (HHVM)

HHVM [1] is our virtual machine and web server that we open sourced in 2013, building on the HPHPC compiler we released in 2010. In the past year alone, we've seen a 29% increase in the number of commits and a 30% increase in the number of forks.

HHVM is most commonly run as a standalone server, replacing both Apache and mod_php, designed to execute programs written in Hack and PHP. It uses a just-in-time compilation approach to achieve superior performance, while maintaining the flexibility that PHP developers are accustomed to. We've reached great milestones in 2015:

1. We made new Async features [2] available by default, including Async MySQL [3] and MCRouter [4] (memcached) support.
2. In December we announced [5] support for all major PHP 7 features at the same time that the language itself was released, and we released our next generation of user documentation.
3. Box announced [6] HHVM as the exclusive engine that serves its PHP codebase.
4. Etsy migrated to HHVM in April, which helped the company address a variety of challenges associated with building mobile products at the scale needed.

React

Facebook open sourced React [7] in May 2013, and in the past year we've continued to see strong collaboration in the



community, including a 75% increase in the number of commits and a 198% increase in the number of forks. React is Facebook's JavaScript library for building user interfaces, and is being used by many companies because it takes a different approach to building applications: React allows you to break the application down into separate components that are decoupled so that the various components can be maintained and iterated on independently.

This year we had two major releases, launched React Native, announced new developer tools [8], and saw more companies—including Netflix [9] and WordPress [10]—use React to build their products.

Presto

Presto [11] is our distributed SQL engine for running interactive analytic queries against data sources of all sizes, ranging from gigabytes to petabytes. We created Presto to help us analyze data faster because our data volume grew and the pace of our product cycle increased.

Since making Presto available to others in November 2013, we've seen a lot of growth, adoption, and support for it, including a 48% increase in the number of commits and a 99% increase in the number of forks in the past year. Companies like Airbnb [12], Dropbox, and Netflix [13] use Presto as their interactive querying engine. We also see growing adoption all over the world, including by Gree [14], a Japanese social media game development company, and Chinese e-commerce company JD.com.

In 2015, Teradata announced [15] plans to join the Presto community, with a focus on enhancing enterprise features and providing support. This emphasizes the level of trust the community has in Presto's ability to be an integral part of the data infrastructure stack. Additionally, Amazon Web Services (AWS) supports Presto as a first-class offering in its EMR service [16], with many production users—including Nasdaq and leading business intelligence tool vendor MicroStrategy—supporting Presto in its flagship MicroStrategy 10 product.

RocksDB

We open sourced RocksDB [17], an embeddable, persistent key-value store for fast storage, in November 2013. Aside from the impressive 52% increase in the number of commits and the 57% increase in the number of forks for this project in the past year, the reason this particular project has resonated so well in the open source community is that the embedded database helps provide a way to work around slow query response time due to network latency, and it is flexible enough to be customized for various emerging hardware trends.

RocksDB powers critical services at companies such as LinkedIn and Yahoo, and a key focus for us in 2015 was to bring the RocksDB storage engine to general-purpose databases, starting with MongoDB. Similar to Teradata's commercial support for Presto, another milestone for RocksDB in 2015 was the announcement of enterprise-level support by Percona's [18] data performance experts.

React Native

React Native [19], one of our newest open source projects, was made available in March 2015. React Native lets engineers use the same React methodology and tools to rapidly build native applications for mobile devices. In addition to developing these tools internally, Facebook collaborates with the open source community to improve the experience for developers worldwide. In its first year, React Native has become the second most popular Facebook open source project, with more than 23,000 followers in GitHub. It was used internally to build the Facebook Ads app for both iOS and Android, resulting in an 85% code reuse by developers whose core competency was JavaScript. The paradigm shift in mobile development that React Native brings to the table makes this one a key highlight of 2015.

Overall, we still have a lot of work to do, but we're proud of what we've been able to accomplish as a community. We want to thank everyone who dedicated time to these projects and helped us make 2015 a great year!

Links

- [1] HipHop Virtual Machine (HHVM): <https://github.com/facebook/hhvm>
- [2] Async introduction: <https://docs.hhvm.com/hack/async/introduction>
- [3] Async MySQL extension: <https://docs.hhvm.com/hack/async/extensions>
- [4] MCRouter extension: <https://docs.hhvm.com/hack/async/extensions#mcrouter>
- [5] Improved user documentation announcement: <http://hhvm.com/blog/10925/improved-user-documentation>
- [6] Box's HHVM migration: <https://code.facebook.com/posts/1607907626123431/under-the-hood-box-s-hhvm-migration/>
- [7] React: <https://github.com/facebook/react>
- [8] New React Developer Tools: <https://facebook.github.io/react/blog/2015/09/02/new-react-developer-tools.html>
- [9] Netflix Likes React: <http://techblog.netflix.com/2015/01/netflix-likes-react.html>
- [10] The Story Behind the New WordPress.com: <https://developer.wordpress.com/2015/11/23/the-story-behind-the-new-wordpress-com/>
- [11] Presto: <https://github.com/facebook/presto>
- [12] Introducing Airpal: <http://nerds.airbnb.com/airpal/>
- [13] Using Presto in our Big Data Platform on AWS: <http://techblog.netflix.com/2014/10/using-presto-in-our-big-data-platform.html>
- [14] A year of using Presto in production: <http://labs.gree.jp/blog/2014/12/12838/>
- [15] Open-Source Presto to the Enterprise: <http://blogs.teradata.com/data-points/bringing-open-source-presto-enterprise/>
- [16] Presto on Amazon EMR: <https://aws.amazon.com/elasticmapreduce/details/presto/>
- [17] RocksDB: <https://github.com/facebook/rocksdb>
- [18] Percona: <https://www.percona.com/>
- [19] React Native: <https://github.com/facebook/react-native>

Author

Christine Abernathy is a developer advocate on the open source team at Facebook. She's held previous developer advocacy roles on both Parse and Facebook Platform. Prior to working at Facebook, Christine lead engineering at MShift, a mobile banking software provider delivering iOS apps and mobile browser-based products.

10 cool tools from the Docker community

BY MANO MARKS

Looking back at 2015, there have been many projects created by the Docker community that have advanced the developer experience. Although choosing among all the great contributions is hard, here are 10 “cool tools” that you should be using if you are looking for ways to expand your knowledge and use of Docker.

1. Container Migration Tool (CMT)

CMT [1] was a winning entry [2] at the Docker Global Hack Day #3. The Container Migration team [3] drew inspiration from a DockerCon talk in which Michael Crosby (@crosby-michael) and Arnaud Porterie (@icecrime) migrated a Quake 3 container around the world, demonstrating container migration while maintaining a TCP connection. The CMT project created an external command-line tool that can be either used with Docker or runC [4] to help “live migrate” containers between different hosts by performing pre-migration validations and allowing it to auto-discover suitable target hosts.

2. Dockercraft

We had to add in a fun one—Dockercraft [5]! Lots of Docker users run custom Minecraft [6] servers in containers. But Dockercraft is a Minecraft client to visualize and manage Docker containers. With the flick of a switch, a container turns on or off. And with the press of a button, you can destroy one. Dockercraft is a fun project—that is surprisingly addictive—from Docker engineers Adrien Duermael and Gaetan de Villele.

3. Docker Label Inspector

The Docker Label Inspector [7] tool helps ensure that developers are providing Docker images with the metadata con-

tainers required when distributed across the Internet. Specifically, this tool enables developers to use Docker labels [8] to create metadata within the domain of container technology, to check labels against official label schema and to validate against provided JSON schema.

4. Dvol

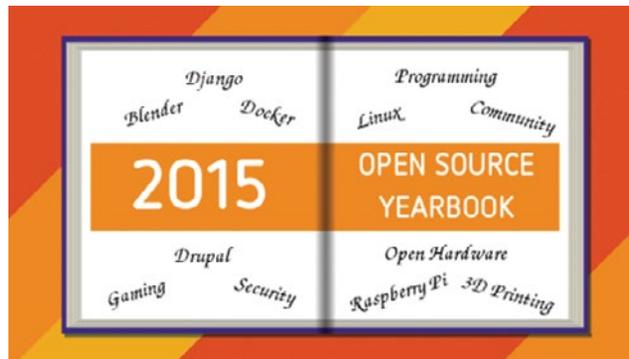
Dvol [9] enables version control for your development databases in Docker. Dvol lets you commit, reset, and branch the containerized databases running on your laptop, so you can easily save a particular state and come back to it later. Dvol can also integrate with Docker Compose to spin up reproducible microservices environments on your laptop.

5. IPVS Daemon GORB

Presented at DockerCon EU, IP Virtual Server (IPVS) for Docker containers enables production-level load balancing and request routing using open source IPVS, which has been part of the Linux kernel for more than a decade. It supports TCP, SCTP, and UDP and can achieve fast speeds, often within five percent of direct connection speeds. Other features include NAT, tunneling, and direct routing. To make IPVS easier to use, the Go Routing and Balancing (GORB) daemon [10] was created as a REST API inside a Docker container to provide IPVS routing for Docker.

6. Libnetwork

Libnetwork [11] combines networking code from both libcontainer and Docker Engine to create a multi-platform library for networking containers. The goal of libnetwork is to deliver



a robust Container Network Model that provides a consistent programming interface and the required network abstractions for applications. There are many networking solutions available to suit a broad range of use-cases. Libnetwork uses a driver/plugin model to support all of these solutions, while abstracting the complexity of the driver implementations by exposing a simple and consistent Network Model to users.

7. The Raspberry Pi Challenge

In the DockerCon closing keynote [12], Dieter Reuter from Hypriot presented a demo running 500 Docker containers on a Raspberry Pi 2 device. Convinced that this number of containers could be at least doubled, Dieter then challenged [13] the Docker community to beat his personal record. As part of his project, Dieter Reuter demonstrated how to get started with Docker on Raspberry Pi and how to scale the number number of web servers running in containers that could reside on a single Raspberry Pi 2. The current record is more than 2,500 web servers running in containers on a single Raspberry Pi 2.

8. Scaling Spark with Zoe analytics

The Zoe open source user-facing tool [14] ties together Spark [15], a data-intensive framework for big data computation, and Docker Swarm [16]. Zoe can execute long-running Spark jobs, but also Scala or iPython interactive notebooks and streaming applications, covering the full Spark development cycle. When a computation is finished, resources are automatically freed and available for other uses, because all processes are run in Docker containers. This tooling can enable application scheduling on top of Swarm and optimized container placement.

9. Unikernel demo source code

First unveiled as a cool hack at DockerCon EU (Unikernels, meet Docker! [17]), this demo [18] showed how unikernels can be treated as any other container. In this demo, Docker was used to build a unikernel microservice and then followed up by deploying a real web application with database, webserver, and PHP code, all running as distinct unikernel microservices built using Rump Kernels [19]. Docker managed the unikernels just like Linux containers, but without needing to deploy a traditional operating system. Apart from the MySQL, NGINX, and PHP with Nibbleblog unikernels shown in the demo, this repository also contains some examples of how to get started.

10. Wagl, DNS service discovery for Swarm

Wagl [20] is a DNS server that allows microservices running as containers on a distributed Docker Swarm [16] cluster to find and talk to each other. Wagl is minimalist and works as a drop-in container in your cluster to provide DNS-based service discovery and simple load balancing by rotating a list of IP addresses in DNS records.

Links

- [1] Container migration tool: <https://github.com/marcosniels/cmt>
- [2] Container migration tool (CMT) (YouTube video): https://youtu.be/pwf0-_cs6U4
- [3] Participating in Docker global hackday #3 (blog post): <http://blog.mantika.io/GHD/>
- [4] runC: <https://runc.io/>
- [5] Dockercraft: <https://github.com/docker/dockercraft>
- [6] Minecraft: <https://minecraft.net/>
- [7] Docker Label Inspector: <https://github.com/garethr/docker-label-inspector>
- [8] Docker labels: <https://docs.docker.com/engine/userguide/labels-custom-metadata/>
- [9] dvol: <https://github.com/ClusterHQ/dvol>
- [10] Go Routing and Balancing (GORB): <https://github.com/kobolog/gorb>
- [11] libnetwork: <https://github.com/docker/libnetwork>
- [12] DockerCon 2015 Videos: Day 2 Closing Keynote – What’s Next for Docker?: <http://blog.docker.com/2015/07/dockercon-2015-videos-day-2-closing-keynote/>
- [13] Raspberry Pi Challenge: <https://github.com/hypriot/rpi-kernel>
- [14] Zoe: <https://github.com/DistributedSystemsGroup/zoe-docker-images>
- [15] Apache Spark: <http://spark.apache.org/>
- [16] Docker Swarm: <https://docs.docker.com/swarm/>
- [17] Unikernels, meet Docker!: <http://unikernel.org/blog/2015/unikernels-meet-docker/>
- [18] Contain Your Unikernels!: <http://unikernel.org/blog/2015/contain-your-unikernels/>
- [19] Rump Kernels: <http://rumpkernel.org/>
- [20] wagl: <https://github.com/ahmetalpbalkan/wagl>

Author

Mano Marks is Director of Developer Relations at Docker, Inc.

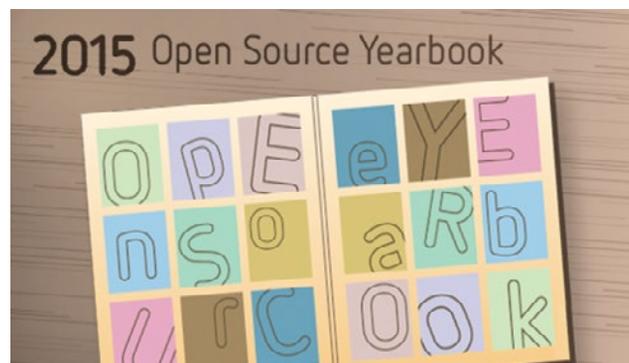
5 handy Drupal modules

BY MICHAEL E. MEYERS

Drupal [1], one of the largest open source projects in the world, is a content management system and application framework that powers millions of websites, web services, and mobile applications. Individuals and organizations in every sector use Drupal for everything from simple blogs and micro-sites, to complex intranets and private internal applications, to some of the largest sites on the web, including several top 100 properties.

More than 3,000 people contribute to the core Drupal codebase that provides a base level of functionality with a modular architecture. Over 10,000 developers contribute and maintain more than 25,000 contributed open source modules that modify and extend the core capabilities, changing behavior, adding new features, and integrating Drupal with other systems. Users can add their own modules, as well as third-party PHP libraries, and leverage a lot of existing functionality, so they can focus on their unique needs, branding and design, and business logic to get up and running quickly. In addition to its highly extensible codebase, Drupal lets users who are not programmers manage and control a lot of how the platform and their site(s) run and operate through point and click end-user interfaces. This enables more people to contribute to an organization's online presence, and site updates happen significantly faster because Drupal users aren't reliant on expensive and limited development resources, or updating code.

The newest major version of Drupal, version 8, came out in November 2015, so let's look at a few of my favorite contributed modules. Whether you're running one Drupal website, or thousands, here are five modules that will empower your entire team and drive the success of your application:



Views

Views [2], one of the most popular contributed Drupal modules, enables site builders, content editors, and other non technical users to surface information from within Drupal—or other systems through Drupal—and display and visualize that information

through a point and click interface. With Views you can create a new page that lists all recent updates to your site, a sidebar block that shows the five most recent comments on your website, a slideshow of your most popular images of the week, and much more. Views can be dynamic, end users can have access to control resorting or filtering the information (by date or other parameters), and privileged users can be granted the ability to perform bulk actions on Views. For example, on a site with moderated comments, admins can create a page where users can quickly select all the comments they want to approve or reject, and do so in bulk to save a lot of time. As of Drupal 8, Views is now a part of Drupal Core and has expanded capabilities so users can build custom admin pages, export any View as JSON or XML, and build most of a site—or the back-end for a mobile application—without writing a line of code.

BigPipe

Site speed is a critical goal for all websites and companies. The BigPipe [3] concept was created by Facebook and was a key component in doubling the speed of their site page load times. BigPipe is a contributed module for Drupal 8 that leverages Facebook's methodology and delivers web pages from servers to clients in a way that allows dramatically improved perceived load time as well as the actual page load time of sites. With BigPipe the server will send an initial response or page with the cacheable components

and place holders for any dynamic or uncacheable parts, and then stream in the dynamic/personalized pieces to replace the placeholders.

Rules

Rules [4] is a contributed module that enables site users to create Event, Condition, Action (ECA) rules. Site builders can create and modify fairly complex business logic without writing any code, and developers have tremendous flexibility to create complex rules, and to extend and modify the behavior of the Rules module itself. For example, when a new comment is posted to a page (an event), a site builder could create a rule that sends an email notifying the author of the page (the action) of the new comment so they can review and respond quickly. Optionally, users can add a condition to send the email only if the comment author is not the author of the page (so the page author doesn't get an email every time they reply to a comment). Many rules can also be triggered manually via the click of a button (e.g., by a site admin) or, via the Rules API, actions can be set to run immediately or scheduled.

Features

Most organizations write code and make changes to their web sites in one or more development environments, then integrate and test these changes in QA or staging environments, and finally release approved changes to their production or live environment. Furthermore, few organizations have a single website; nowadays even individuals run many websites, and large organizations can have more than 10,000 sites. Features [5] is a contributed module that enables you to import and export configuration and or code as a package or collection that you can move across Drupal sites. For example, site builders could create a rule using the Rules module and then using Features, export the rule so that it can be imported into the many other environments and web sites you have.

Lightning

Lightning [6] isn't actually a module; it's a collection of Drupal modules provided as a distribution. Drupal distributions are a way to package up and distribute a Drupal site (modules, themes, configurations, automated QA tests, content, etc.). Launching a new website can be done quickly by deploying a distribution for a specific use case, and then adding any customizations required. The Lightning distribution is meant to be a framework that you build on top of—it provides you with pre-configured popular modules that work well together for common use cases, with the ability to disable what you don't need, and launch a new site on Drupal using best practices for the most common needs. Lightning helps new users advance through the learning curve much faster, and enables all users to rapidly deploy new sites.

What are your favorite Drupal modules? From the obscure problem solvers, to the ones you simply can't live without, share your story and send Drupal article proposals to open@opensource.com.

Disclosure: In partnership with Acquia, OpenSource.com and Red Hat are Drupal users and open source contributors to the Drupal project.

Links

- [1] Drupal: <https://www.drupal.org/>
- [2] Views: <https://www.drupal.org/project/views>
- [3] BigPipe: https://www.drupal.org/project/big_pipe
- [4] Rules: <https://www.drupal.org/project/rules>
- [5] Features: <https://www.drupal.org/project/features>
- [6] Lightning: <https://www.drupal.org/project/lightning>

Author

Michael E. Meyers is the VP of Developer Relations at Acquia. He's been driving innovation in the Drupal community for more than a decade. Prior to joining Acquia, Michael was the CTO of Examiner.com, a top 50 site, and the leading contributor to Drupal 7. Follow Michael on Twitter [@MichaelEMeyers](https://twitter.com/MichaelEMeyers).

