

Copyright © 2011  
Free Software Foundation  
51 Franklin Street, 5th Floor  
Boston, MA 02110-1301  
(617) 542-5942  
info@fsf.org  
www.fsf.org

This bulletin was published by the **GNU Press** (gnupress.org) and produced using only free software – Inkscape, GIMP, Markdown, and L<sup>A</sup>T<sub>E</sub>X.

Remember, if you have any ideas, feedback or suggestions for things you'd like to see in future editions of the *FSF Bulletin*, we'd love to hear them!

The articles in this bulletin are individually licensed under the **Creative Commons Attribution No Derivative Works 3.0 United States License**. To view a copy of this license, visit [creativecommons.org/licenses/by-nd/3.0/us/](http://creativecommons.org/licenses/by-nd/3.0/us/) or write to us at the address above.

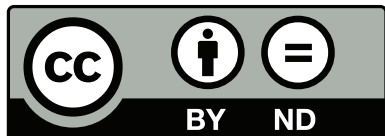


Image credits: GNU artwork by Etienne Suvasa.

## How to Contribute

### Associate Membership:

Become an associate member of the FSF. Members will receive a bootable USB card and e-mail forwarding. To sign-up or get more information, visit [member.fsf.org](http://member.fsf.org) or write to [membership@fsf.org](mailto:membership@fsf.org).

**Online:** Use your credit card or PayPal account to make a donation at [donate.fsf.org](http://donate.fsf.org) or contact [donate@fsf.org](mailto:donate@fsf.org) for more information on supporting the FSF.

**Jobs:** List your job offers on our jobs page. See [fsf.org/jobs](http://fsf.org/jobs) for details.

**United Way:** As a 501(c)(3) tax-exempt organization, the FSF is eligible to receive United Way funds. See [fsf.org/uw](http://fsf.org/uw) for more information.

**Free Software Directory:** Browse and download from thousands of different free software projects at [directory.fsf.org](http://directory.fsf.org).

**Volunteer:** To learn more, visit [fsf.org/volunteer](http://fsf.org/volunteer).

**Free Software Supporter:** Make sure you're getting all the very latest news from the FSF at [fsf.org/fss](http://fsf.org/fss).

**LibrePlanet:** Find local groups in your area or start your own at [libreplanet.org](http://libreplanet.org)!



# Bulletin

Issue 18  
June 2011

## In this issue of the Bulletin...

Welcome to the Bulletin	1
Why should I care about that?	2
LibrePlanet 2011	4
Recommending licenses for new projects	5
The danger of ebooks	6
Spotlight: Merlin Cloud	8
GNU Press news	10
Free Technology Academy	10

## Welcome to the Bulletin

by *Matt Lee*  
Editor

Welcome to the first *FSF Bulletin* of our 26th year. In this issue we have insights on some of the exciting developments in free software, as well as articles reminding us of potential threats to our freedom and autonomy.

We also have some changes at the Free Software Foundation. After ten years with the Foundation, our former executive director Peter Brown has decided to move on to new pastures — handing over the reins to a familiar face, John Sullivan. John has held a number of positions in the Foundation, most recently serving as operations manager. He has returned to Boston after his brief dalliance with life on the West Coast. Josh Gay, who some of you may remember as a campaigns manager from a few years ago, has also returned to continue that work.

Finally, throughout this issue you will see various pieces of GNU artwork. These were recently rescued from our decaying archive and scanned. Our thanks go to Etienne Suvasa for all of his work over the years providing us with these glorious pieces of art. ♥



Come on in!



example, should not come with restrictions that say you can't use it to write Republican campaign materials. Because running a program on a computer to do something one wants to do is the minimal definition of what it means to use a computer at all, the importance of this freedom is easy to explain.

Likewise, *Freedom 2*, the freedom to redistribute copies so you can help your neighbor, is straightforward. Like freedom 0, it is not hard for everyone who encounters computers to see how this freedom directly impacts their lives. When using a computer, it is *prima facie* beneficial and empowering to be able to exchange with others copies of programs that do useful things.