Top 5 open source frameworks every application developer should know

BY JOHN ESPOSITO

Software has been eating the world for far longer than four years [1]. But developers think of software a little differently. Insofar as we're solving real-world problems [2], we're not thinking mainly of the metal [3]. And as the problems get bigger and the solutions more complex, pragmatic (and not-too-leaky) abstractions become more important than ever.

The upshot: From a productivity-oriented developer's point of view, *frameworks* are eating the world. But which ones are eating how much of which parts of the world?

Given the insane variety of superb open source frameworks available, I picked our top 5 open source frameworks of 2015 not from a single ranked order, but from all levels of the stack. (For front-ends, I focused on the web and, still more narrowly, true client-side frameworks—simply because browsers and mobile devices are growing increasingly capable, and because SPAs [single page applications] and the like avoid sending data over the wire unnecessarily.)

1. Presentation: Bootstrap

Let's start at the top of the stack: the presentation layer, the stuff developers and us folks both touch. Here the clear winner remains Bootstrap [4]. The forecast looks outstanding, leaving old alternatives, such as Foundation [5], and new kids, such as Material Design Lite [6], in the dust [7]. Bootstrap dominates usage trends on BuiltWith [8], and on GitHub remains easily the most starred and most forked framework of all time.

And Bootstrap is still under active development. In August 2015, Bootstrap celebrated its fourth birthday with version 4 alpha release [9], a major simplification and expansion of an already powerful feature set. The gist of the update: Everything got more programmatic. Bootstrap moved from Less to Sass, centralized all HTML resets in a single module, tossed a load of style customizations directly into Sass variables, and ES6-ified all JavaScript plugins.



The team even launched an official Bootstrap Themes marketplace [10], complementing an already massive theme ecosystem.

2. Web MVC: AngularJS

As the web platform continues to mature, developers enjoy increasingly well-crafted abstraction-distance from the still-markup-colored DOM. The work begun by XMLHttpRequest reaches its zenith in modern Single-Page Applications (SPAs), and the most popular SPA framework by far is AngularJS [11].

What's so special about AngularJS? In a word: directives [12]. Just a little ng- can bring a (static, markup) tag to (dynamic, JavaScript-executing) life. (Dependency injection is pretty neat, too, and like many of Angular's features aims to simplify maintenance and abstract more fully from the DOM.)

The basic principle is just the perfectly reasonable separation of declarative view from imperative domain logic, familiar to anyone who has scanned a POM file or wrestled with an ORM (and hey, some of us used to enjoy XAML, too). This is thrilling, liberating, and a little weird all at once—giving HTML power it kinda really shouldn't have.

Perhaps a bit sadly, the most aggressive concept behind AngularJS—two-way data binding, which effortlessly keeps views and models synced—is going away in Angular2 [13], which is “very close” to beta release [14]. So a bit of the magic will disappear, but so will some massive performance headaches at scale and some hair-pulling tough debugging (just think about two-way for a moment and feel the cliff drop out from under your feet)—a trade off that grows more valuable as page size and SPA complexity balloon.

3. Enterprise Java: Spring Boot

What’s so great about Java? Fast, mature, comprehensive class library, gigantic ecosystem, write-once-run-everywhere, active community—but not painless bootstrapping. Even hard-core Java developers resort to Ruby or Python to write quick one-off programs (admit it). And yet Java continues to dominate the enterprise for those other reasons listed above.

Enter Spring Boot, the boilerplate evaporator—a framework that lets you fit a working Spring application in a single tweet:



No unpleasant XML config, no sloppy generated code. How is this possible? Simple: Spring Boot has some pretty strong opinions. Read that tweet above and suddenly it all makes sense! When you realize that the framework automatically spins up an embedded servlet container to handle incoming requests on port 8080—a decision you didn’t explicitly configure it to make, but a conventional (and fire-wall-friendly) call.

How hot is Spring Boot? It’s by far the most forked [15] and most downloaded Spring project (not counting the master framework itself). And in 2015, for the first time, Google logged more searches for *spring boot* than for *spring framework* [16].

4. Data processing: Apache Spark

Once upon a time (in 2004), Google developed a programming model (MapReduce [17]) that generalized many

distributed batch processing job structures, then wrote a famous paper [18] about it; then some Yahoo folks wrote a Java framework (Hadoop [19]) that implemented MapReduce and a distributed file system [20] to simplify data access for MapReduce tasks.

For nearly a decade Hadoop dominated the “Big Data” framework ecosystem, despite the limited problem space addressed (optimally) batch processing—partly because business and scientific users were accustomed to batch analysis of large datasets anyway. But not all large datasets are optimally processed in batches. In particular, streaming data (such as sensor inputs) and data analyzed iteratively (like machine learning algorithms love to do) don’t like batch processing. So dozens of new Big Data frameworks were born, new programming models, application architectures [21], and data stores [22] gained traction (including a new cluster management system [23], decoupled from MapReduce, for Hadoop itself).

But of all these new frameworks, Apache Spark [24] (developed at Berkeley’s AMPLab) is the easy-choice 2015 standout. Surveys (DZone report [25]; Databricks infographic [26]; Typesafe report [27]) show huge growth in Spark adoption. GitHub commits have been growing linearly since 2013, and Google Trends show exponential (yes, literally) growth in searches over 2015 [28].

So Spark is popular. But what does it do? Well, very fast batch processing; but this depends on one killer feature that allows for vastly more programming models than Hadoop. Spark makes data available in Resilient Distributed Datasets (RDDs) that remain in memory on multiple nodes after processing, but *without replication* (by storing info on how to recreate; compare to CQRS [29], pragmatism [30], Kolmogorov complexity [31]). This (obviously) lets algorithms to iterate without reloading from a (s)lower rung in the distributed memory hierarchy. And this means that batch processing need no longer suffer the indignity of the “long stroke” of Nathan Marz’s lambda architecture. RDDs even allow Spark to simulate true (push) stream processing by running small batch jobs fast enough to keep latency within “effective streaming” bounds for many applications.

5. Delivery: Docker

Okay, so Docker [33] isn’t a “framework” in the sense of “code library, generously defined, that imposes a specific set of conventions to solve large and recurrent problem sets”. But if frameworks are just things that let you write code at a more suitable level of abstraction, then Docker is a framework extraordinaire. (Let’s call it an *exoskeletal framework*, just to mix metaphors confusingly.) And it would feel funny to name “top 2015 anything for developers” without including Docker on the list.

Why is Docker great? First, why are containers (earlier: FreeBSD Jail, Solaris Zones, OpenVZ, LXC) great? Simple: isolation without a full operating system; or, safety and convenience of a VM with far less overhead. But isolation

10 helpful tools for a sys admin's toolbox

BY BEN COTTON

Sys admins, no matter what platforms they work on, are awash in great open source software tools. In this article, we highlight well-known—and not-so-well-known—tools that have released new versions in 2015.

BlackBox

Managing secrets is tough, especially when the secrets need to end up on a server. That's why Stack Exchange [1] developed the BlackBox [2] utility. BlackBox uses GNU Privacy Guard [3] (GPG) to securely encrypt secrets, such as passwords and private keys, and store them in a version control system. BlackBox also provides a mechanism for configuration management tools (specifically Puppet [4]) to extract the secrets when needed.

The plugin architecture of BlackBox makes it easy to extend. Although BlackBox was originally designed to work with git [5] repos, support for Mercurial [6], Subversion [7], and Perforce [8] has been added. BlackBox supports Linux, OS X, and Windows (via Cygwin [9]), so it can be used in mixed environments.

KeePass

Managing secrets gets a little easier when they don't have to go anywhere. Years ago, I started using KeePass [10] because it had been ported to the Maemo [11] platform. If abandoned mobile operating systems aren't your thing, KeePass is available for Linux, OS X, Windows, Android, and iOS.

When I was a newly minted sys admin, I had a notebook of rarely used passwords that I kept in a locked safe. KeePass is like that, except easier to back up and sync across locations. KeePass entries store URLs, user names, passwords, and free text. Entries can be sorted into groups and given custom icons for ease of use. Recent releases have focused on improving performance and UX.

ownCloud

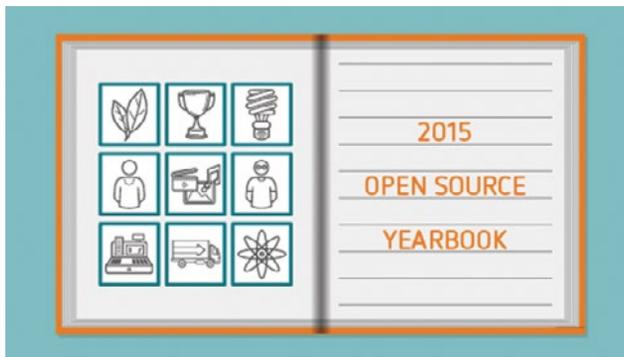
Need a way to synchronize your KeePass database? ownCloud's recent 8.2 release brings improved design and ease of administration. For those who can't or won't use third-party services, ownCloud [12] provides a great way to share documents, photos, calendars, and other data securely. A community-developed application repository allows admins to extend the capabilities in a wide variety of ways, including receiving SMS messages from Android phones. If you need to give your users (or yourself) the ability to collaborate while keeping control over your data, this is a can't-miss package.

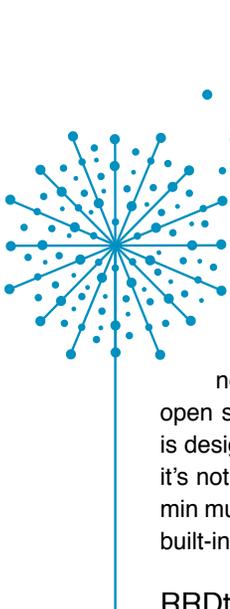
Wireshark

Any day that I have to resort to analyzing packet captures is a bad day, but the venerable Wireshark [13] always helps make it a little easier. Although the project is almost old enough to vote in the US, Wireshark 2.0 [14] was released in November 2015. To say 2.0 is a major change is an understatement: The UI was rewritten in Qt [15], and many of the interactions have been simplified. This new release brings a lot of improvements, but it's still the same Wireshark you've known for years.

VirtualBox

Desktop virtualization comes in many flavors these days, but I find that I always come back to Oracle's VirtualBox [16]. At work, having the same operating system and version of our software as a customer runs is valuable to me, so quick-and-easy virtual machines save me a lot of time. VirtualBox 5.0, released in July, added the ability to start VMs in "headless" mode, which works for me because I generally SSH into my virtual machines for interactions. 5.0 also brings paravirtualization support for better guest performance, disk encryption, and USB 3 support.





Visual Studio Code

The editor wars had reached a somewhat uneasy truce, but then a new kid appeared on the block. Microsoft released Visual Studio Code [17] for Windows, Linux, and OS X under the MIT license, which was big news for a company typically seen as hostile to open source software. Although Visual Studio Code is designed as a code editor, don't take that to mean it's not useful for sys admins (after all, every sys admin must do some coding). Visual Studio Code offers built-in git interactions and Docker language support.

RRDtool

Even if you've never interacted directly with RRDtool [18], chances are good you've used a project that incorporates it. RRDtool is a great package for storing and graphing time series data, which is why it's commonly used in performance monitoring systems, such as Cacti [19] and Xymon [20]. If static PNGs aren't your thing, Tobi Oetiker has also developed a JavaScript library [21] for generating RRD graphs.

Mosh

Being away from the desk can't stop the modern sys admin from keeping the bits flowing. Unfortunately, not all networks are created equal. If you've ever tried to do an SSH session over a mobile phone tether or an overburdened conference WiFi network, you know how painful it can be. Mosh eases that pain. Mosh allows for roaming network connections, and makes slow connections more tolerable by locally echoing as you type and allowing the connection to catch up. After almost two years of silence, Mosh 1.2.5 was released in July. Along with the usual bug fixes and improvements in platform support, the latest release adds IPv6 support.

RackTables

In a small environment, the sys admin may know exactly where every server is without having to think about it. In my first job, the department had about half a rack of gear. Easy peasy. When I moved to a larger group, suddenly there were 4,000 machines to know about. At that sort of scale, a tool like RackTables [22] becomes invaluable. RackTables is a tool for documenting hardware information, rack space, network configuration, and more. Think of RackTables as that terrible spreadsheet you're keeping, except not terrible.

Docker

Certainly you've heard all about Docker [23] by now. Containers are the next big thing in the devops world, and Docker is *the* container platform. The 1.9 release, which shipped at the beginning of November, adds several new features. Multi-host networking allows virtual networks to span hosts, giving admins more flexibility in their network topology. Persistent storage support is improved, as well. Docker brings a level of flexibility and scalability to infrastructure that will have big impacts on sys admins and users in the coming year.

Did we leave your favorite open source tool for sys admins off the list? We'd love to hear about it: open@opensource.com.

Links

- [1] Stack Exchange: <http://stackexchange.com/>
- [2] Blackbox: <https://github.com/StackExchange/blackbox>
- [3] GNU Privacy Guard: <https://www.gnupg.org/>
- [4] Puppet: <https://puppetlabs.com/>
- [5] Git: <https://git-scm.com/>
- [6] Mercurial: <https://www.mercurial-scm.org/>
- [7] Subversion: <http://svnbook.red-bean.com/>
- [8] Perforce: <https://www.perforce.com/>
- [9] Cygwin: <https://www.cygwin.com/>
- [10] KeePass: <http://keepass.info/>
- [11] Maemo: <http://maemo.org/>
- [12] ownCloud: <https://owncloud.org/>
- [13] Wireshark: <https://www.wireshark.org/>
- [14] Wireshark 2.0 announced: <https://www.wireshark.org/news/20151118.html>
- [15] Qt: <http://www.qt.io/>
- [16] VirtualBox: <https://www.virtualbox.org/>
- [17] Visual Studio Code: <https://code.visualstudio.com/>
- [18] RRDtool: <http://oss.oetiker.ch/rrdtool/>
- [19] Cacti: <http://www.cacti.net/>
- [20] Xymon: <http://xymon.sourceforge.net/>
- [21] RrdGraphJS: <https://github.com/oetiker/RrdGraphJS>
- [22] RackTables: <http://racktables.org/>
- [23] What is Docker: <https://opensource.com/resources/what-docker>

Author

Ben Cotton is a meteorologist by training and a high-performance computing engineer by trade. Ben works at Cycle Computing, leading the support engineering group. Ben is a community moderator for [Opensource.com](https://opensource.com). Find him on Twitter [@FunnelFiasco](https://twitter.com/FunnelFiasco).



Best Couple

Best Couple of 2015: *tar and ssh*

.....BY DAVID BOTH

The best couples complement each other, and each member of the couple contributes unique and irreplaceable parts to the whole. But some couples are very odd. Such is the case with our best couple this year: the **tar** and **ssh** commands.



So now I can use the results of that command on my local host because the standard output data stream is sent through the SSH tunnel to the local host. OK, that is good, but what does it mean?

Let's look at the **tar** command before answering that question.

Wait—what?!

Yup, that's right, the **tar** and **ssh** commands work together in interesting ways, especially when used with full consideration of the capabilities of Standard I/O (STDIO), which is also known as standard streams.

ssh

The **ssh** command is a secure and sophisticated form of terminal emulator that allows one to log in to a remote computer to access a shell session and run commands. So I could log in to a remote computer and run the **ls** command on the remote computer. The results are displayed in the **ssh** terminal emulator window on my local host. The Standard Output (STDOUT) of the command is displayed on my terminal window, but it remains on the remote host and cannot be used by the local host.

That is trivial and everyone does that. But the next step is a bit more interesting. Rather than maintain a terminal session on the remote computer and issuing multiple commands, I can simply use a command like the following to run a single command on the remote computer with the results being displayed on the local host. This assumes that SSH public/private keypairs (PPKP) are in use and I do not have to enter a password each time I issue a command to the remote host:

```
ssh remotehost ls
```

tar

The **tar** command is used to make backups. The name **tar** stands for Tape ARchive, but the command can be used with any type of recording media such as tape, hard drives, thumb drives and more. A command like the following can be used to create a backup of a home directory on the local host:

```
tar -cvf /tmp/home.tar /home
```

This command created a **tar** file—also called a tarball—named *home.tar* in the */tmp* directory. That file is a backup of everything in the home directory. Well, that's nice, but also not very interesting because it is very common.

But what can be interesting is, although many people do not realize it, if the target output file is not specified using the **-f** option, the output of the **tar** command is sent directly to STDOUT:

```
tar -cv /home
```

That means that the complete output of the **tar** command—the files being backed up—is sent to the terminal, which opens up some interesting possibilities, such as redirecting the STDOUT data stream to a backup file. That looks like the following command:

customize the dashboard to view the metrics that you want to see. To make your life easier, Piwik integrates with over 65 content management, ecommerce, and online forum systems like WordPress, Magento, Joomla!, and vBulletin using plugins. With anything else, you just need to add a tracking code to a page on your site. A number of web hosting firms offer Piwik as part of their one-click install packages. You can test drive [15] Piwik or use a hosted version [16].”

Fun fact: Maker of the LulzBot 3D printer, Aleph Objects, uses Piwik to run their analytics [17].

R

In the era of big data, now may be the time to learn R [18], which has become the programming language of choice for data scientists and others interested in statistical computing and graphics, and is touted by influencers in big data [19] like Revolution Analytics. In 2015, the R Consortium [20] became a Linux Foundation Collaborative project, created to provide support for the development of R-Hub, a new code-hosting platform for developing and distributing packages for R.

SugarCRM

SugarCRM [21] is the 800-pound gorilla in the open source customer relationship management space, and has previously been featured as one of our top 5 CRM tools [22]. The community edition of SugarCRM can be used out of the box as a complete solution for organizations hoping to do a better job of keeping their contacts manageable, or who want to turn a list of names into something actionable. Complete with huge list of features and a pluggable infrastructure that allows for even more customization, SugarCRM is a great solution for organizations that want to get a handle on their contacts. (Editor’s note: SugarCRM 6.5 Community Edition is the most recent open source version of SugarCRM and is still widely used. Open source alternatives built on SugarCRM CE are growing in popularity.)

Vagrant

In a nutshell, Vagrant [23] is a command-line tool for launching and configuring virtual machines. With Vagrant, environments are reproducible and portable, and the data that defines the environment is stored in text files, making it easy to version control your environments and manage your virtual machines just as you would code. Vagrant allows you to set up development environments on your local machine that are nearly identical to your production environment, regardless of what your host operating system is. Plus, learning how to get started with Vagrant is easy.

Thanks to Jason Baker for his help on this article.