So far, so good. But now we are left with the two freedoms for which access to the "source code" is a precondition: *Freedom 1*, the freedom to study how the program works and to change it to make it do what you wish; and *Freedom 3*, the freedom to subsequently distribute copies of your modified versions to others.

The importance of these freedoms has always been difficult to explain to people who aren't familiar with how programs are written. We lack an equivalent of Schoolhouse Rock's "How a Bill Becomes a Law" showing how source code becomes a running program. Even if we had one, we couldn't expect everyone to be interested.

Unfortunately, things on this front have recently taken a turn for the worse. As of this writing, the top result when searching Google for "source code" is not a definition talking about the human-readable language in which programs are written. It's a Jake Gyllenhaal movie. I haven't seen this

restrictions; and more specifically to contribute to the building of a critical mass in the use of free technologies.

- The Free Technology Academy consists of an advanced virtual campus with course materials that are followed entirely online.
- The contents of the program are focused on free technologies and designed by e-learning experts.
- The Free Technology Academy is specially oriented to IT professionals, students, educators and decision makers.
- All learning materials used in the FTA are published under licenses that allow them to be freely used, modified and redistributed. Learners who enroll in the FTA receive tuition from the partner universities' teaching staff, and FTA credits are recognized by these universities.
- The FTA aims to share the burden and benefits of developing and exploiting course modules related to free technology. Organizations interested in joining forces are welcome to strengthen the Partner Network.

## Why?

The use of free technologies is considered a key factor for sustainability: Free standards guarantee interoperability and provide a level playing field for developers; free software implementations help to prevent the formation of monopolies that hinder free competition in the ICT sector.

The presence of free technologies is steadily growing in many sectors,

such as public bodies, entertainment, embedded systems, mobile computing, etc. The FTA courses cover these fields from several points of view: technical, economic, organizational, and legal.

## For whom?

FTA is for people who want to acquire knowledge about these topics and obtain a certificate that is accepted at the master level by the university partners. However, holding a university degree is not required to enroll. Typical participants are ICT professionals, educators, students, civil servants, and decision-makers from different countries.

## Learning methodology

All courses provided by the Free Technology Academy are conducted entirely online at the FTA Virtual Campus. The FTA learning methodology allows learners to define their own study schedules: asynchronous communication tools, few deadlines, and activities that can be joined at different dates and times. This model allows for anyone to follow FTA courses, regardless of their location and job, as long as they have regular access to the Internet. 🍷

## Why should I care about that?

by John Sullivan  
Executive Director

At the FSF, we have a goal of increasing the size and strength of the free software movement. To expand the movement, we need to get the attention of people who have never heard of free software before, and explain why it's important. We have to make the case for the four freedoms that characterize free software, and to succeed fully, we must be able to do this for people who don't care to know more than they have to about computers.

*Freedom 0*, the freedom to run the program for any purpose, is the easiest to explain. A word processor, for

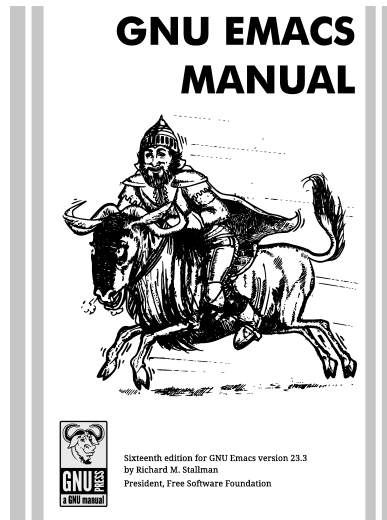
We use it internally to provision new virtual machines within our private cloud. Merlin will then provision these instances and even register them automatically in DNS. It makes the rapid creation of large numbers of provisioned virtual machines particularly quick and easy, which I would say is a general, if unwritten, expectation of cloud computing. ☹



We've also introduced a new addition to the GNU menagerie, the adorable and soft baby gnu. We still have our traditional stuffed gnu as well, but for a limited time only.