Links

- [1] Apache Spark: <http://spark.apache.org/>
- [2] World record set for 100 TB sort by open source and public cloud team: <https://opensource.com/business/15/1/apache-spark-new-world-record>

- [3] Blender: <https://www.blender.org/>
- [4] Opensource.com Blender articles: <https://opensource.com/tags/blender>
- [5] *Blender for Dummies*: <http://blenderbasics.com/>
- [6] D3: <http://d3js.org/>
- [7] 8 excellent open source data visualization tools: <https://opensource.com/life/15/6/eight-open-source-data-visualization-tools>
- [8] Dolphin: <https://userbase.kde.org/Dolphin>
- [9] A comprehensive guide to Dolphin, a KDE file manager: <https://opensource.com/life/15/8/comprehensive-guide-dolphin-file-manager>
- [10] Git: <https://git-scm.com/>
- [11] What to know before transitioning your team to Git: <https://opensource.com/business/15/7/interview-emma-jane-hogbin-westby-git>
- [12] 4 open source alternatives to Slack for team chat: <https://opensource.com/business/15/9/alternatives-slack-team-chat>
- [13] Piwik: <http://piwik.org/>
- [14] Top 3 open source alternatives to Google Analytics: <https://opensource.com/business/14/10/top-3-open-source-alternatives-google-analytics>
- [15] Piwik demo: <http://demo.piwik.org/index.php?module=CoreHome&action=index-&idSite=7&period=day&date=yesterday#/module=Dashboard&action=embeddedIndex&idSite=7&period=day&date=yesterday&idDashboard=1>
- [16] Piwik Cloud: <http://piwik.org/hosting/>
- [17] Open ethos powers Aleph Objects’ success: <https://opensource.com/business/15/11/open-ethos-powers-lulzbots-success>
- [18] What is R?: <https://www.r-project.org/about.html>
- [19] Why now is the time to learn R: <https://opensource.com/business/14/12/r-open-source-language-data-science>
- [20] R Consortium: <https://www.r-consortium.org/>
- [21] SugarCRM: <https://www.sugarcrm.com/>
- [22] Top 5 open source customer relationship management tools: <https://opensource.com/business/14/7/top-5-open-source-crm-tools>
- [23] Vagrant: A powerful tool for configuring development environments: <https://opensource.com/business/15/9/ato-inter-view-seth-vargo>

Authors

Jen Wike Huger is Content Manager at Opensource.com, where she manages the site’s publication calendar, its editing team, and its community of writers. Follow her on Twitter @jenwike.

Jason Baker is passionate about using technology to make the world more open. His interests range from software development to bringing sunlight to local governments. Jason is particularly interested in data visualization and analysis, DIY/maker culture, simulations/modeling, geospatial technologies, and cloud computing (especially OpenStack). Follow him on Twitter @jehb.

Top 5 open source community metrics to track

BY JESUS M. GONZALEZ-BARAHONA

So you decided to use metrics to track your free, open source software (FOSS) community. Now comes the big question: Which metrics should I be tracking?

To answer this question, you must have an idea of what information you need. For example, you may want to know about the sustainability of the project community. How quickly does the community react to problems? How is the community attracting, retaining, or losing contributors? Once you decide which information you need, you can figure out which traces of community activity are available to provide it. Fortunately, FOSS projects following an open development model tend to leave loads of public data in their software development repositories, which can be analyzed to gather useful data.

In this article, I'll introduce metrics that help provide a multi-faceted view of your project community.

1. Activity

The overall activity of the community and how it evolves over time is a useful metric for all open source communities. Activity provides a first view of *how much* the community is doing, and can be used to track different kinds of activity.

Number of commits and number of merged changes after code review in the OpenStack project, as found in the OpenStack Activity Dashboard [1]. Evolution over time (weekly data).



For example, the number of commits gives a first idea about the volume of the development effort. The number of tickets opened provides insight into how many bugs are reported or new features are proposed. The number of messages in mailing lists or posts in forums gives an idea of how much discussion is being held in public.

2. Size

The size of the community is the number of people participating in it, but, depending on the kind of participation, size numbers may vary. Usually you're interested in active contributors, which is good news. Active people may leave traces in the repositories of the project, which means you

Number of authors and number of posters in mailing lists in the Xen project, as found in the Xen Project Development Dashboard [2]. Evolution over time (monthly data).



can count contributors who are active in producing code by looking at the *Author* field in git repositories, or count people participating in the resolution of tickets by looking at who is contributing to them.

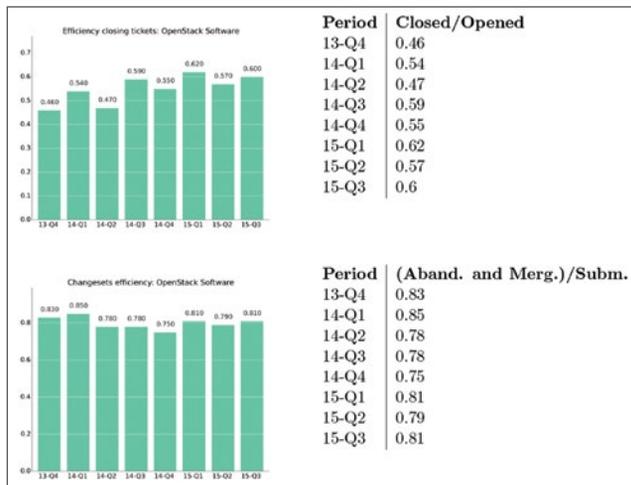
This basic idea of *activity* (somebody did something) can be extended in many ways. One common way to track activity is to look at how many people did a sizable chunk of the activity. Generally most of a project’s code contributions, for example, are from a small fraction of the people in the project’s community. Knowing about that fraction helps provide an idea of the core group (i.e., the people who help lead the community).

3. Performance

So far, I have focused on measuring quantities of activities and contributors. You also can analyze *how* processes and people are performing. For example, you can measure how long processes take to finish. Time to resolve or close tickets shows how the project is reacting to new information that requires action, such as fixing a reported bug or implementing a requested new feature. Time spent in code review—from the moment when a change to the code is proposed to the moment it is accepted—shows how long upgrading a proposed change to the quality standards expected by the community takes.

Other metrics deal with how well the project is coping with pending work, such as the ratio of new to closed tickets, or the backlog of still non-completed code reviews. Those parameters tell us, for example, whether or not the resources put into solving issues is enough.

Ratio of tickets closed by tickets opened, and ratio of change proposals accepted or abandoned by new change proposals per quarter. OpenStack project, as shown in the OpenStack Development Report, 2015-Q3 [3].



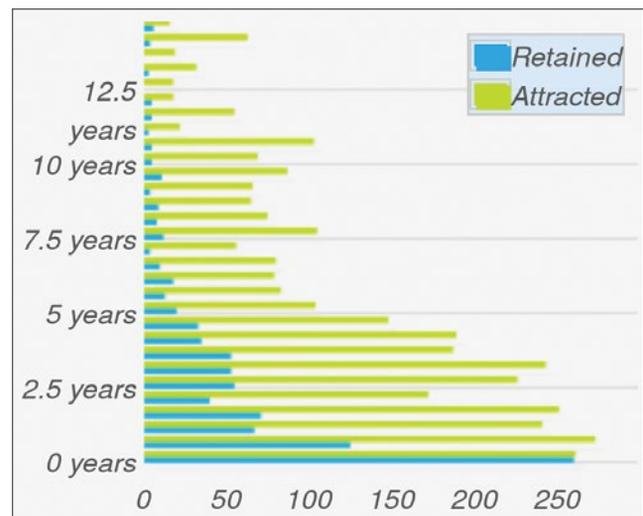
4. Demographics

Communities change as contributors move in and out. Depending on how people enter and leave a community over time, the age (time since members joined the community)

of the community varies. The community aging chart [4] nicely illustrates these exchanges over time. The chart is structured as a set of horizontal bars, two per “generation” of people joining the community. For each generation, the *attracted* bar shows how many new people joined the community during the corresponding period of time. The *retained* bar shows how many people are still active in the community.

The relationship between the two bars for each generation is the retention rate: the fraction of people of that generation who are still in the project. The complete set of attracted bars show how attractive the project was in the past. And the complete set of the retention bars shows the current age structure of the community.

Community aging chart for the Eclipse community, as shown in the Eclipse Development Dashboard [4]. Generations are defined every six months.



5. Diversity

Diversity is an important factor in the resiliency of communities. In general, the more diverse communities are—in terms of people or organizations participating—the more resilient they are. For example, when a company decides to leave a FOSS community, the potential problems the departure may cause are much smaller if its employees were contributing 5% of the work rather than 85%.

Pony and Elephant Factor for several FOSS projects in the area of cloud computing, as presented in The quantitative state of the open cloud 2015 (slides) [8].

	Pony Factor	Elephant Factor	Commits (excl bots)
OpenNebula	4	1	12K
Eucalyptus	5	1	25K
CloudStack	14	1	42K
OpenStack	>100	6	126K
CloudFoundry	41	1	60K
OpenShift	10	1	15K
Docker	15	1	18K
Kubernetes	12	1	7K

The Pony Factor [6], a term defined by Daniel Gruno [7] for the minimum number of developers performing 50% of the commits. Based on the Pony Factor, the Elephant Factor is the minimum number of companies whose employees perform 50% of the commits. Both numbers provide an indication of how many people or companies the community depends on.

There are many other metrics to help measure a community. When determining which metrics to collect, think about the goals of your community, and which metrics will help you reach them.

Links

- [1] OpenStack Activity Dashboard: <http://activity.openstack.org/>
- [2] Zen Project Development Dashboard: <http://projects.bitergia.com/xen-project-dashboard/browser/>
- [3] OpenStack Development Report, 2015-Q3: http://activity.openstack.org/dash/reports/2015-q3/pdf/2015-q3_OpenStack_report.pdf
- [4] Measure your open source community's age to keep it healthy: <http://radar.oreilly.com/2014/10/measure-your-open-source-communitys-age-to-keep-it-healthy.html>

- [5] Eclipse Development Dashboard: <http://dashboard.eclipse.org/demographics.html>
- [6] Pony Factor Math: <https://ke4qqq.wordpress.com/2015/02/08/pony-factor-math/>
- [7] Daniel Gruno: <https://twitter.com/humbedooh>
- [8] The quantitative state of the open cloud 2015 (slides): <https://speakerdeck.com/jgbarah/the-quantitative-state-of-the-open-cloud-2015-edition>

Author

Jesus M. Gonzalez-Barahona is co-founder of Bitergia (bitergia.com), the software development analytics company specializing in the analysis of free and open source software projects. He also teaches and researches in Universidad Rey Juan Carlos in Spain as part of the GSyC/LibreSoft research group. His interests include the study of software development communities, and he focuses on quantitative and empirical studies of these groups. Find him on Twitter [@jgbarah](https://twitter.com/jgbarah).

WRITE FOR US

Would you like to write for [Opensource.com](https://opensource.com)? Our editorial calendar includes upcoming themes, community columns, and topic suggestions: <https://opensource.com/calendar>

Learn more about writing for [Opensource.com](https://opensource.com) at: <https://opensource.com/writers>

We're always looking for open source-related articles on the following topics:

Apache: Stories and open source updates about Apache Software Foundation projects.

Big Data: Open source big data tools, stories, communities, and news.

Command-line tips: Tricks and tips for the Linux command-line.

DevOps: DevOps best practices, case studies, and “getting started” stories.

Diversity: How you increased diversity in your community, open source-related grants and scholarship opportunities, tips for attracting a more diverse pool of contributors, how increasing diversity helped your project, and other practical advice for fostering inclusive communities.

Geek Culture: Open source-related geek culture stories.

Hardware: Open source hardware projects, maker culture, new products, howtos, and tutorials.

HFOSS: Humanitarian open source project stories, news, communities, and updates.

Content management systems: Howtos and articles on Drupal, ezPublish, Joomla, TYPO3, WordPress, and other open source content management systems.

Education: Open source projects, tools, solutions, and resources for education.

Science: Open source tools, programs, projects, and howtos for science.

Diversity in open source highlights from 2015

BY CINDY PALLARES-QUEZADA

The pool of people participating in open source communities still lacks diversity, but the good news is that many people, projects, and organizations are working to improve it. I've collected a few highlights from 2015 efforts to increase diversity in open source communities. Which 2015 diversity in open source stories would you add to the list?

Outreachy grows its diversity internship program
Outreachy [1], a member project of Software Freedom Conservancy [2], expanded its internship program to be open to people of color underrepresented in tech in the U.S. The program has been a great success in providing paid remote internships in free and open source organizations for women, trans men, and genderqueer people across the globe.

The expansion was driven by demographic data collected and released by leading U.S. tech companies, which verified a lack of racial and ethnic diversity among employees in technical roles in comparison to U.S. population demographics. Outreachy organizers have observed the same disparity of demographic representation in free and open source communities and conferences, and they hope to create more opportunities for other underrepresented groups in the future.

Minority opportunities in open source

Richard Tapia Celebration of Diversity in Computing [3], a conference aimed at providing a supportive networking environment for underrepresented groups across the broad range of computing and information technology, hosted a panel called "Opportunities in open source" that introduced many students to

free and open source software licenses, discussed barriers and accessibility within free and open source projects, and shed light on paid internships and jobs in free and open source software.

Grace Hopper Celebration of Women in Computing embraces open source

Grace Hopper Celebration of Women in Computing [4], the world's largest gathering of women technologists, hosted an Open Source Day [4]. Attendees interested in helping free and open source organizations such as OpenStack, Mozilla, Cloudera, and OpenHatch, had a chance to contribute. The track included numerous presentations and panels on security, humanitarian efforts, open hardware, infrastructure, and more by women working in free and open source software. The event helped teach beginners about decentralization, software licenses, and the open source philosophy.

Ada Initiative ends, but its work continues

The Ada Initiative, which supported women in open technology and culture through unconferences, workshops, and more, announced its shutdown [6] in 2015. As an effort to preserve and continue the work started by the organization, the Ada Initiative published an AdaCamp toolkit [7] and Imposter Syndrome training under a Creative Commons Attribution ShareAlike license, making them freely available for use by anyone.

Defining a diverse community

Sarah Sharp [8], open graphics developer at Intel and former Linux Kernel developer, created a step-by-step breakdown [9] of a diverse community. In it she mentions how



a good open source community includes a good code of conduct, has ways of handling burnout, values non-technical skills, and has mentorship programs. It's a great place to start when trying to learn about diversity in open source communities.

OpenStack gets a diversity working group

The OpenStack Foundation's board of directors supported a diversity working group [10]. The group will sponsor and create programs to encourage diversity by identifying and removing barriers to joining and contributing to the OpenStack community. The group already started to target gender and geographic diversity and is planning on expanding the OpenStack code of conduct [11].

Red Hat presents its first Women in Open Source Awards

Red Hat Women in Open Source Awards [12] celebrate women who are already contributing to open source and shine a light on them as role models for others. The awards have two categories: "community" for professionals and "academic" for students. Red Hat judges reviewed more than 100 community-submitted nominations and selected five finalists in each category. The open source community then voted on the winners. The winners were Sarah Sharp, an open graphics developer at Intel, and Kesha Shah [13], who contributed to Systems [14] and BRL-CAD [15] through Google Summer of Code. Both Sarah and Kesha have done extensive mentoring in the community. Voting for the 2015 Women in Open Source Award start in February.

Links

- [1] Outreachy: <https://www.gnome.org/outreachy/>
- [2] Software Freedom Conservancy: <https://sfconservancy.org/>
- [3] ACM Richard Tapia Celebration of Diversity in Computing: <http://tapiaconference.org/>
- [4] Grace Hopper Celebration of Women in Computing: <http://ghc.anitaborg.org/>
- [5] Open Source Day 2015: <http://ghc.anitaborg.org/conference-overview/open-source-day-2015/>
- [6] Ada Initiative organization to end, but its work will continue: <https://opensource.com/business/15/8/ada-initiative-legacy>
- [7] AdaCamp Toolkit: <http://adacamp.org/>
- [8] Sarah Sharp: <https://twitter.com/sarahsharp>
- [9] What makes a good community?: <http://sarah.thesharps.us/2015/10/06/what-makes-a-good-community/>
- [10] OpenStack Diversity Working Group: <https://wiki.openstack.org/wiki/Diversity>
- [11] OpenStack Foundation Community Code of Conduct: <https://www.openstack.org/legal/community-code-of-conduct/>
- [12] Red Hat Women in Open Source Awards: <http://www.redhat.com/en/about/women-in-open-source>
- [13] Kesha Shah: <https://twitter.com/geekkygirl>
- [14] Systems: <http://anitaborg.org/get-involved/systems/>
- [15] BRL-CAD: <http://brlcad.org/>

Author

Cindy Pallares-Quezada is an Outreachy alumna and a Red Hat intern working on OpenStack. She enjoys helping contributors get started on open source and making things with open source hardware.

2015 was a good year for creating the world's 'missing maps' with OpenStreetMap

BY DRISHTIE PATEL

The Missing Maps project [1], which launched in 2014, aims to literally and figuratively put more than 20-million at-risk people on the map using OpenStreetMap (OSM) [2] as a platform. We need to fill in “missing maps” before the next disaster strikes, ensuring the maps have detail sufficient for emergency responders to hit the ground running.

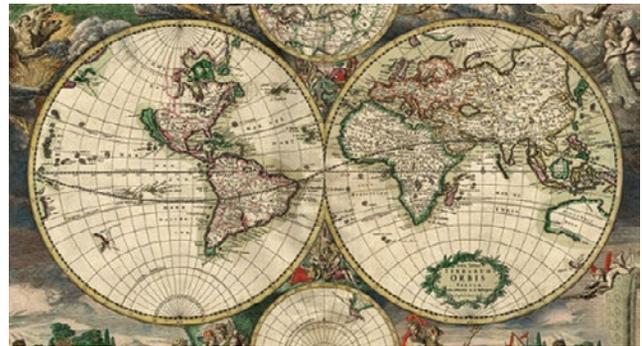
OpenStreetMap is an open and free source of geographic data. Anyone with a username can add, edit, or update data, so the Missing Maps project is community driven and focuses on local knowledge. Remote volunteers around the world use satellite imagery to trace features, such as roads and buildings. Community members and volunteers in the area then use the base map to add local data to these shapes, including street names, addresses, building types, and points of interest.

As we look back at 2015, here are a few success stories from the Missing Maps project.

Zimbabwe

One area Missing Maps started working in is Dzivarasekwa, a high-density urban township in Harare's North District in Zimbabwe. Community mapping in Dzivarasekwa was performed as part of an urban disaster risk-reduction project, with the goal of strengthening community safety and resilience to reduce the number of deaths, injuries, and socio-economic impacts caused by urban disasters.

Community discussions and the baseline survey resulted in guided decisions about which details to collect during the community mapping activity. Volunteers in Dzivarasekwa surveyed more than 8,000 buildings across 7 square kilometers. One output of the mapping is shown below. The results showed a correlation between housing quality and building material type, with concrete homes observed to be in worse condition than brick homes. When combined, this data provided the locations of residents most



vulnerable to disasters, such as flooding, making it easier to plan future disaster risk-reduction activities. The Zimbabwe Red Cross can now plan to serve areas that are deemed high priority based on survey and map results and target specific homes that require help immediately. Now with detailed information including house addresses, the process is much more efficient.

Community mapping was conducted to complement a baseline survey and urban risk profile, and has helped to inform the activities that will be implemented in the next stages of the resilience project.



Tanzania

Ushirombo is a predominantly rural area in northwest Tanzania, with about 84% of the residents relying on agriculture as the primary livelihood source. The rest depend on activities such as small-scale gold mining, lumbering, and informal micro-enterprises.

Community discussions and a baseline survey revealed waterborne and water-related disease is one of the primary issues in this region because most of the residents are dependent on collecting water from traditional wells and natural springs that are neither clean nor safe. Therefore, recording water and sanitation points was a priority during community mapping.

Missing Maps trained volunteers and community leaders on how to use Field Papers [3] and OpenMapKit [4] to collect data. Mappers were able to collect data in Ushirombo quickly because buildings are within close proximity to each other on flat terrain; however, many challenges still hindered the collection process, such as the lack of road names and formal building addresses. Old satellite imagery also contributed to the problem because the base maps on Field Papers and the OpenMapKit app did not match the reality on the ground.

Volunteers using Field Papers and OpenMapKit to collect local data that will be added to OpenStreetMap. Copyright American Red Cross.



Despite these challenges, the participation of community leaders in the community-mapping process led to the success of mapping Ushirombo. Leaders such as village chiefs joined the field mapping expeditions to help navigate and manage community relationships. This aided data quality and collection efficiency because mappers were able to continue collecting data while community leaders spent time answering questions from residents.

Map data is now used by the Tanzania Red Cross and American Red Cross to make programmatic decisions for an urban risk reduction program in the area.

Rwanda

Imagine not being able to use your phone to look up directions, nearby hospitals, restaurants, or just about any answer to a question at the tips of your fingers. For many people this is reality, because most of the world is not on the map. Thousands of towns and cities are invisible on the world map. This makes it difficult for disaster responders to come to residents' aid when an earthquake, landslide, or typhoon strikes.

Communities in Cyahinda, Rwanda, an agricultural region in the Nyaruguru district near the Burundi border, face challenges such as poor health and sanitation, environmental degradation, and disaster risks related to landslides and drought.

American Red Cross staff setting up drop off and pick up locations for volunteer mappers in Cyahinda, Rwanda during community mapping in the field. Copyright American Red Cross.



Low-density villages and collections of buildings are spread across hills terraced for small-scale farming, connected by dirt roads and tracks. The Rwanda Red Cross and American Red Cross spent two weeks in April mapping the area. Field mapping in Cyahinda's rural setting was also slower than other mapping projects due to the size of the area, wide distribution of buildings, and tiring walks through the hilly terrain.

The validated base data was uploaded to OpenStreetMap, while the project-specific data is being used to create maps to assist with planning and decision-making. For example, residents may decide to look at the distribution of mudslides to decide where to safely build their homes, schools, and churches.

South Africa

Khayelitsha is home to roughly 400,000 people covering an area of 39 square kilometers made up of old formal areas and a majority newer informal areas. Red Cross partners have been working in the area and looked into the major

concerns the community is facing. Fires are at the top of the list. Fires regularly occur as a result of indoor stove use, trash burning, faulty wires, and residents trying to keep warm. Rapid, unplanned development results in close construction of homes, which helps fires spread quickly. Pathways between homes are narrow and often blocked, making evacuations chaotic and dangerous. Residents commonly do not know who to call for firefighting assistance, and if firefighters are available, they have a difficult time finding and responding to fires.

Volunteers using Field Papers and GPS to collect local data that will be used for programs and added to OpenStreetMap. Copyright American Red Cross.



Red Cross partners are piloting a project to solve this issue: a low-cost, meshed network of smart home sensors affixed to each home within the informal settlement. The American Red Cross GIS team recently set out to

Khayelitsha to support the community in mapping the area for better program planning and decision making. Remote tracing was done ahead of time by digital humanitarians around the world using OpenStreetMap and this was hugely beneficial, giving us a fairly accurate foundation of data to add to.

Volunteers on the ground were trained to add local data such as street names, house numbers, points of interest, and water sources among other features. The area is now full for relevant information, not just for this project, but data is being used by other humanitarian organizations working in the area and more importantly the community living in Khayelitsha. The American Red Cross GIS has created large maps of the different sections to print and share with the communities. The volunteers are continuing to gather data, collecting the locations of the fire sensors to then overlay OSM data to analyze the coverage of the project.

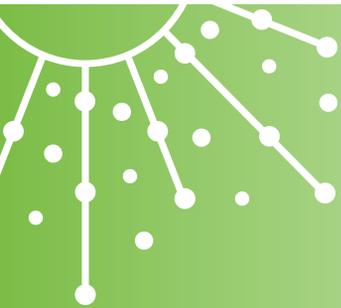
Links

- [1] Missing Maps: <http://www.missingmaps.org/>
- [2] OpenStreetMap: <https://www.openstreetmap.org/#map=5/51.500/-0.100>
- [3] Field Papers: <http://fieldpapers.org/>
- [4] OpenMapKit: <https://github.com/americanredcross/openmapkit>

Author

Drishtie Patel is passionate about open data and community engagement. As a GIS analyst, and as Missing Maps Project Coordinator at the American Red Cross, she works to empower local vulnerable communities by developing and strengthening their skills using the OpenStreetMap platform.





Most likely to succeed

10 projects to fork in 2016

.....BY JASON BAKER

2015 was a year of many new open source projects hitting the scene with a splash. From enterprise solutions to home brewed open source concoctions, many of the projects released as open source software this year have made a huge impact on the world of computing in a very short amount of time. While flash stardom isn't always the best predictor of longevity, we think these 10 projects just might have come onto the scene with enough momentum to continue their success in the new year.

Fortunately, it's easy to measure developer interest for projects being released on GitHub. This list is a compilation of some of our favorites from the most starred releases from GitHub repositories created in the past year with open source licenses. They piqued our curiosity, and if they pique yours too, why not take a look? Clone your favorite and try it out. And if you want to join in on the action, why not fork a copy, contribute some changes, and help make these projects continue to succeed in the new year.

React Native

React Native [1] is a JavaScript framework that makes it easy for developers to create native applications for Android or iOS using the same principles that React has enabled them to use for web applications. React, created by Facebook, was first released in 2013, and the native framework was made available earlier this year with much anticipation. Already receiving over 4,000 forks on GitHub and almost 25,000 stars, React Native has already racked up over 450 individual contributors to its repository. React Native is available under a BSD license [2].

Swift

Swift [3], a new programming language by Apple designed to be a successor to Objective-C as the language of choice for writing applications for Apple's platforms and beyond, was released to the world in 2014. But earlier this month, Apple took the step of releasing the code base as open source under an Apache 2.0 license [4]. Swift is a compiled language, and many efforts have been made to optimize it for speed, readability, and modern programming techniques.

The Art of Command Line

Have you always wanted to become a command-line ninja? Who hasn't wanted to be able to effortlessly punch out a seemingly cryptic series of commands to instantly bend their machine to their will? Okay, this repository isn't going to replace years of training and practice to instantly turn you into an elite system administrator, but if you're a new Linux user or just one who hasn't spent much time behind a terminal window, you might want to check this out. The Art of Command Line [5] isn't a how-to or tutorial in its own right, but more of a study guide or a body of knowledge for those seeking to upgrade their terminal-fu. It's a giant list of things to learn in order to become the master of the shell, and available under a Creative Commons Attribution-ShareAlike 4.0 license [6].

Material Design Lite

The Material Design Lite [7] components by Google are designed to make it easy to add a material design look and feel to your website and web applications. What is material design? It's a consistent look and feel designed to give virtual components, from buttons to toggles to input fields, a visual style which feels intuitive to interact with through the careful use of light, shadows, movement, and interactivity. You'll immediately recognize it as the look of Google-made web applications. Material Design Lite is released under the Apache 2.0 license [8].

TensorFlow

TensorFlow [9], another contribution to this list from Google, is an open source library for numerical computation using data flow graphs. It connects operations and data with nodes and graph edges, much in the same way the neural networks of the human brain are organized. TensorFlow is therefore a framework for building machine learning applications, making it ideal for complex applications like language processing which are difficult to impossible to develop with traditional algorithmic paradigms. TensorFlow is released as open source under the Apache 2.0 license [10].

Dragula

Dragula [11] sums up its tag line as “Drag and drop so simple it hurts.” We hope it doesn’t hurt, but Dragula does make it easy to create web applications with drag and drop functionality without relying on huge frameworks or complicated custom code. Just insert a few lines of code into your application to tell Dragula which components are draggable, and it handles everything else: clicks, touch interaction, shadows, even the sort order. Dragula is available under an MIT license [12].

Redux

Redux [13] is a tiny JavaScript library which provides the ability to create predictable state containers for your applications. By imposing a few artificial constraints, and integrating seamlessly with other frameworks like React, you can easily build applications with a model-view-controller (MVC) architecture that behave consistently across different platforms and devices, and help to make the testing of these applications easier and more predictable. Redux is made available as open source under an MIT license.

Big List of Naughty Strings

The Big List of Naughty Strings [15] isn’t a compilation of things you shouldn’t say to your mother. Rather, it’s a list of strings that are likely to cause problems for applications if they don’t adequately clean user input. It’s safe to say that if someone can input a harmful string into your application, eventually someone will. Whether those strings inject data via SQL, through server code or command injection, or introduce other vulnerabilities, it’s important to make sure that your program cleans its string input to prevent malicious or just unexpected behavior. Use this collection as a part of your security tests to make sure your applications behaves properly. The Big List of Naughty Strings is available under an MIT license [16].

Linux insides

Have you ever wanted to understand how the Linux kernel works? As in, really works, including the nitty-gritty details? Much of the freely available documentation out there focuses on the user-facing parts of the Linux operating system, along with its components like various GNU components, desktop environments, and applications; and for a project as large and complex as Linux, a beginner would easily get lost just trying to dive straight into the source code. Linux insides [17] is a series of articles written to give you a picture of the inner workings of the Linux kernel, and it’s completely free under a Creative Commons Attribution NonCommercial ShareAlike 4.0 license [18].

Deepdream

Deepdream [19] is a Google-created computer vision program that uses neural networks to interpret images by detecting patterns and amplifying them. For an artificial intelligence researcher or developer, it can be used to classify images based on their contents. For everyone else, it’s a library for creating some really psychedelic colorful interpretations of

the input images. Packaged with an IPython notebook to make experimenting easy, this repository lets even novice users turn images into dream-like landscapes. Deepdream is released under an Apache 2.0 license [20].

Links

- [1] React Native: <https://github.com/facebook/react-native>
- [2] React Native license: <https://github.com/facebook/react-native/blob/master/LICENSE>
- [3] Swift: <https://github.com/apple/swift>
- [4] license: <https://github.com/apple/swift/blob/master/LICENSE.txt>
- [5] The Art of Command Line: <https://github.com/jlevy/the-art-of-command-line>
- [6] Attribution-ShareAlike 4.0 International (CC BY-SA 4.0): <http://creativecommons.org/licenses/by-sa/4.0/>
- [7] Material Design Lite: <https://github.com/google/material-design-lite>
- [8] Material Design Lite license: <https://github.com/google/material-design-lite/blob/master/LICENSE>
- [9] TensorFlow: <https://github.com/tensorflow/tensorflow>
- [10] TensorFlow license: <https://github.com/tensorflow/tensorflow/blob/master/LICENSE>
- [11] Dragula: <https://github.com/bevacqua/dragula>
- [12] Dragula license: <https://github.com/bevacqua/dragula/blob/master/license>
- [13] Redux: <https://github.com/rackt/redux>
- [14] Redux license: <https://github.com/rackt/redux/blob/master/LICENSE.md>
- [15] Big List of Naughty Strings: <https://github.com/minimaxir/big-list-of-naughty-strings>
- [16] Big List of Naughty Strings license: <https://github.com/minimaxir/big-list-of-naughty-strings/blob/master/LICENSE>
- [17] Linux insides: <https://github.com/0xAX/linux-insides>
- [18] Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0): <http://creativecommons.org/licenses/by-nc-sa/4.0/>
- [19] Deepdream: <https://github.com/google/deepdream>
- [20] Deepdream license: <https://github.com/google/deepdream/blob/master/LICENSE>



Author

Jason Baker is passionate about using technology to make the world more open. His interests range from software development to bringing sunlight to local governments. Jason is particularly interested in data visualization and analysis, DIY/maker culture, simulations/modeling, geospatial technologies, and cloud computing (especially OpenStack). Follow him on Twitter @jehb.

5 great Raspberry Pi projects for the classroom

BY BEN NUTTALL

We round up 5 fun Raspberry Pi projects for the classroom:

1. Minecraft Pi

Minecraft is the favorite game of pretty much every teenager in the world—and it's one of the most creative games ever to capture the attention of young people. The version that comes with every Raspberry Pi is not only a creative thinking building game, but comes with a programming interface allowing for additional interaction with the Minecraft world through Python code.

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



Minecraft: Pi Edition is a great way for teachers to engage students with problem solving and writing code to perform tasks. You can use the Python API to build a

house and have it follow you wherever you go, build a bridge wherever you walk, make it rain lava, show the temperature in the sky, and anything else your imagination can create.

Learn more in [Getting Started with Minecraft Pi](#) [1].

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



2. Reaction game and traffic lights

It's really easy to get started with physical computing on Raspberry Pi—just connect up LEDs and buttons to the GPIO pins, and with a few lines of code you can turn lights on and control things with button presses. Once you know

the code to do the basics, it's down to your imagination as to what you do next!

If you know how to flash one light, you can flash three. Pick out three LEDs in traffic light colors and you can code the traffic light sequence. If you know how to use a button to trigger an event, then you have a pedestrian crossing! Also look out for great pre-built traffic light add-ons like Pi Traffic Light [3], Pi-Stop [4], TrafficHAT [5], and more.

Courtesy of Low Voltage Labs [2]. CC BY-SA 4.0.



It's not always about the code—this can be used as an exercise in understanding how real world systems are devised. Computational thinking is a useful skill in any walk of life.

Next, try wiring up two buttons and an LED and making a two-player reaction game—let the light come on after a random amount of time and see who can press the button first!

To learn more, check out GPIO Zero recipes [6]. Everything you need is in CamJam EduKit 1 [7].

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



3. Sense HAT Pixel Pet

The Astro Pi—an augmented Raspberry Pi—went to space in December 2015, but you haven't missed your chance to get your hands on the hardware. The Sense HAT is the sen-

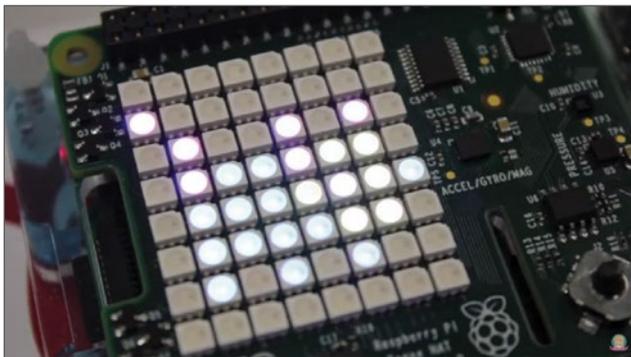
sensor board add-on used in the Astro Pi mission and it's available for anyone to buy. You can use it for data collection, science experiments, games and more. Watch the Raspberry Pi Sense HAT Pixel Tutorial video [8] from Raspberry Pi's Carrie Anne (on the Geek Gurl Diaries YouTube channel [9]) for a great way to get started—by bringing to life an animated pixel pet of your own design on the Sense HAT display.

Learn more in “Exploring the Sense HAT” [10].

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



4. Infrared bird box

A great exercise for the whole class to get involved with—place a Raspberry Pi and the NoIR camera module inside a bird box along with some infra-red lights so you can see in the dark, then stream video from the Pi over the network or on the internet. Wait for birds to nest and you can observe them without disturbing them in their habitat. Learn all about

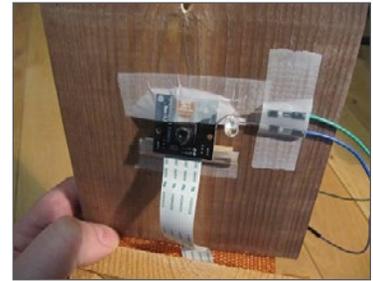
Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



infrared and the light spectrum, and how to adjust the camera focus and control the camera in software.

Learn more in “Make an infrared bird box” [11].

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



5. Robotics

With a Raspberry Pi and as little as a couple of motors and a motor controller board, you can build your own robot. There is a vast range of robots you can make, from basic buggies held together by sellotape and a homemade chassis, all the way to self-aware, sensor-laden metallic stallions with camera attachments driven by games controllers.

Learn how to control individual motors with something straightforward like the RTK Motor Controller Board (£8/\$12), or dive into the new CamJam robotics kit (£17/\$25) which comes with motors, wheels and a couple of sensors—great value and plenty of learning potential.

Courtesy of the Raspberry Pi Foundation. CC BY-SA 4.0.



Alternatively, if you'd like something more hardcore, try PiBorg's 4Borg [12] (£99/\$150) or DiddyBorg [13] (£180/\$273) or go the whole hog and treat yourself to their DoodleBorg Metal edition (£250/\$380) — and build a mini version of

their infamous DoodleBorg tank [14] (unfortunately not for sale).

Check out the CamJam robotics kit worksheets [15].

Do you have a fun Raspberry Pi project to share? Send us your story: open@opensource.com.

Courtesy of Low Voltage Labs. CC BY-SA 4.0.



3 open hardware projects for beginners

BY ALICIA GIBB

I'm part of the Lunchbox

Electronics [1] team, where we create innovative, new products with great imagination and passion for open source hardware. Our goal is to combine the art and engineering of an R&D Lab with playful products and projects. We believe that learning should be intuitive and fun.

Looking back over 2015, we picked three of our favorite projects and one open hardware book to keep you learning and making.

The stop-and-go DIY soldering kit

As the name suggests, you can build the stop and go DIY soldering kit [2] yourself. This kit cycles through red, yellow, and green LEDs from an ATtiny85 and is a great way to practice your soldering skills. It's quick to put together (takes about 10 minutes) and the chip has been programmed for you. No need to know how to code.

The holes on the base of the stop and go light are LEGO compatible, so you can build this kit into your various brick projects. You are welcome to create a derivative of this kit as long as it abides by the open source hardware definition [3].

Make this stoplight to learn more about soldering. Image courtesy of Lunchbox Electronics. CC BY-SA 4.0.



Build upons: World's tiniest light-up bricks

Build Upons [4] are light-up bricks that are compatible with LEGO. The Build Upons system has three types of bricks: an LED brick, a power brick, and a bridge brick. Build Upons have a 1x1 LED brick to enable the user to create elegant designs. With Build Upons, the user is able to place the LED bricks anywhere throughout their design for ultimate flexibility. You

Image courtesy of Lunchbox Electronics. CC BY-SA 4.0.



don't need to know electronics to use these bricks—you simply build just like you've always done and you will learn along the way. The source for Build Upons will be released as open source hardware once the designs have been finalized for our production pieces.

Blinky Buildings

The Blinky Buildings Empire State kit is a simple kit to teach people how to make derivatives of open source hardware. Make a derivative of a building in your city! You are welcome to create a derivative of this kit as long as it abides by the open source hardware definition [3].

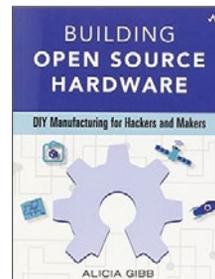
Building Open Source Hardware

Building Open Source Hardware [5] is a field guide I wrote

for designing and open sourcing your hardware. I curated a selection of chapters written by a group of community members that include pointers on how to manufacture products,

which license is best for you, and the history of the open source hardware movement. It's a hands-on guide with several checklists to help you understand how best to open source your device. Plus, royalties of this book are given to the Open Source Hardware Association [6]. Happy Hacking!