Looking ahead, we expect to release a new edition of the *GNU Emacs Manual* soon, updated for Emacs 23.3.

As always, FSF associate members can get a discount of 20% off all purchases at [shop.fsf.org](http://shop.fsf.org). ☹



The new GNU Emacs manual design

## GNU Press news

by **Donald Robertson III**  
Sales Administrator

The FSF is proud to announce some new merchandise available at [shop.fsf.org](http://shop.fsf.org). We have two new books out right now: *Free as in Freedom (2.0): Richard Stallman and the Free Software Revolution* and the second edition of *Free Software Free Society: Selected Essays of Richard M. Stallman*. Signed copies are also available through the store, but supplies are limited.

## Free Technology Academy

by **Matt Lee**  
Campaigns Manager

The Free Technology Academy provides formal training at the university level about free software and standards. The FTA Consortium aims to contribute to a society that permits all users to study, participate and build upon existing knowledge without

movie, but I'm told that in it, the "Source Code" is a government experiment revolving around a program that enables one person to cross over into another's identity in the last eight minutes of life. I've *mostly* gotten over the theory that this movie is in fact a conspiracy designed to make our jobs harder.<sup>1</sup>

The prominence of Gyllenhaal's *Source Code* is just more buckshot for naysayers who claim we will never be able to convince people on a large scale that the freedom to modify a program is critical, because most people in their lives will never want to modify a program. So why should they care?

Richard Stallman has been asked and has answered this question many times. In his answers, he highlights the importance of what amounts to literacy. He says that, with access to source code, anyone can learn to be a programmer and come to value this freedom; and it doesn't take much work to learn how to do a lot, even for people who don't ever intend on becoming master programmers.

I can testify that his answer worked for me. What little I know about programming is self-taught from free software, other than a couple high school classes in Pascal and BASIC. But I know enough — most importantly enough about how to look up answers for the things I don't know — that I can accomplish a great deal in my personal computing environment.

Even such minimal literacy is emancipatory; it is the surest way to make sure that you are not a prisoner of the software you use. But I also recognize that this appeal doesn't work

<sup>1</sup>But if you have evidence, please contact me.

for everyone, and that's okay. I believe we can also earn the support of people who can think of nothing they would like to do less than learn or change how their computer works.

In this effort, we are up against powerful marketing machines like Apple, who spends millions of dollars to tell us repeatedly that nobody should want anything more than for their technology to *just work*, in the process paradoxically defining freedom to mean the antithesis of both the desire and the ability to change anything about one's circumstances. This marketing strategy runs from their software down to their physical devices, which are known for prohibiting simple things like changing the battery or installing a new storage card.

Of course people want their technology to work. But Apple's trend-setting overemphasis on this idea has made them the Fox News of technology. Just as Fox News rides on the motto "Fair and Balanced," heavily implying that viewers don't need any other news sources, so Apple and its budding copycats ride on the idea that users don't need real freedom, because they are already having the best possible experience.

Free software certainly isn't always best when it comes to functionality or ease of use, and in many places it can learn from proprietary software. But just as plenty of people who do not write news articles themselves still strongly support freedom of the press so that others can write articles for them to read, so plenty of people can come to recognize the ways in which proprietary software company efforts to crush the creativity and intelligence of those who *do* want to modify source code — of the non-Gyllenhaal variety

— end up hurting everyone who ever needs to run a program to do anything. Like the news media in places with no free press, proprietary software means gate-kept control over information, tools, and knowledge, and that’s scary.

“Fair and Balanced” exists in the context of the First Amendment. Likewise, companies like Apple can exist and offer their users streamlined choice-free experiences serving media certified by Steve Jobs, but they must do so in a free world, eschewing the subsidized backing of government coercion they currently enjoy in the form of copyrights, patents, and Digital Restrictions Management. They can have their image, but they must not rely on fundamental user freedom being illegal in order to do so.

Many people can and do value freedoms that they themselves do not exercise, because they empathize with the importance of those freedoms to others, and recognize that the alternative is for their own, more dear freedoms to someday be threatened (“First they came for the...”). Many people also care about the way in which the things they use are produced, and they want those methods to be ethical. When inviting new voices to the free software movement, we can and should appeal to these empathies, but we don’t need to depend on them. The character of the software available to any of us is dependent on the ethical environment in which that software is made. Whether programmers or not, we all need the programs we use to be made and run in freedom. 🍷



## LibrePlanet 2011

by *Matt Lee*  
*Campaigns Manager*

In March we had our third LibrePlanet conference here in Boston. This year we wanted to run a smaller event, with the expectation that next year’s conference will be a larger affair — our biggest conference to date.

Because we planned a smaller event, we chose a smaller venue than in previous years. With only a single track of talks, it was clear we could find a suitable lecture hall at many of the universities and colleges in the Boston area. For us, Bunker Hill Community College proved ideal for our needs. Minutes from the heart of downtown Boston and so close to the subway system that it has its own station, BHCC features prominently in the movie *Good Will Hunting* in the office of Robin Williams’s character.

can mean Amazon AWS, which has many significant ethical concerns, but it can also mean something like Eucalyptus/UEC (the GPLv3-licensed version), or other free software which provides similar services.

Merlin allows you to quickly request the creation of new virtual machines and storage volumes, and can also assist in provisioning them (getting them ready to do real work). It handles a lot of details in the background (such as Puppet certificate signing, BIND dynamic DNS updates, etc.) and aims to make day-to-day cloud operations as easy as a couple of clicks. It also aims to provide a free software option versus the proprietary software or closed web services that are used to interface and interact with clouds today.

**What technologies are used to make Merlin work?**

As I said, Merlin is a Ruby on Rails project. I actually used the project as an excuse to learn Ruby/Rails, as I had only used Django or Zope in the past. Underneath the hood, it uses ActiveMQ to pass messages between the front end and the back end, and of course a whole basket of Ruby gems. Merlin also assumes that you are using Puppet to provision your systems.

**What can developers who are interested in Merlin do to help it succeed?**

Merlin, primarily, needs more attention. It works for me, in my environment; it would be helpful if others would try using it. It really needs better packaging and documentation: it isn’t as simple as installing it and clicking a few things. It also needs quite a bit of work on its UI, and I am hoping to get it to a beta release where a lot of the basic functionality will be a bit

more implemented than it is now. In the future, Merlin will support spawning entire groups of machines, or entire self-contained systems, within the cloud. I would also like Merlin to be API-agnostic, and support a wide variety of cloud software. All that being said, the software is available today, so anyone who wants to hack on it can do so.

**We’ve seen some rather gloomy-looking options from big companies who are talking up the cloud as the solution to all our problems. What can Merlin do to give people some flexibility without requiring them to give up their control?**

Primarily, the real issue with the pervasiveness of clouds with ethical issues for users is simply that they are there, and they are the only real option unless you are looking very carefully. The average user isn’t yet aware of the traps and pitfalls involved in using cloud services, and is very much attracted to the ease of use and simplicity that the cloud promises — especially the idea of their files or data being available “everywhere.” Unfortunately, this simplicity comes with a price — it is inherently linked to an expropriation of freedom and privacy which becomes harder and harder to reverse the more one relies on the corporate clouds.

To my knowledge, no one has attempted to build a large, scalable free software cloud that inherently respects the freedom and privacy of its users. I sincerely hope Merlin can be helpful in this regard, even if it is only to encourage communities of users to build their own small private clouds instead of always using the big companies.

**How do you use Merlin?**

- To design players so users can send authors anonymous voluntary payments.

**Ebooks need not attack our freedom, but they will if companies get to decide. It's up to us to stop them. The fight has already started.** ♡



See [gnu.org/graphics/license-logos.html](http://gnu.org/graphics/license-logos.html) for more logos.