The Blinky Building Kit. Photo courtesy of Lunchbox Electronics. CC BY-SA 4.0.



Links

- [1] Lunchbox Electronics: <http://www.lunchboxelectronics.com/>
- [2] Stop and Go kit: <http://www.lunchboxelectronics.com/product/stopandgo>
- [3] Open source hardware definition: <http://www.oshwa.org/definition/>
- [4] Build Upons: <https://www.kickstarter.com/projects/1828089281/build-upons-worlds-tiniest-light-up-bricks>
- [5] Building Open Source Hardware: <https://www.sparkfun.com/products/13220>
- [6] Open Source Hardware Association: <http://www.oshwa.org/>

Author

Alicia Gibb is CEO of Lunchbox Electronics. She's a jack-of-all-trades (even though she holds two masters degrees). She teaches physical computing in the ATLAS Institute (part of the Engineering Department at CU Boulder), and is the director of a hackerspace—the Blow Things Up Lab—there.

Publisher's picks: 29 open source books for 2015

BY RIKKI ENDSLEY

Whether you prefer reading technical content online, in print, or both, technical book publishers have you covered. Six publishers share picks for top 2015 open source-related book releases.

Addison-Wesley Professional

The Go Programming Language

[http://www.informit.com/store/](http://www.informit.com/store/go-programming-language-9780134190440)

[go-programming-language-9780134190440](http://www.informit.com/store/go-programming-language-9780134190440)

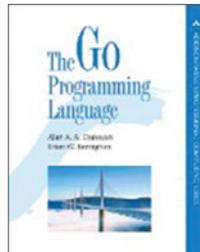
by Alan A. Donovan and Brian W. Kernighan

ISBN: 0134190440

400 Pages

Publication Date: October 2015

Available eBook Formats: EPUB, MOBI, and PDF



The Go Programming Language is the authoritative resource for any programmer who wants to learn Go. It shows how to write clear and idiomatic Go to solve real-world problems. The book does not assume prior knowledge of Go nor experience with any specific language, making it accessible to readers whether they're most comfortable with JavaScript, Ruby, Python, Java, or C++. The book features hundreds of interesting and practical examples of well-written Go code that cover the whole language, its most important packages, and a wide range of applications. Each chapter has exercises to test for understanding and explore extensions and alternatives.

Effective Python: 59 Specific Ways to Write Better Python

[http://www.informit.com/store/effective-](http://www.informit.com/store/effective-python-59-specific-ways-to-write-better-python-9780134034287)

[python-59-specific-ways-to-write-better-](http://www.informit.com/store/effective-python-59-specific-ways-to-write-better-python-9780134034287)

[python-9780134034287](http://www.informit.com/store/effective-python-59-specific-ways-to-write-better-python-9780134034287)

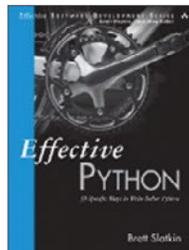
by Brett Slatkin

ISBN: 0-13-403428-7

256 Pages

Publication Date: February 2015

Available eBook Formats: EPUB, MOBI, and PDF



This book will help readers master a truly “Pythonic” approach to programming, harnessing Python’s full power to

write exceptionally robust and well-performing code. Using the concise, scenario-driven style pioneered in Scott Meyers’ best-selling *Effective C++* [1], Brett Slatkin brings together 59 Python best practices, tips, and shortcuts, and explains them with realistic code examples. Drawing on his years of experience building Python infrastructure at Google, Slatkin uncovers little-known quirks and idioms that powerfully impact code behavior and performance. He teaches the best way to accomplish key tasks, in order to write code that’s easier to understand, maintain, and improve.

The Dart Programming Language

[http://www.informit.com/store/dart-](http://www.informit.com/store/dart-programming-language-9780321927705)

[programming-language-9780321927705](http://www.informit.com/store/dart-programming-language-9780321927705)

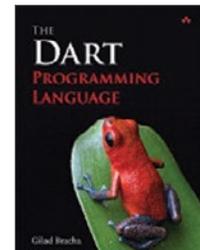
by Gilad Bracha

ISBN: 0-321-92770-2

224 Pages

Publication Date: December 2015

Available eBook Formats: EPUB, MOBI, and PDF



In this rigorous but readable introductory text, Dart specification lead Gilad Bracha fully explains both the language and the ideas that have shaped it, offering an authoritative description of Dart for programmers, computer science students, and other well-qualified professionals. The text illuminates key programming constructs with significant examples, focusing on principles of the language, such as optional typing and pure object-orientation. Bracha thoroughly explains reflection in Dart, showing how it is evolving into a form that programmers can easily apply without creating excessively large programs. He also shares valuable insights into Dart’s actor-style model for concurrency and asynchronous programming. Throughout, he covers both language semantics and the rationale for key features, revealing not just what Dart does, but why it works the way it does.

Clojure Recipes

<http://www.informit.com/store/clojure-recipes-9780321927736>

by Julian Gamble

ISBN: 0-321-92773-7

288 Pages

Publication Date: October 2015

Available eBook Formats: EPUB, MOBI, and PDF



This is a wide-ranging, up-to-date “code recipe book” for an increasingly popular language. With practical and self-contained examples, Gamble illuminates Clojure’s key features and best practices, showing how to solve real-world problems one step at a time. Focusing on Clojure 1.7 and higher, the book fully reflects recent enhancements to ensure the most up-to-date code for a given project.

Apress**Shell Scripting Recipes, Second Edition**

<http://www.apress.com/9781484202210?gtmf=s>

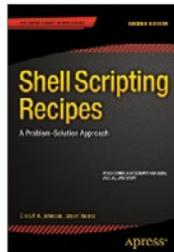
by Jayant Varma

ISBN13: 978-1-484202-21-0

380 Pages

Publication Date: October 2015

Available eBook Formats: EPUB, MOBI, PDF



Shell Scripting Recipes is filled with more than 150 much-needed and practical recipes that follow a problem-solution format, and help all Unix users regain some of the lost time spent creating and testing shell scripts. Most scripts in this book are POSIX (Portable Operating System Interface)-compliant, so they are supported by many of the major shell variants, including Bash, ksh, and sh, among others. This completely updated second edition takes into account the many changes to shell scripting that have occurred since its original publication nearly 10 years ago, and all the recipes are now relevant for a modern audience.

Each real-world example recipe follows the same structure and easily shows you what’s going on in each case. File conversion (DOS, UNIX, and Mac), system administration, and resource monitoring are just a few of the topics covered in this highly practical shell scripting reference. You will also find recipes for filename management, complex date calculations, screen control capabilities, and more.

Pro Bash Programming, Second Edition

<http://www.apress.com/9781484201220?gtmf=s>

by Jayant Varma, Chris F. A. Johnson

ISBN13: 978-1-484201-22-0

256 Pages

Publication Date: June 2015

Available eBook Formats: EPUB, MOBI, PDF



Pro Bash Programming teaches you how to effectively utilize the Bash shell in your programming. The Bash shell is a complete programming language, not merely a glue to combine external Linux commands. By taking full advantage of Shell internals, Shell programs can perform as snappily as utilities written in C or other compiled languages. And you will see how, without assuming Unix lore, you can write professional Bash 4.3 programs through standard programming techniques. This second edition has updated for Bash 4.3, and many scripts have been rewritten to make them more idiomatically Bash, taking better advantage of features specific to Bash. It is easy to read, understand, and will teach you how to get a grip on Bash programming, without drowning you in pages and pages of syntax.

Pro Freeware and Open Source Solutions for Business

<http://www.apress.com/9781484211311?gtmf=s>

by Philip Whitt

ISBN13: 978-1-484211-31-1

280 Pages

Publication Date: September 2015

Available eBook Formats: EPUB, MOBI, PDF



A practical guide for the small business owner seeking viable alternative to expensive commercial software packages and subscriptions. This reference guide provides an overview of many popular and powerful free alternatives to proprietary software. In this book you will find free and open source solutions for office productivity, PDF creation, accounting, image editing and graphic design, desktop publishing, 3D design, CAD, audio and video editing, website and blog creation, customer relationship management, point of sale, networking and security, and alternatives to the Windows and Macintosh operating systems. You can save a substantial sum of money by replacing just a few commercial software titles with free and open source solutions.

Beginning Ubuntu for Windows and Mac Users

<http://www.apress.com/9781484206096?gtmf=s>

by Nathan Haines

ISBN13: 978-1-484206-09-6

244 Pages

Publication Date: September 2015

Available eBook Formats: EPUB, MOBI, PDF



Want to move to an open source operating system? Or want to help someone else make that move? If you've struggled to find comprehensive guides or didn't even know where to start, this book will delight you. It teaches all newcomers to Ubuntu how to get the most out of Ubuntu for work, home, and play. Get started on your open source journey using this excellent beginner's guide.

OpenStack Trove

<http://www.apress.com/9781484212226?gtmf=s>

by Amrith Kumar and Douglas Shelley

ISBN13: 978-1-484212-22-6

336 Pages

Publication Date: August 2015

Available eBook Formats: EPUB, MOBI, PDF



Your step-by-step guide to set up and run a secure and scalable cloud Database as a Service (DBaaS) solution. The book shows you how to set up and configure the Trove DBaaS framework, use prepackaged or custom database implementations, and provision and operate a variety of databases—including MySQL, PostgreSQL, MongoDB, Cassandra, and Redis—in development and production environments.

Authors Amrith Kumar and Douglas Shelley, both active technical contributors to the Trove project, describe common deployment scenarios and walk you through the set-up, configuration, and ongoing management of complex database topics like replication, clustering, and high availability. The book provides detailed descriptions of how Trove works and gives you an in-depth understanding of its architecture.

No Starch

Automate the Boring Stuff with Python

<https://www.nostarch.com/automatestuff>

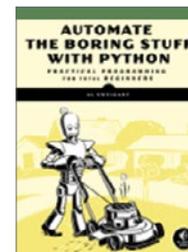
by Al Sweigart

ISBN: 978-1-59327-599-0

504 Pages

Publication Date: April 2015

Available eBook Formats: EPUB, MOBI, PDF



Why should you spend hours (or weeks or months!) doing tedious computer tasks when you can write a program to do the dirty work for you? *Automate the Boring Stuff with Python* assumes no prior programming experience—all you need is a mundane task to “automate” and the desire to dive into Python basics. Python programs can effortlessly accomplish tasks like splitting, merging, watermarking and encrypting PDFs, updating Excel spreadsheets, and filling out online forms. *Automate the Boring Stuff with Python* can streamline anyone's work day and free them from time-consuming computer tedium.

To read more from Al Sweigart, author of *Automate the Boring Stuff with Python*, visit his Opensource.com author page [2].

Teach Your Kids to Code

<https://www.nostarch.com/teachkids>

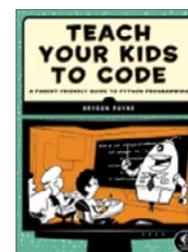
by Bryson Payne

ISBN: 978-1-59327-614-0

336 Pages

Publication Date: April 2015

Available eBook Formats: EPUB, MOBI, PDF



Even if don't know your loops from your variables, with *Teach Your Kids to Code*, you can walk your child through fundamental programming concepts with Python. The book employs step-by-step explanations and visual and game-oriented examples to grab kids' attention while strengthening their problem-solving and computational thinking skills. As you both work through the book, you'll learn how to build games (like Yahtzee and Pong), make a program that encodes and decodes messages, draw colorful shapes with Turtle graphics, and more.

To read more from Bryson Payne, author of *Teach Your Kids to Code*, visit his Opensource.com author page [3].

SparkFun Guide to Processing<https://www.nostarch.com/sparkfunprocessing>

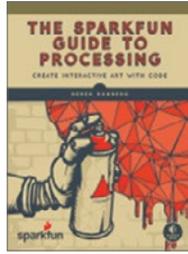
by Derek Runberg

ISBN: 978-1-59327-612-6

312 Pages

Publication Date: August 2015

Available eBook Formats: EPUB, MOBI, PDF



The *Sparkfun Guide to Processing* uses the artist-friendly programming language Processing to unleash your creativity through interactive art and games. You'll get a primer on animation and programming, and then work your way through a series of step-by-step, hands-on projects, including pixel art, a playable maze game, and psychedelic visualizations that respond to light, sound, and temperature.

The Official ScratchJr Book<https://www.nostarch.com/scratchjr>

by Marina Umaschi Bers and Mitchel Resnick

ISBN: 978-1-59327-671-3

160 Pages

Publication Date: October 2015

Available eBook Formats: EPUB, MOBI, PDF



Scratch is already the standard for teaching kids to program, with more than 11 million users. Now, there's a new kid in town—ScratchJr! With this programming language designed for kids ages 5 to 7, young ones can get a head start on picking up programming fundamentals by creating interactive stories and games on Android and iPad tablets. Written by ScratchJr's creators, *The Official ScratchJr Book* contains several activities that build on one another, with each chapter culminating in a fun final project. Along the way, kids develop computational-thinking, problem-solving, and design skills. The book includes helpful tips for grown-ups, creative challenges to take the learning further, and suggestions for connecting the activity with literacy and math concepts.

Doing Math with Python<https://www.nostarch.com/doingmathwithpython>

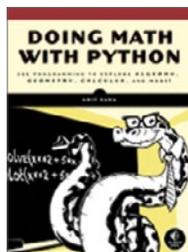
by Amit Saha

ISBN: 978-1-59327-640-9

264 Pages

Publication Date: August 2015

Available eBook Formats: EPUB, MOBI, PDF



Doing Math with Python is the perfect book for anyone who loves, or studies, math. Doing Math with Python shows you how to use Python to delve into high school-level math topics, such as statistics, geometry, probability, and calculus.

Creative coding challenges and applied examples will put your new math and coding skills into practice. You'll write an inequality solver, plot gravity's effect on how far a bullet will travel, shuffle a deck of cards, estimate the area of a circle by throwing 100,000 "darts" at a board, explore the relationship between the Fibonacci sequence and the golden ratio, and more. Along the way, you'll discover new ways to explore math and gain valuable programming skills that you'll use throughout your study of math and computer science.

O'Reilly Media**Fluent Python**<http://shop.oreilly.com/product/0636920032519.do>

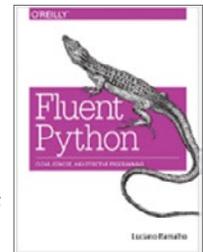
by Luciano Ramalho

ISBN: 978-1-4919-4600-8

770 Pages

Publication Date: July 2015

Available eBook Formats: DAISY, EPUB, MOBI, PDF



Python's simplicity lets you become productive quickly, but this often means you aren't using everything it has to offer. With this hands-on guide, you'll learn how to write effective, idiomatic Python code by leveraging its best—and possibly most neglected—features. Author Luciano Ramalho takes you through Python's core language features and libraries, and shows you how to make your code shorter, faster, and more readable at the same time. Many experienced programmers try to bend Python to fit patterns they learned from other languages, and never discover Python features outside of their experience. With this book, those Python programmers will thoroughly learn how to become proficient in Python 3.

Learning Spark<http://shop.oreilly.com/product/0636920028512.do>

by Holden Karau, Andy Konwinski,

Patrick Wendell, Matei Zaharia

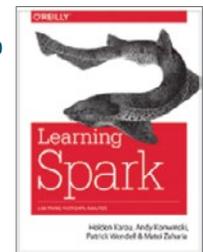
ISBN: 978-1-4493-5862-4

276 Pages

Publication Date: January 2015

Available eBook Formats: DAISY, EPUB, MOBI,

PDF



Data in all domains is getting bigger. How can you work with it efficiently? Recently updated for Spark 1.3, this book introduces Apache Spark, the open source cluster computing system that makes data analytics fast to write and fast to run. With Spark, you can tackle big datasets quickly through simple APIs in Python, Java, and Scala. This edition includes new information on Spark SQL, Spark

Streaming, setup, and Maven coordinates. Written by the developers of Spark, this book will have data scientists and engineers up and running in no time. You'll learn how to express parallel jobs with just a few lines of code, and cover applications from simple batch jobs to stream processing and machine learning.

Git for Teams

<http://shop.oreilly.com/product/0636920034520.do>

by Emma Jane Hogbin Westby

ISBN: 978-1-4919-1118-1

356 Pages

Publication Date: August 2015

Available eBook Formats: DAISY, EPUB, MOBI, PDF



You can do more with Git than just build software. This practical guide delivers a unique people-first approach to version control that also explains how using Git as a focal point can help your team work better together. You'll learn how to plan and pursue a Git workflow that not only ensures that you accomplish project goals, but also fits the immediate needs and future growth of your team. The first part of the book on structuring workflow is useful for project managers, technical team leads, and CTOs. The second part provides hands-on exercises to help developers gain a better understanding of Git commands.

Living Clojure

<http://shop.oreilly.com/product/0636920034292.do>

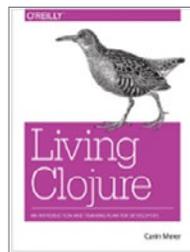
by Carin Meier

ISBN: 978-1-4919-0904-1

242 Pages

Publication Date: April 2015

Available eBook Formats: DAISY, EPUB, MOBI, PDF



If you're an experienced programmer who has not worked with Clojure before, this guide is the perfect thorough but gentle introduction for you. Author Carin Meier not only provides a practical overview of this JVM language and its functional programming concepts, but also includes a complete hands-on training course to help you learn Clojure in a structured way. The first half of the book takes you through Clojure's unique design and lets you try your hand at two Clojure projects, including a web app. The holistic course in second half provides you with critical tools and resources, including ways to plug into the Clojure community.

Docker Cookbook

<http://shop.oreilly.com/product/0636920036791.do?sortBy=publicationDate>

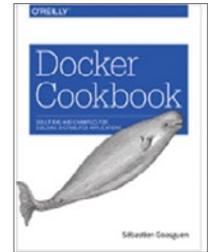
by Sébastien Goasguen

ISBN: 978-1-4919-1971-2

366 Pages

Publication Date: November 2015

Available eBook Formats: DAISY, EPUB, MOBI, PDF



Whether you're deploying applications on premise or in the cloud, this cookbook provides developers, operators, and IT professionals with more than 130 proven recipes for working with Docker. With these practical solutions, experienced developers with no previous knowledge of Docker will be able to package and deploy distributed applications within a couple of chapters. IT professionals will be able to solve everyday problems, as well as create, run, share, and deploy Docker images. Operators will quickly be able to adopt the tools that will change the way they work.

Packt

Python Machine Learning

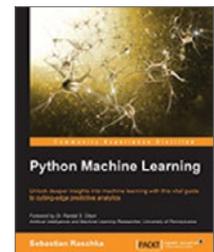
https://www.packtpub.com/big-data-and-business-intelligence/python-machine-learning/?utm_source=opensource&utm_medium=referral

by Sebastian Raschka

ISBN: 9781783555130

454 Pages

Publication Date: September 2015



Visit just about any tech or business website from the past decade and you'll find someone talking about Big Data. But whatever "revolution" took place within business over the past few years is well and truly over—Big Data is now the norm. In 2015, it's all about what you do with your data. And that's where machine learning comes in, allowing you to model and analyze your data in ways that help you predict future outcomes and behavior. In a global economy where the margins between success and failure are small and the future looks just a little bit frightening, the control that machine learning insights can bring will give you confidence that the world around you isn't as impenetrable as you first thought.