## Spotlight: Merlin Cloud

by *Matt Lee*  
Campaigns Manager

Justin Baugh is a former systems administrator at the Free Software Foundation. Now he works deploying large-scale clusters for major corporations, and in his limited spare time works on a new free software project to help people deploy their own virtualization.

**First, let's dispel this myth about cloud computing. What is it, and why is it compatible with free software ethics?**

"Cloud computing," unfortunately, is one of the most misunderstood terms in the recent history of computing. Literally, it has a million different definitions to a million different people. I like to think of cloud computing simply as an abstraction: a system that provides methods for requesting and utilizing computing resources without having to know anything about the underlying systems providing them.

Primarily, cloud computing is about (a) aggregating server, network, and storage resources into a seemingly contiguous system ("the cloud"), (b) providing some kind of interface for the user to request or release these resources, and (c) making these resources network or location agnostic, so that the resources are accessible from anywhere, even in the face of system or network failures.

I agree that there are many ethical considerations with "the cloud," especially in its current invocation by larger corporations. However, their cloud need not be ours; it is becoming more and more possible to use free software to build clouds that respect free software ethics and don't require a user to compromise on their freedoms or privacy.

For instance, one could imagine a GNU cloud which only ran free software, and had strong privacy and data protection safeguards, but provided users the same type of experience they might expect from Amazon's EC2. This kind of vision is entirely possible.

**Okay, so what's Merlin? Give me the 30-second overview...**

Primarily, Merlin is a Ruby on Rails application that I created as a way of interacting and controlling EC2 API-compatible clouds. This

The event kicked off with a welcome from our newly appointed executive director John Sullivan, who listed some of the things he'd like to achieve now that he's running the show: better access to the people running the organization and an increased focus on our campaigns work. This was followed by a talk from Brett Smith about the work of the GPL Compliance Lab. The Compliance Lab is responsible for resolving license violations that involve GNU software. Brett acts as the liaison between the FSF and its attorneys at the Software Freedom Law Center (SFLC) in New York City.

With the first two FSF talks out of the way, it was time for an outside perspective. This came from conference regular Máirín Duffy, who works on the Fedora distribution of GNU/Linux. Her talk *Empowering Girl Scouts using free software* described a collaborative effort to teach groups of Girl Scouts and other young people in Massachusetts how to use free creative software like GIMP and Inkscape. The children made a variety of creative works, including t-shirts and posters, which were later printed and put on display.

After lunch, Richard Stallman gave his keynote speech. Unlike previous years, he did not announce a new philosophical essay, but instead offered a warning about the dangers of cell phone tracking and proprietary software on mobile devices. There was positive news to report in this space: the Replicant project has successfully built a fully free, functional version of the Android/Linux operating system for the HTC Dream phone.

Lightning talks followed. Bob Call, Jason Self, Asheesh Laroia, Mary-Anne Wolf, and Dave Crossland of-

ferred short but sweet insights into their work on router hacking, Python advocacy, real-world accessibility for disabled people, and free fonts.

The conference concluded with our annual Free Software Awards ceremony. GNU Gnash maintainer and GNU veteran Rob Savoye received the Award for the Advancement of Free Software, while Andrew Lewman from the Tor Project stepped up to take home the Award for Projects of Social Benefit. ♡



## Recommending licenses for new projects

by *Brett Smith*  
License Compliance Engineer

We recently published a new page on our site, entitled "How to choose a license for your own work."<sup>2</sup> It's a comprehensive set of license recommendations for new projects. It explains what factors are important to

<sup>2</sup>[gnu.org/licenses/license-recommendations.html](http://gnu.org/licenses/license-recommendations.html)

consider when making licensing decisions, and suggests specific licenses for different scenarios.

Publishing resources like this helps the Compliance Lab achieve a couple of different education goals. Most obviously, it serves as a guide to help new projects make good licensing decisions. It also illustrates how the FSF uses licensing as a tool to encourage the adoption and development of free software. Sometimes people have the mistaken impression that we care more about a program's license than freedom. They claim on Internet forums that the FSF wants everything to be GPLed. This page clears up that misconception and puts the GPL in its proper context as a means to an end.