Python Machine Learning connects directly to this trend, providing an accessible and practical route through the topic with one of the world's most popular programming languages. But its accessibility doesn't mean it lacks depth—it's comprehensive, too, guiding readers through some of the latest Python machine learning libraries, such as Theano and Keras. It also takes in a number of different applications, from sentiment analysis to neural networks. At Packt, we see the book not only as further evidence of Python's popularity, but also indicative of a technical workforce interested in delving in and out of a range of problems quickly.

Go Programming Blueprints

https://www.packtpub.com/application-development/go-programming-blueprints/?utm_source=opensource&utm_medium=referral

by Mat Ryer

ISBN: 9781783988020

274 Pages

Publication Date: January 2015



Go, in many ways, echoes Python. Like Python, Go is an expressive multipurpose language, demonstrating a current trend for languages that are capable of providing rapid solutions to a wide range of programming challenges. One of the most interesting things about Go is that everything about it is very deliberate. Perhaps this is unsurprising for a programming language that has been developed by Google, an organization that has made matters of detail and engineering a core part of their brand, but it's still worth thinking about in the context of the future of open source, in terms of what it means and what it will look like for programmers.

Go's elegant and expressive syntax isn't something that should be learned, but instead should be experienced. That's where we started with *Go Programming Blueprints*. Exploring Go through a number of different projects, including a chat application (using web-sockets) and a random recommendation service (you'll have to look inside to find out what that looks like), the book demonstrates how Go makes building for scalability and performance simple. At a time when these issues have never been more important in the industry, we believe that it's only by diving straight into these problems that we can properly tackle them—and build better software.

Machine Learning with Spark

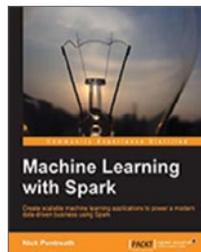
https://www.packtpub.com/big-data-and-business-intelligence/machine-learning-spark/?utm_source=opensource&utm_medium=referral

by Nick Pentreath

ISBN: 9781783288519

338 Pages

Publication Date: February 2015



We've already seen that machine learning is becoming the area that's defining data science as we know it—from Big Data to small data, from data mining to visualization, at the core of all these elements is the ability to learn and predict

from data. But with Spark, we're looking at the future of machine learning—its speed and processing power mean it's the tool that's redefining what's possible with Big Data. Spark's in-memory data storage is perhaps one of its most exciting features, but it's only once you've seen it in action, processing data across clusters, that you can begin to get a sense of how it can be used for high-velocity data analysis.

Machine Learning with Spark really communicates the full extent of what's possible with Spark—at Packt we know that technological innovation is really driven by the triumvirate of speed, efficiency, and size, and that's certainly what makes Spark such a big deal for us. The book does look like a standard machine learning guide, but because of Spark's integrated offering, you'll find chapters that explore data processing and preparation, as well as a chapter dedicated to real-time machine learning, the field where Spark really stands out.

React.js Essentials

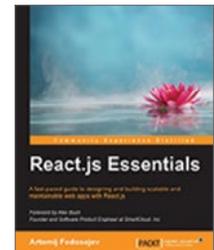
https://www.packtpub.com/web-development/reactjs-essentials/?utm_source=opensource&utm_medium=referral

by Artemij Fedosejev

ISBN: 9781783551620

208 Pages

Publication Date: August 2015



In 2014, we were looking toward MEAN [4] as the crucial full-stack web development solution. But that's started to change, with the rise of unbelievably lightweight tools that bridge the divide between framework and library. React.js rises above all others. Created by Facebook, React.js is now being used by some of the biggest tech companies in the world, including AirBnB and Instagram. The fact that these organizations depend upon the great UI and user experience of their apps and websites is really a testament to React's power. The reason why React is so impressive is simple—it lets you build UI components that immediately update with data changes. It lives up to its name—React reacts as soon as it needs to.

Inspired by React's rapid response time, *React.js Essentials* responds to the web development world's current love for components. Offering an accessible and fast-paced journey through the world of React.js, it underlines just why React is becoming more and more popular. Perfect for web developers interested in the cutting-edge of their field, the book looks set to remain relevant throughout 2016, as the world plays catch up with those most committed to and driven by innovation.

Python Data Science Essentials

https://www.packtpub.com/big-data-and-business-intelligence/python-data-science-essentials/?utm_source=open-source&utm_medium=referral

by Alberto Boschetti, Luca Massaron

ISBN: 9781785280429

258 Pages

Publication Date: April 2015



If Python is the one language that could lay a claim to defining programming in 2015 (yes, there are some close contenders, but no other language is quite so ubiquitous as Python), it's only fair that we return to where we started and let another of our most popular Python titles have the final word.

Python Data Science Essentials opens up a whole new approach to data science. The book was developed, like many Packt titles, with accessibility in mind. This doesn't just mean it's "easy to follow" (although, if you must know, it is). It means, more specifically, that the book offers a range of readers—whether you're a web developer working with Python or a data scientist that's always worked with R, but is now curious about Python—a simple way to explore Python's data science capabilities. Everyone wants to better understand data. *Python Data Science Essentials* shows you that in-depth insight doesn't have to be difficult.

Wiley

OpenStack Cloud Application Development

<http://www.wiley.com/WileyCDA/WileyTitle/productCd-1119194318,descCd-buy.html>

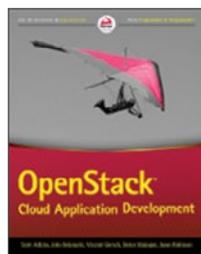
by Scott Adkins, John Belamaric, Vincent Giersch, Denys Makogon, Jason E. Robinson

ISBN: 978-1-119-19431-6

168 Pages

Publication Date: November 2015

Available eBook Formats: EPUB, PDF



OpenStack Cloud Application Development is a fast-paced, professional book for OpenStack developers, delivering comprehensive guidance without wasting time on development fundamentals. Written by experts in the OpenStack community from Infoblox, Gigaspaces, GoDaddy, and Comcast, this book shows you how to work effectively and efficiently within the OpenStack platform to develop large, scalable applications without worrying about underlying hardware. Follow along with an OpenStack build that illustrates how and where each technology comes into play, as you learn expert tips and

best practices that make your product stronger. Coverage includes OpenStack service primitives, networking within the OpenStack Ecosystem, deployment of Virtualized Network Functions for Enterprises, containers, data protection, and much more.

Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data

<http://www.wiley.com/WileyCDA/WileyTitle/productCd-111887613X.html>

by EMC2 Education Services

ISBN: 978-1-118-87613-8

432 Pages

Publication Date: January 2015

Available eBook Formats: EPUB, PDF



Data Science and Big Data Analytics is about harnessing the power of data for new insights. The book covers the breadth of activities and methods and tools that data scientists use. The content focuses on concepts, principles, and practical applications that are applicable to any industry and technology environment, and the learning is supported and explained with examples that you can replicate using open-source software.

Linux Bible, 9th Edition

<http://www.wiley.com/WileyCDA/WileyTitle/productCd-1118999878,descCd-buy.html>

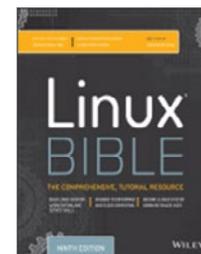
by Christopher Negus

ISBN: 978-1-118-99987-5

912 Pages

Publication Date: April 2015

Available eBook Formats: EPUB, PDF



Linux Bible is the ultimate hands-on Linux user guide, whether you're a true beginner or a more advanced user navigating recent changes. This updated ninth edition covers the latest versions of Red Hat Enterprise Linux 7 (RHEL 7), Fedora 21, and Ubuntu 14.04 LTS, and includes new information on cloud computing and development with guidance on OpenStack and Cloudforms. With a focus on RHEL 7, this practical guide gets you up to speed quickly on the new enhancements for enterprise-quality file systems, the new boot process and services management, firewalld, and the GNOME 3 desktop. Written by a Red Hat expert, this book provides the clear explanations and step-by-step instructions that demystify Linux and bring the new features seamlessly into your workflow.

Professional Python

<http://www.wiley.com/WileyCDA/WileyTitle/productCd-1119070856.html>

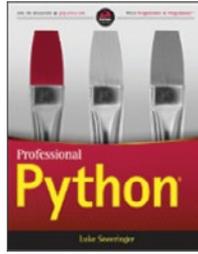
by Luke Sneeringer

ISBN: 978-1-119-07085-6

288 Pages

Publication Date: October 2015

Available eBook Formats: EPUB, PDF



Professional Python goes beyond the basics to teach beginner- and intermediate-level Python programmers the little-known tools and constructs that build concise, maintainable code. Design better architecture and write easy-to-understand code using highly adoptable techniques that result in more robust and efficient applications. Coverage includes decorators, context managers, magic methods, class factories, metaclasses, regular expressions, and more, including advanced methods for unit testing using asyncio and CLI tools. Each topic includes an explanation of the concept and a discussion on applications, followed by hands-on tutorials based on real-world scenarios. All sample code is available for download from the companion website, and the “Python 3 first” approach covers multiple current versions, while ensuring long-term relevance.

Links

- [1] <http://www.informit.com/authors/bio/ea8b25c7-a974-480e-a723-991b1fdd70da>
- [2] <https://opensource.com/users/alsweigart>
- [3] <http://opensource.com/users/professorpayne>
- [4] <http://mean.io/#/>

Reliable JavaScript: How to Code Safely in the World's Most Dangerous Language

<http://www.wiley.com/WileyCDA/WileyTitle/productCd-1119028728.html>

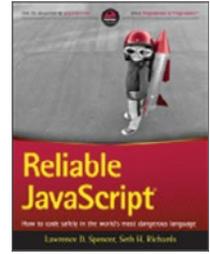
by Lawrence Spencer, Seth Richards

ISBN: 978-1-119-02872-7

528 Pages

Publication Date: July 2015

Available eBook Formats: EPUB, PDF



Reliable JavaScript: How to Code Safely in the World's Most Dangerous Language demonstrates how to create test-driven development for large-scale JavaScript applications that will stand the test of time and stay accurate through long-term use and maintenance. Taking a test-first approach to software architecture, this book walks you through several patterns and practices and explains what they are supposed to do by having you write unit tests. Write the code to pass the unit tests, so you not only develop your technique for structuring large-scale applications, but you also learn how to test your work. You'll come away with hands-on practice that results in code that is correct from the start, and has the test coverage to ensure that it stays correct during subsequent maintenance. All code is provided both in the text and on the web, so you can immediately get started designing more complete, robust applications.

Author

Rikki Endsley is a community manager for [Opensource.com](http://opensource.com). She has worked as the community evangelist on the Open Source and Standards (OSAS) team at Red Hat; a freelance tech journalist; community manager for the USENIX Association; associate publisher of *Linux Pro Magazine*, *ADMIN*, and *Ubuntu User*; and the managing editor of both *Sys Admin* magazine and UnixReview.com. Follow her on Twitter [@rikkiends](https://twitter.com/rikkiends).

8 books to make you a more open leader

BY BRYAN BEHRENSHAUSEN

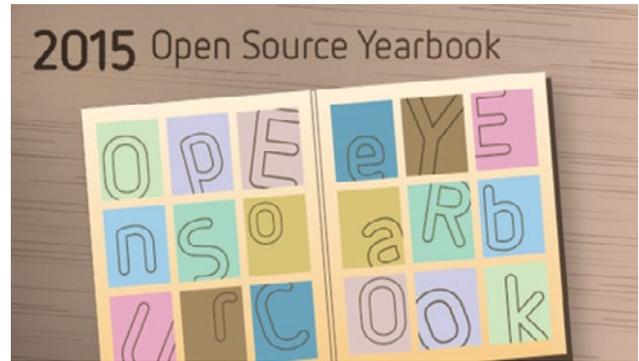
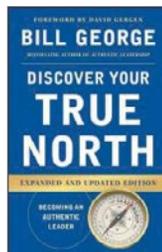
Open source principles continue to invade new domains, and in 2015 we saw this trend accelerate. Traditional organizational structures are giving way to new models—more open models—and the nature of our collaborations is changing. Rapidly.

When Jim Whitehurst published *The Open Organization* in June, he crystallized something that's been nascent for years: a new management paradigm, one fit for the 21st century. Leaders everywhere are scrambling to understand their roles under this paradigm, and our community has come to their aid with recommendations for books that will guide them across uncertain terrain. If you've resolved to become a more open leader in 2016, then you'll want to start here.

Discover Your True North

by Bill George (Recommended by Sam Knuth [1])

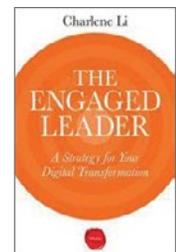
Bill George has written extensively on authenticity in leadership, which includes transparency, honesty, and openness. *Discover Your True North* [2] draws on George's personal experiences with many authentic leaders across many industries, pulling together the themes that contribute to their successes and failures. Consistently, the same principles Jim explores in *The Open Organization* [3] emerge as critical leadership traits in the experiences of both George and the leaders he profiles. *Discover Your True North* is not only a great collection of case studies and experiences but also a guide to practicing authentic, open leadership for people at any stage in their careers.



The Engaged Leader

by Charlene Li (Recommended by Jason Hibbets [4])

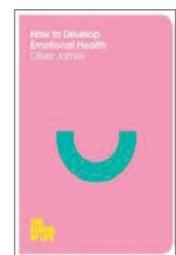
Using social media to engage and listen isn't a science. It's an art. And in *The Engaged Leader* [5], Li helps you navigate these waters with examples from different industries and from leaders at various levels in their organizations. I've recommended this book to several of my colleagues at Red Hat (and beyond) who want to use social media and other digital tools to be more engaged with their customers, people they lead, and especially other leaders. Li provides practical stories and tips to help you to adapt to this new era of digital engagement.



How to Develop Emotional Health

by Oliver James (Recommended by Laura Hilliger [6])

A quick and fun read, *How to Develop Emotional Health* [7] will help you identify personal hang-ups that may stop you from being an open and engaged leader. James offers solutions and insights on how to work through your past, present, and future to lead a mentally healthy, happy, and productive life—touchstones for good leaders and, indeed, “open sourcers” in general.



Rework

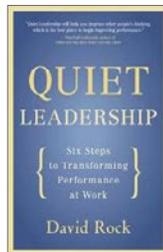
by Jason Fried & David Heinemeier Hansson
(Recommended by Marten Mickos [8])

Rework [9] examines organizing work in the modern world, where people work from home and largely know what to do. It's full of pithy, illustrated advice ("Pick a fight." "Meetings are Toxic." "Inspiration if Perishable.") for anyone looking to revamp the workplace. Although the authors may not use the term, that openness is an underlying principle of this book is easy to see.

**Quiet Leadership**

by David Rock (Recommended by
Rebecca Fernandez [10])

Whether you're a manager, a mentor, or both, you want to coach and give feedback in a way that promotes another person's growth and contributes to autonomous decision-making. Through compelling neuroscience, *Quiet Leadership* [11] shows why leaders should stop giving advice and offering up ideas. Learn how to listen more, talk less, and ask the kinds of questions that help people find their own solutions.

**The Open Organization**

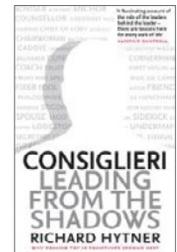
by Philip A. Foster (Recommended by
Bryan Behrenshausen [12])

Before *The Open Organization* by Jim Whitehurst was *The Open Organization* [13] by Philip A. Foster. Although Jim admits that his book isn't management theory ("I'll leave that to the academics," he says), Foster's is unabashedly so. Published in 2014, *The Open Organization* is quite frankly the textbook on what both authors call a "new management paradigm." Meticulously researched and annotated, Foster's volume draws on sources running the gamut from management theorists to practitioners. And as a neat bonus, Foster offers a compelling case study: GitHub, whose mission to "create awesome" prompted interesting organizational choices. For enthusiasts, it's a moment of precious insight into daily life at a firm embracing the open source mission. For leaders, it's simply required reading.

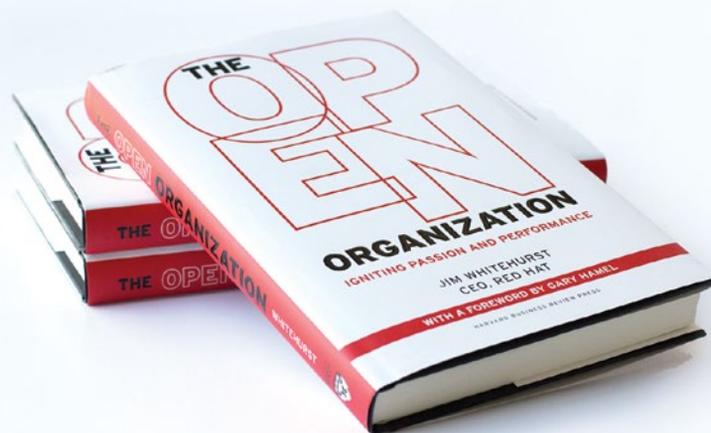
**Consiglieri**

by Richard Hytner (Recommended by
Laura Hilliger [6])

Although *Consiglieri* [14] weighs in heavily on the "evil corporate" examples and name-dropping, it contains some wonderful undertones about being a leader who guides rather than directs. The so-called



How can open source values transform the way we work, manage, and lead?

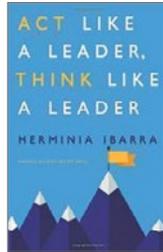


Join the conversation at:
opensource.com/open-organization

“second in command” is all of us in any given conversation. Knowing to respect and wield that power can help make you an influential leader in distributed networks.

Act Like a Leader, Think Like a Leader

by Herminia Ibarra (Recommended by Jeff Mackanic [15])



In *The Open Organization* Jim Whitehurst writes: “What I need to do is create the context for Red Hat associates so they can do their best work. My goal is to get people to believe in the mission and then create the right structures that empower them to achieve what once might have been impossible.” In *Act Like a Leader, Think Like a Leader* [15], Ibarra offers advice on empowering leaders this way. She writes that in order to act like leaders, we will have to devote much of our time to the following practices: bridging across diverse people and groups, envisioning new possibilities, engaging people in the change process, and embodying the change. Ibarra then provides detailed ideas for ways leaders can grow in each of these areas. This book is full of insights, and can certainly help you become a more open leader.

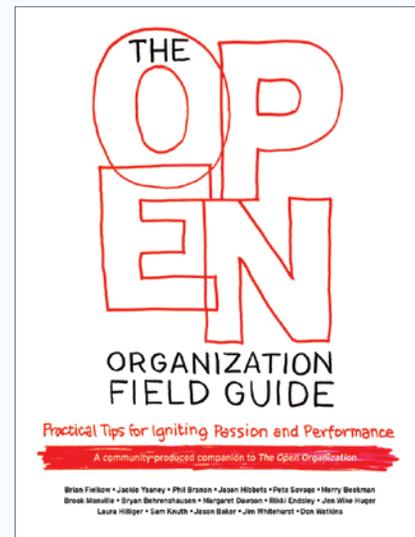
Links

- [1] Sam Knuth: <https://opensource.com/users/samfw>
- [2] Discover Your True North: <http://discoveryourtruenorth.org/>
- [3] The Open Organization: <https://opensource.com/open-organization>
- [4] Jason Hibbets: <https://opensource.com/users/jhibbets>
- [5] The Engaged Leader: <http://wdp.wharton.upenn.edu/book/engagedleader/>
- [6] Laura Hillinger: <https://opensource.com/users/laurahillinger>
- [7] How to Develop Emotional Health: <http://www.theschooloflife.com/shop/how-to-develop-emotional-health-oliver-james/>
- [8] Marten Mickos: <https://opensource.com/users/martenmickos>
- [9] Rework: <https://37signals.com/rework/>
- [10] Rebecca Fernandez: <https://opensource.com/users/rebecca>
- [11] Quiet Leadership: <http://www.quietleadership.com/public/content.aspx?Page=index&AspxAutoDetectCookieSupport=1>
- [12] Bryan Behrenshausen: <https://opensource.com/users/bbehrens>
- [13] The Open Organization: <http://www.ashgate.com/default.aspx?page=637&calcTitle=1&isbn=9781472440112&lang=cy-gb>
- [14] Consiglieri: <http://www.consiglieribook.com/>
- [15] Jeff Mackanic: <https://opensource.com/users/mackanic>
- [16] Act Like a Leader, Think Like a Leader: <http://herminiaibarra.com/books/>

Author

Bryan Behrenshausen has been a member of the [Opensource.com](https://opensource.com) team since 2011. When he’s not thinking or writing about all things open source, he’s playing retro video games or reading classic science fiction. In 2015, he earned his PhD in Communication from UNC, Chapel Hill. Around the Net, he goes by the nickname “semioticrobotic.”

**The conversation continues
in this community-produced
companion to
*The Open Organization.***



Download your free copy at:
opensource.com/open-organization/resources/field-guide



Most Playful

Best open source games of 2015

.....BY ROBIN MULWIJK

2015 has been a great year for open source and Linux gaming. The number of Linux games continues to grow [1], and Steam Machines [2] finally hit the market on November 10. I'd like to give you my picks for the top five open games of 2015. Which one is your favorite? And which games do you think should have made the list?

5. SuperTuxKart

At number five we have SuperTuxKart [3], a hugely popular racing game.

SuperTuxKart's gameplay is similar to Mario Kart. What makes it unique is that it is open source, and its characters are open source mascots, with Mozilla Thunderbird being the race referee. A major update for SuperTuxKart was its 0.9 release in April 2015. This release included a highly modified version of Irrlicht [4], including an entirely new graphics renderer dubbed Antarctica [5], which enabled better graphics appearance and features such as dynamic lighting, ambient occlusion, depth of field, global illumination, and more. The game has been downloaded more than 2 million times so far. This number applies to the fork of TuxKart, and also when Joerg Henrichs resurrected the project with the help of Eduardo Hernandez Munoz. More of the game's history is available on Wikipedia [6].

Credit and copyright SuperTuxKart blog.



4. Magarena

At number four comes Magarena [7], a single-player fantasy card game.

Based on Magic: The Gathering [8], the game lets you play Magarena against a computer opponent. Magarena includes

an advanced AI, intuitive interface, engaging gameplay, and program stability.

Credit and copyright Magarena website.



3. Battle of Wesnoth

With number three, we head into a classic, Battle of Wesnoth [9].

Gamers that prefer a turn-based tactical strategy game will love Battle for Wesnoth. Making this game unique is its high fantasy theme. The game allows you to build your own army, out of 200 unit types. It also includes 16 races, 6 major factions, and hundreds of years of history. As player, you are the heir to Wesnoth, and you fight to regain its throne. "The world of Wesnoth is absolutely huge and only limited by your creativity—make your own custom units, create your

Credit and copyright Battle of Wesnoth website.



own maps, and write your own scenarios or even full-blown campaigns.”

2. 0 A.D.

Switching to a historical real time strategy game, at number two, we have 0 A.D [10].

Although you can already download 0 A.D., this game is in alpha; still, it is playable, and available as native client for Linux. 0 A.D. is unique in its graphics and rendering, backed by positive acclaim. As leader of an ancient civilization, your goal is to gather the resources you need to raise a military force and dominate your enemies. Your civilizations and battles take part over the millennium of 500 B.C. to 500 A.D. Hence, the name of this game, being the midpoint: 0 A.D.

Credit play0ad.com, license CC-BY-SA.



1. Minetest

That leaves us with number one, Minetest [11], an open source alternative to Minecraft.

Minetest’s gameplay is similar to Minecraft’s. You can create and remove various types of blocks in a 3D open world. Minetest supports singleplayer and multiplayer games. Minetest also includes many features, such as support for mods, texture packs, and more. Minetest is released under the GNU Lesser General Public License.

Credit Minetest, license CC BY-SA 3.0 Unported.



Honorable mentions

My top 5 is not a complete list of great open games. For example, we covered some interesting and fun

games in our weekly round-ups, including:

- Relic Hunters Zero [12] is a nice game in which you get to shoot evil space ducks in the face with tiny cute guns. “It’s fast, it’s tactical, and feels delicious and smooth to play.”
- Scrabble3D [13] is open source, based on a GPLv3 license. You can play against the computer, against players on your local network, or connect to a game server online. The real fun in this game is that you can create words in three directions instead of just two.
- OpenTTD [14] is an open source simulation game based upon Transport Tycoon Deluxe. As entrepreneur, your goal is to make as much profit as possible by transporting passengers and various goods by road, rail, sea, and air.
- SuperTux [15] is a classic 2D jump’n run sidescroller game in a style similar to the original Super Mario games covered under the GNU GPL.

You can also find a list of open source games and projects on GitHub [16].



Links

- [1] Steam database: <https://steamdb.info/>
- [2] Steam machines: <http://store.steampowered.com/universe/machines/>
- [3] SuperTuxKart: <http://supertuxkart.blogspot.co.uk/>
- [4] Irrlicht: <http://irrlicht.sourceforge.net/>
- [5] Antarctica: http://supertuxkart.sourceforge.net/Antarctica:_Overview
- [6] SuperTuxKart history: <https://en.wikipedia.org/wiki/SuperTuxKart>
- [7] Magarena: <https://magarena.github.io/>
- [8] Magic the Gathering: <http://magic.wizards.com/>
- [9] Battle of Wesnoth: <http://www.wesnoth.org/>
- [10] 0 A.D.: <http://play0ad.com/>
- [11] Minetest: <http://www.minetest.net/>
- [12] Relic Hunters Zero: <http://zueiradigital.com/presskit/rhz/>
- [13] Scrabble3D: <http://scrabble.sourceforge.net/wiki/>
- [14] OpenTTD: <https://www.openttd.org/en/>
- [15] SuperTux: <http://supertuxproject.org/>
- [16] Games on GitHub: <https://github.com/leereilly/games>

Author

Robin Muilwijk is an advisor on Internet and e-Government. He also serves as a community moderator for Opensource.com, and on the board for the eZ Publish community. Follow him on Twitter @i_robin.

10 tools for visual effects in Linux with Kdenlive

BY SETH KENLON

Kdenlive [1] is one of those applications: You can use it daily for a year and wake up one morning only to realize that you still have only grazed the surface of all of its potential. That's why it's nice every once in a while to sit back and look over some of the lesser-used tricks and tools in Kdenlive. Even though something's not used as often as, say, the Spacer or Razor tools, it still may end up being just the right finishing touch on your latest masterpiece.

Most of the tools I'll discuss here are not officially part of Kdenlive; they are plugins from the Frei0r [2] package. These are ubiquitous parts of video processing on Linux and Unix, and they usually get installed along with Kdenlive as distributed by most Linux distributions, so they often seem like part of the application. If your install of Kdenlive does not feature some of the tools mentioned here, make sure that you have Frei0r plugins installed.

Because many of the tools in this article affect the look of an image, here is the base image, without effects or adjustment:

Let's get started.

*Still image grabbed from a video by Footage Firm, Inc.
CC BY-SA 4.0.*



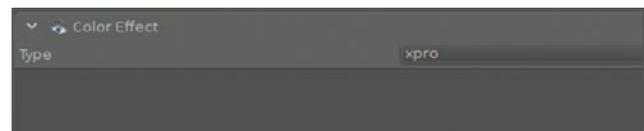
1. Color effect

You can find the **Color Effect** filter in **Add Effect > Misc** context menu. As filters go, it's mostly just a preset; the only controls it has are which filter you want to use.

Normally that's the kind of filter I avoid, but I have to be honest: Sometimes a plug-and-play solution is exactly what you want. This filter has a few different settings, but the two that make it worth while (at least for me) are

the Sepia and XPro effects. Admittedly, controls to adjust how sepia tone the sepia effect is would be nice, but no matter what, when you need a quick and familiar color effect, this is the filter to throw onto a clip.

It's immediate, it's easy, and if your client asks for that look, this does the trick every time.



2. Colorize

The simplicity of the **Colorize** filter in **Add Effect > Misc** is also its strength. In some editing applications, it takes two filters and some compositing to achieve this simple color-wash effect. It's refreshing that in Kdenlive, it's a matter of one filter with three possible controls (only one of which, strictly speaking, is necessary to achieve the look).

Its use is intuitive; use the **Hue** slider to set the color. Use the other controls to adjust the luma of the base image as needed.

This is not a filter I use every day, but for ad spots, bumpers, dreamy sequences, or titles, it's the easiest and quickest path to a commonly needed look. Get a company's color,



use it as the colorize effect, slap a logo over the top of the screen, and you've just created a winning corporate intro.

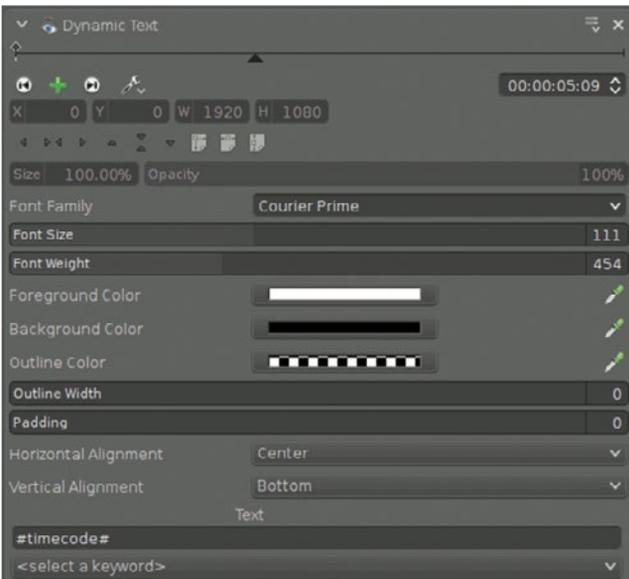


3. Dynamic Text

For the assistant editor, the **Add Effect > Misc > Dynamic Text** effect is worth the price of Kdenlive. With one mostly pre-set filter, you can add a running timecode burn-in to your project, which is an absolute must-have safety feature when round-tripping your footage through effects and sound.



The controls look more complex than they actually are.



The font settings are self-explanatory. Placement of the text is controlled by the Horizontal and Vertical Alignment settings; steer clear of the Size setting (it controls the size

of the “canvas” upon which you are compositing the burn-in, not the size of the burn-in itself).

The text itself doesn't have to be timecode. From the drop-down menu, you can choose from a list of useful text, including frame count (useful for VFX, since animators work in frames), source frame rate, source dimensions, and more.

You are not limited to just one choice. The text field in the control panel will take whatever arbitrary text you put into it, so if you want to burn in more information than just timecode and frame rate (such as **Sc 3 - #timecode# - #meta.media.0.stream.frame_rate#**), then have at it.

4. Luminance

The **Add Effect > Misc > Luminance** filter is a no-options filter. Luminance does one thing and it does it well: It drops the chroma values of all pixels in an image so that they are displayed by their luma values. In simpler terms, it's a grayscale filter.



The nice thing about this filter is that it's quick, easy, efficient, and effective. This filter combines particularly well with other related filters (meaning that yes, I'm cheating and including three filters for one).



Combining, in this order, the **RGB Noise** for emulated grain, **Luminance** for grayscale, and **LumaLiftGainGamma** for levels can render a textured image that suggests the classic look and feel of Kodak Tri-X [3] film.

5. Mask0mate

Better known as a four-point garbage mask, the **Add Effect > Alpha Manipulation > Mask0mate** tool is a quick, no-frills way to ditch parts of your frame that you don't need. There isn't much to say about it; it is what it is.

Image by Footage Firm, Inc.



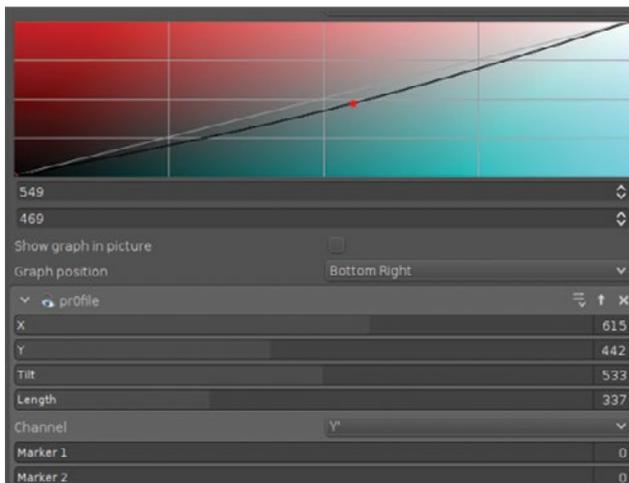
The confusing thing about the effect is that it does not imply compositing. You can pull in the edges all you want, but you won't see it unless you add the **Composite** transition to reveal what's underneath the clip

(even if that's nothing). Also, use the **Invert** function for the filter to act like you think it should act (without it, the controls will probably feel backward to you).



6. Pr0file

The **Add Effect > Misc > Pr0file** filter is an analytical tool, not something you would actually leave on a clip for final export (unless, of course, you do). Pr0file consists of two components: the Marker, which dictates what area of the image



is being analyzed, and the Graph, which displays information about the marked region.

Set the marker using the **X, Y, Tilt, and Length** controls. The graphical readout of all the relevant color channel information is displayed as a graph, superimposed over your image.

The readout displays a profile of the colors within the region marked. The result is a sort of hyper-specific vectorscope (or oscilloscope, as the case may be) that can help you zero in on problem areas during color correction, or compare regions while color matching.

In other editors, the way to get the same information was simply to temporarily scale your image up to the region you want to analyze, look at your readout, and then hit undo to scale back. Both ways work, but the Pr0file filter does feel a little more elegant.

7. Vectorscope

Kdenlive features an inbuilt vectorscope, available from the View menu in the main menu bar. A vectorscope is not a filter, it's just another view the footage in your Project Monitor, specifically a view of the color saturation in the current frame. If you are color correcting an image and you're not sure what colors you need to boost or counteract, looking at the vectorscope can be a huge help.



There are several different views available. You can render the vectorscope in traditional green monochrome (like the hardware vectorscopes you'd find in a broadcast control room), or a chromatic view (my personal preference), or subtracted from a color-wheel background, and more.

The vectorscope reads the entire frame, so unlike the Pr0file filter, you are not just getting a reading of one area in the frame. The result is a consolidated view of what colors are most prominent within a frame. Technically, the same sort of information can be intuited by several trial-and-error passes with color correction, or you can just leave your vectorscope open and watch the colors float along the color wheel and make adjustments accordingly.

Aside from how you want the vectorscope to look, there are no controls for this tool. It is a readout only.

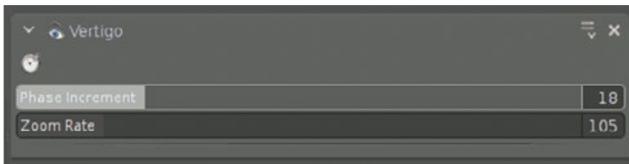
8. Vertigo

There's no way around it; **Add Effect > Misc > Vertigo** is a gimmicky special effect filter. So unless you're remaking

Fear and Loathing or the movie adaptation of Dead Island, you probably aren't going to use it that much; however, it's one of those high-quality filters that does the exact trick you want when you happen to be looking for it.

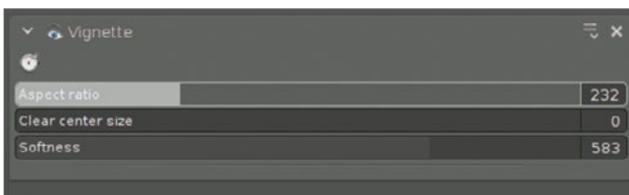


The controls are simple. You can adjust how distorted the image becomes and the rate at which it distorts. The overall effect is probably more drunk or vision-quest than vertigo, but it's good.



9. Vignette

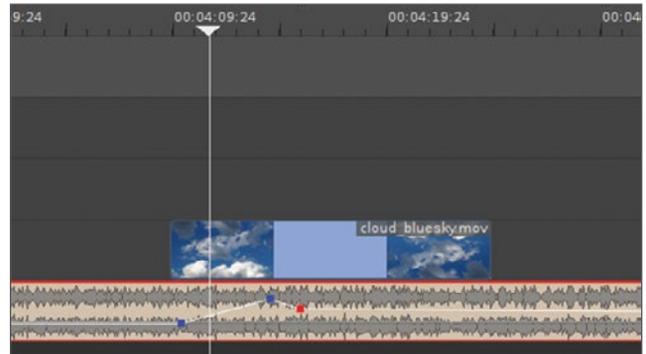
Another beautiful effect, the **Add Effect > Misc > Vignette** darkens the outer edges of the frame to provide a sort of portrait, soft-focus nouveau look. Combined with the Color Effect or the Luminance faux Tri-X trick, this can be a powerful and emotional look.



The softness of the border and the aspect ratio of the iris can be adjusted. The **Clear Center Size** attribute controls the size of the clear area, which has the effect of adjusting the intensity of the vignette effect.

10. Volume

I don't believe in mixing sound within the video editing application, but I do acknowledge that sometimes it's just necessary for a quick fix or, sometimes, even for a tight production schedule. And that's when the **Audio correction > Volume (Keyframable)** effect comes in handy.



The control panel is clunky, and no one really wants to adjust volume that way, so the effect is best when used directly in the timeline. To create a volume change, double-click the volume line over the audio clip, and then click and drag to adjust. It's that simple.

Should you use it? Not really. Sound mixing should be done in a sound mixing application. Will you use it? Absolutely. At some point, you'll get audio that is too loud to play as you edit, or you'll be up against a deadline without a sound engineer in sight. Use it judiciously, watch your levels, and get the show finished.

Everything else

This has been 10 (OK, 13 or 14) effects and tools that Kdenlive has quietly lying around to help your edits become great. Obviously there's a lot more to Kdenlive than just these little tricks. Some are obvious, some are cliché, some are obtuse, but they're all in your toolkit. Get to know them, explore your options, and you might be surprised what a few cheap tricks will get you.