During the drafting process for this guide, we decided to make one adjustment to our existing recommendations: in situations where a copyleft license is not appropriate, we now recommend the Apache License 2.0. This shift was spurred on by several changes that have taken place over the past few years. As sad as I am to admit it, patents remain a growing threat to free software. Since the US Supreme Court's decision in *Bilski v. Kappos*, software patent litigation has continued as before, and cases that have followed like *Bedrock v. Google* are clear attacks against free software. We will never be completely safe until software patents are abolished, but any defenses we can implement today are worthwhile.

The Apache License 2.0 is the best non-copyleft license that does what a copyright license can to mitigate threats from software patents, and the Apache Software Foundation deserves credit for their efforts in this space. The Apache License is a well-

established, mature license that users, developers, and distributors alike are all comfortable with. It also enjoys support from other free software licenses: GPLv3 is compatible with it, and the forthcoming Mozilla Public License 2.0 should be as well.

It's unfortunate that the Apache License 2.0 isn't compatible with older free software licenses like GPLv2. As we considered this change to our recommendations, this point was easily the most important one weighing against it. Fortunately, every major copyleft license has or will soon have Apache compatibility in their latest versions, which mitigates those concerns. Ultimately, we went ahead with this change because we want our recommendations to help projects make decisions that will serve them well for a long time to come, rather than focusing too much on today's immediate circumstances.

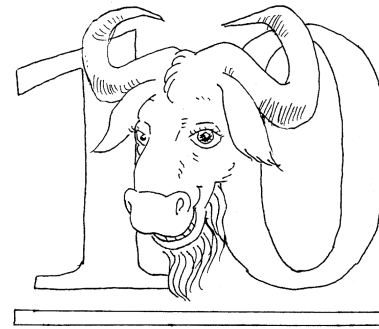
I hope our new guide and this article help people understand how good licensing decisions can go a long way to help protect and promote free software. If you have questions or feedback about any of this, please don't hesitate to write us at [licensing@fsf.org](mailto:licensing@fsf.org); we're always happy to hear from our supporters. 🍷

## The danger of ebooks

by *Richard Stallman*  
*President*

In an age where business dominates our governments and writes our laws, every technological advance offers business an opportunity to impose new restrictions on the public.

Technologies that could have em-



powered us are used to chain us instead.

With printed books:

- You can buy one with cash, anonymously, and then you own it.
- You are not required to sign a license that restricts your use of it.
- The format is known, and no proprietary technology is needed to read the book.
- You can give, lend or sell the book to another.
- You can, physically, scan and copy the book, and it's sometimes lawful under copyright.
- Nobody has the power to destroy your book.

Contrast that with Amazon ebooks:

- Amazon requires users to identify themselves to get an ebook.
- In some countries, Amazon says the user does not own the ebook.

- Amazon also requires the user to accept a restrictive license on use of the ebook.
- The format is secret, and only proprietary user-restricting software can read it at all.
- An ersatz “lending” is allowed for some books, for a limited time, but only by specifying by name another user of the same system.
- No giving or selling is permitted.
- To copy the ebook is impossible due to Digital Restrictions Management in the reader and prohibited by the license, which is more restrictive than copyright law.
- Amazon can remotely delete the ebook using a back door — It used this back door in 2009 to delete thousands of copies of George Orwell's *1984*.

Even one of these infringements makes ebooks a step backward from printed books. We must reject ebooks until they respect our freedom.

The ebook companies say denying our traditional freedoms is necessary to continue to pay authors — the current copyright system does a lousy job of that; it is much better suited to supporting those companies. We can support authors better in other ways that don't require curtailing our freedom, and even legalize sharing.

Two methods I've suggested are:

- To distribute tax funds to authors based on the cube root of each author's popularity.<sup>3</sup>

<sup>3</sup>[stallman.org/articles/internet-sharing-license.en.html](http://stallman.org/articles/internet-sharing-license.en.html)