Links

- [1] Kdenlive: <https://kdenlive.org/>
- [2] Frei0r package: <https://www.dyne.org/software/frei0r/>
- [3] Kodak Tri-X film: <http://www.kodak.com/global/en/professional/products/films/bw/triX2.jhtml>

Author

Seth Kenlon is an independent multimedia artist, free culture advocate, and UNIX geek. He is one of the maintainers of Slackrmedia, a Slackware-based multimedia production project.

Adafruit's best open source wearables of 2015

BY BECKY STERN

Wearable electronics have exploded in the past year. Countless small devices are now on the market for not only fitness tracking, but posture improvement, sunscreen reminders, muscle-sensing gesture control, and much more. As technology on the body becomes more pervasive than ever, having open source tools for developing wearable technology is more important than ever, so that we can create the future of fashion tech while maintaining data privacy of biometric sensor data.

Adafruit [1] makes open source hardware parts, tutorials, and source code for creating your own DIY electronics projects. Here are our favorite open source wearable products and projects from 2015.

FLORA v2 (and Getting Started with Adafruit FLORA)

The FLORA wearable microcontroller [2] got a hardware upgrade this year, adding a micro USB port and onboard programmable NeoPixel for prototyping color-changing, Arduino-compatible code before ever connecting it up with conductive thread or wire. 2015 also brought *Getting Started with Adafruit FLORA* [3], the beginner's guide to wearable microcontroller projects.



The book covers the basics of the FLORA family of parts, tools, and techniques, and three projects to get you started, including textile switches, dance-activated lights, and a DIY GPS watch. There's a FLORA Book Pack [4] available with parts used in the book, and more for building the dozens of other open source FLORA tutorials available on the Adafruit Learning System [5].

SMSsenger bag with FLORA + FONA

One recent project made with FLORA is the SMSsenger Bag [6] —an addressable LED matrix displaying text messages it receives to its onboard FONA open source cell phone module. Developing wearable devices that communicate wirelessly is tremendously fun and rewarding, and the open source Arduino libraries for FONA and NeoPixels make it simple for beginners, students, and hobbyists to get started creating personal area network wearables.

FLORA Bluefruit LE Module (and Bluefruit LE Connect for iOS/Android)

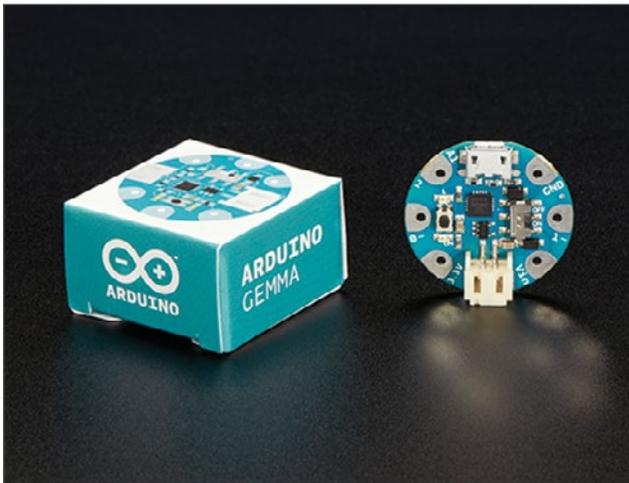
On the topic of networked wearables, the FLORA Bluefruit LE [7] debuted in 2015. This sewable Bluetooth low energy module makes it easy to connect wearables to your phone or tablet for controlling, for example, the color of the LEDs in your dress, and also pushing sensor data to cloud services like adafruit.io. Build around the Nordic nRF51822, the FLORA Bluefruit LE module can also act as an HID keyboard or UriBeacon. The Bluefruit family of boards works great with the (also open source [8]) Adafruit Bluefruit LE Connect apps for Android [9] and iOS [10].

Zac Posen “Made with Code” dress

This year, Google teamed up with Zac Posen as part of their Made with Code [11] initiative to create an LED dress with animating patterns coded by girls all over the world. Mad-dy Maxey helped bring the dress to life and documented its hardware construction [12] (using FLORA and NeoPixels) in an Instructable about the project, which debuted on the runway of New York Fashion Week [13].

Arduino GEMMA

FLORA's not the only board that saw a makeover this year—its little sister GEMMA got a micro USB port and on/off switch and debuted its new blue cousin, the official Arduino GEMMA [14]. This tiny board built around the ATtiny85 microcontroller can control NeoPixels, read simple sensors, and is small and low-cost enough to easily embed in wearables projects. While you can get GEMMA v2 in Adafruit or Arduino flavors, the official partnership provides native support for Arduino GEMMA in the Arduino software IDE, boxed packaging, and best of all supports Arduino.



Links

- [1] Adafruit: <https://www.adafruit.com/>
- [2] FLORA wearable microcontroller: <https://www.adafruit.com/products/659>
- [3] Getting Started with Adafruit FLORA: <https://www.adafruit.com/products/1839>
- [4] FLORA Book Pack: <https://www.adafruit.com/products/2404>
- [5] FLORA tutorials available on the Adafruit Learning System
- [6] SMSsenger bag: <https://learn.adafruit.com/smssenger-bag>
- [7] FLORA Bluefruit LE: <https://www.adafruit.com/products/2487>
- [8] Adafruit Industries on GitHub: https://github.com/adafruit?utf8%3D%25E2%259C%2593%26query%3Dbluefruit_le_connect
- [9] Adafruit Bluefruit LE Connect (Android): <https://play.google.com/store/apps/details?id=com.adafruit.bluefruit.le.connect>
- [10] Adafruit Bluefruit LE Connect (iOS): <https://itunes.apple.com/us/app/adafruit-bluefruit-le-connect/id830125974>
- [11] Made with Code: <https://www.madewithcode.com/>
- [12] Make a Runway Ready Programmable LED Matrix Dress: <http://www.instructables.com/id/Make-a-Runway-Ready-Programmable-LED-Matrix-Dress/?ALL-STEPS&sa=D&usg=AFQjCNFA7DchWaPAck-8jA8aLO5gef9RI0w>
- [13] How To Make a Runway Ready Programmable LED Matrix Dress: https://blog.adafruit.com/2015/09/16/how-to-make-a-runway-ready-programmable-led-matrix-dress-zac_posen-madewithcode-thecrated-wearablewednesday/
- [14] Arduino Gemma: <https://www.arduino.cc/en/Main/ArduinoGemma>

Author

Becky Stern is a DIY guru and Director of Wearable Electronics at Adafruit.



5 favorite 3D printing projects of 2015

BY HARRIS KENNY

The desktop 3D printing industry continues to grow at a rapid pace. Thanks to its deep-seated commitment to free software, libre innovation, and open source hardware [1], there is a proliferation of projects, products, and companies that also share these values.

This post highlights five of the top innovative 3D-printing-related projects in 2015, but with so much going on in the community there are bound to be great projects that were not included.

OpenBCI

OpenBCI [2] is dedicated to developing biosensing for everybody. Their flagship product, the Ultracortex, is capable of recording research-grade brain activity (EEG), muscle activity (EMG), and heart activity (ECG). OpenBCI effectively uses 3D printing for prototyping and production of their headsets and included components. Learn more about the Ultracortex Mark 3 (GNU-GPLv3 licensed) in the OpenBCI store [3].

Conor Russomanno of OpenBCO describes his latest project based on the OpenBCI open source brain-computer interface. Photo by old_school_maker. CC BY 2.0.



SatNOGS DB

SatNOGS [4] is a modular and scalable stack for Satellite Ground Station implementation. SatNOGS is driven by four sub-projects: SatNOGS Network, SatNOGS DB (launched

this summer), SatNOGS Client, and SatNOGS Ground Station. Their team uses 3D printing to prototype hardware components, specifically for use in ground stations. In 2014, SatNOGS won the prestigious and competitive 2014 Hackaday Prize [5].

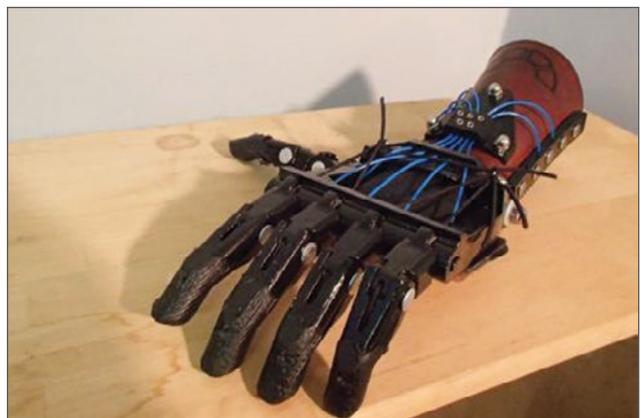
A SatNOGS version 2 ground station deployed during FOSDEM 2015. Photo by Nikos Roussos. CC BY 2.0.



e-NABLE

e-NABLE [6] works together to design, improve, and produce hand devices that empower the user. e-NABLE effectively created a low-cost, distributed way to give a helping

Talon hand, custom e-NABLE prosthetic hand



hand, and created some of the go-to designs to demonstrate the power of distributed manufacturing through 3D printing. Some of their most popular projects include The Raptor Hand [7] (BSD licensed) and the Talon Hand [8] (GNU-GPLv2 licensed). An exciting related emerging project is the HACKberry [9] by exiii (CC0 1.0 Universal and CC BY-SA 4.0 licensed), a 3D printed bionic hand.

Michigan Tech Open Sustainability Technology Lab (MOST)

MOST [10] is a research group known for conducting some of the most cutting edge research on 3D printing and open source hardware. Earlier this year, their team started to integrate a new set of recycling codes into all 3D printed polymer designs [11], enabling them to close the material loop in their lab. They also made great strides in low-cost metal printing, addressing the substrate release problem for metal printing [12]. Learn more about the newest design of their metal printer, code-named Magneto [13] (GNU-GPLv3 licensed).

*MOST's Magneto 3D printer.
Photo by J.M. Pearce. GNU Free Documentation License v 1.3.*



RepRap

RepRap [14] has been instrumental to the early and continued free/libre/open development of desktop 3D printing. The community continues to grow this year, with early developers like Josef Prusa's Prusa Research [15] growing, electronics company developer UltiMachine [16] advancing core technologies, software projects like Cura [17],

RepRap workshop in Gdańsk, Poland



OctoPrint [18], Slic3r [19], BotQueue [20], and MatterControl [21] continuing to grow, and rapid growth in the availability of 3D printing materials.

After a great 2015, projects like this inspire us about the future and what is possible with a collaborative approach that respects user freedom.

Links

- [1] What is open hardware?: <https://opensource.com/resources/what-open-hardware>
- [2] OpenBCI: <http://www.openbci.com/>
- [3] OpenBCI store: <http://openbci.myshopify.com/collections/frontpage/products/ultracortex-mark-3-eeg-headset-kit-assembly-required>
- [4] SatNOGS: <https://satnogs.org/>
- [5] SatNOGS wins the 2014 Hackaday Prize: <http://hackaday.com/2014/11/13/satnogs-wins-the-2014-hackaday-prize/>
- [6] e-NABLE: <http://enablingthefuture.org/>
- [7] Raptor Hand: <http://enablingthefuture.org/upper-limb-prosthetics/the-raptor-hand/>
- [8] Talon Hand: <http://enablingthefuture.org/upper-limb-prosthetics/talon-hand/>
- [9] HACKberry: <http://exiii-hackberry.com/>
- [10] Michigan Tech Open Sustainability Technology Lab (MOST): <http://www.mse.mtu.edu/~pearce/Index.html>
- [11] Polymer recycling codes for distributed manufacturing with 3-D printers: http://www.appropedia.org/Polymer_recycling_codes_for_distributed_manufacturing_with_3-D_printers
- [12] In Situ Formation of Substrate Release Mechanisms for Gas Metal Arc Weld Metal 3-D Printing: http://www.appropedia.org/In_Situ_Formation_of_Substrate_Release_Mechanisms_for_Gas_Metal_Arc_Weld_Metal_3-D_Printing
- [13] Integrated Voltage—Current Monitoring and Control of Gas Metal Arc Weld Magnetic Ball-Jointed Open Source 3-D Printer: http://www.appropedia.org/Integrated_Voltage%2%80%94Current_Monitoring_and_Control_of_Gas_Metal_Arc_Weld_Magnetic_Ball-Jointed_Open_Source_3-D_Printer
- [14] RepRap: <http://reprap.org/>
- [15] Prusa Research: <http://shop.prusa3d.com/en/>
- [16] UltiMachine: <https://ultimachine.com/>
- [17] Cura: <https://ultimaker.com/en/products/cura-software>
- [18] OctoPrint: <http://octoprint.org/>
- [19] Slic3r: <http://slic3r.org/>
- [20] BotQueue: <https://www.botqueue.com/>
- [21] MatterControl: <http://www.mattercontrol.com/>

Author

Harris Kenny is Vice President of Marketing at Aleph Objects, Inc., makers of the LulzBot line of desktop 3D printers, parts, and plastics. Prior to joining Aleph Objects, he worked in management and ERP consulting.

Would you like to write for us?

Our editorial calendar includes upcoming themes, community columns, and topic suggestions: <https://opensource.com/calendar>

Getting Started in Open Source

We're looking for articles about how to start using an open source solution or contribute to a project, how to find the right open source community to join, advice for making money in open source, how to find (or start) local user groups or meetups, and tips for attending tech conferences.

Careers in Open Source

We want to hear about your open source-related work, tips for succeeding in a tech career and for showing open source skills on tech resumes, advice for advancing your career by growing your network, and insights into how attending conferences can get you hired.

Diversity in Open Source

How did you increase diversity in your project and community? Tell us about your open source-related grants or scholarship opportunities. Do you have tips for attracting a more diverse pool of contributors? How has increasing diversity in your community helped?

Open Science

Interesting open science topics include open hardware tools with sensors being used to take field measurements; big data processing; education and science software for Linux (for example, Celestia space simulator, Avogadro for molecular modeling, GRASS/QGIS software for geospatial sciences, R/Julia for statistics, and niche applications we've never heard of); why openness matters in the sciences; recent discoveries made using open tools and/or open data; a roundup of journals being published under open licenses.

Open Hardware

Show off your HowTos, tutorials, and demos of open hardware in the wild.

Open Source in Entertainment

We're looking for stories about how open source tools, projects, and communities keep us entertained.

Open Source Events

Share best practices for organizing a local meetup or online hack-a-thon, tips for organizing a community conference, and lesson learned from hosting or attending an open source-related event.

Open Source in Education

Which open source tools are helpful for the classroom? How are open source technologies being used or taught in your schools? We're always excited to hear how open source is helping education, so send us your stories.

Open Source Around the World

What role does open source play in your country's government? Which projects are active in your local communities? How are humanitarian open source solutions helping international communities? How has your work in open source helped you connect with people in other countries and collaborate with international contributors? How does your remote international team collaborate? Tell us about how your career in open source has allowed you to travel the globe.

7 big reasons to contribute to Opensource.com:

- 1 Career benefits:** “I probably would not have gotten my most recent job if it had not been for my articles on Opensource.com.”
- 2 Raise awareness:** “The platform and publicity that is available through Opensource.com is extremely valuable.”
- 3 Grow your network:** “I met a lot of interesting people after that, boosted my blog stats immediately, and even got some business offers!”
- 4 Contribute back to open source communities:** “Writing for Opensource.com has allowed me to give back to a community of users and developers from whom I have truly benefited for many years.”
- 5 Receive free, professional editing services:** “The team helps me, through feedback, on improving my writing skills.”
- 6 We’re loveable:** “I love the Opensource.com team. I have known some of them for years and they are good people.”
- 7 Writing for us is easy:** “I couldn’t have been more pleased with my writing experience.”

Email us to learn more or to share your feedback about writing for us: <https://opensource.com/story>

Visit our Participate page to more about joining in the Opensource.com community: <https://opensource.com/participate>

Find our editorial team, moderators, authors, and readers on Freenode IRC at #opensource.com: <https://opensource.com/irc>

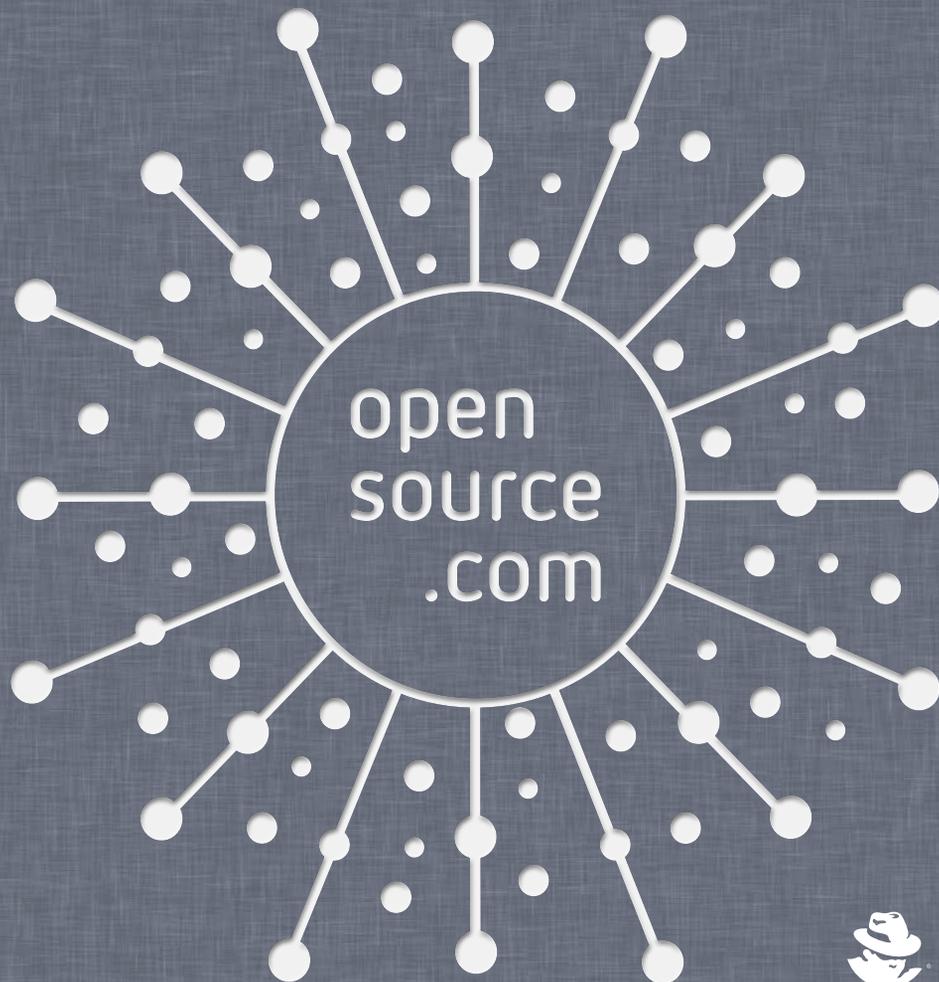
Twitter @opensourceway: <https://twitter.com/opensourceway>

Google+: <https://plus.google.com/+opensourceway>

Facebook: <https://www.facebook.com/opensourceway>

Instagram: <https://www.instagram.com/opensourceway>

IRC: #opensource.com on Freenode



SUPPORTED BY RED HAT