ECE444: Software Engineering

Open Source

Shurui Zhou

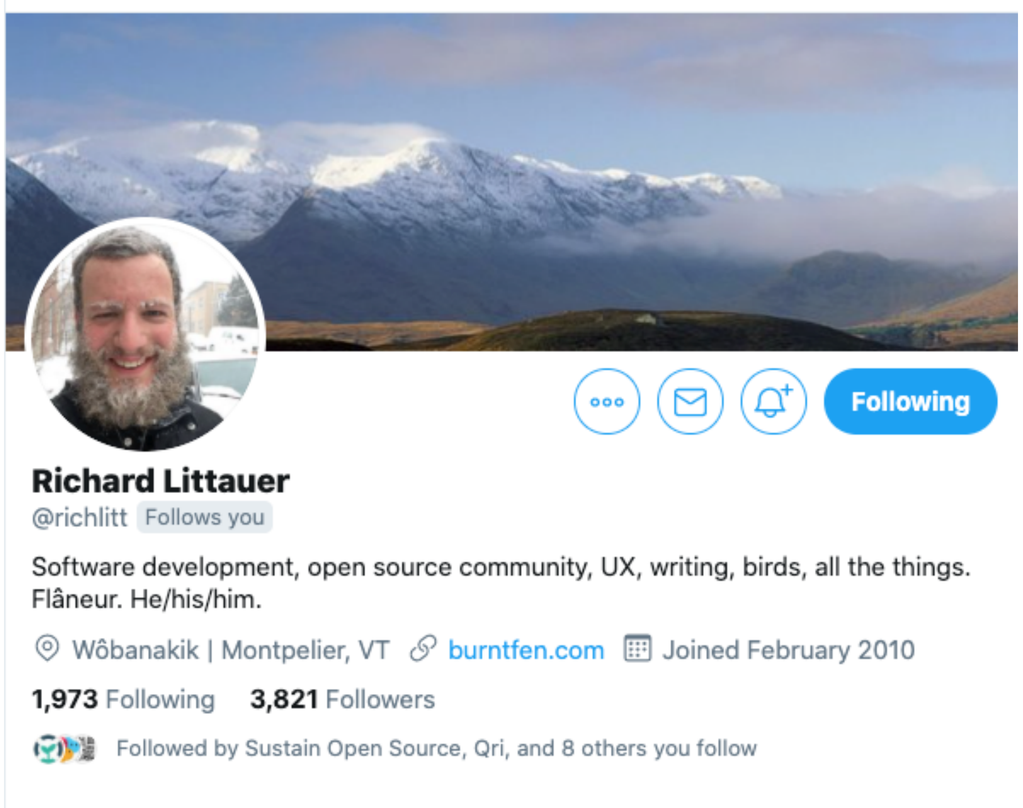
The Edward S. Rogers Sr. Department
of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

Guest Lecture on Friday

- We are using zoom
- Please turn-on your video to make the speaker feel welcome if possible

  (Virtual background competition!)
- Prepare some questions

https://twitter.com/richlitt



Richard Littauer

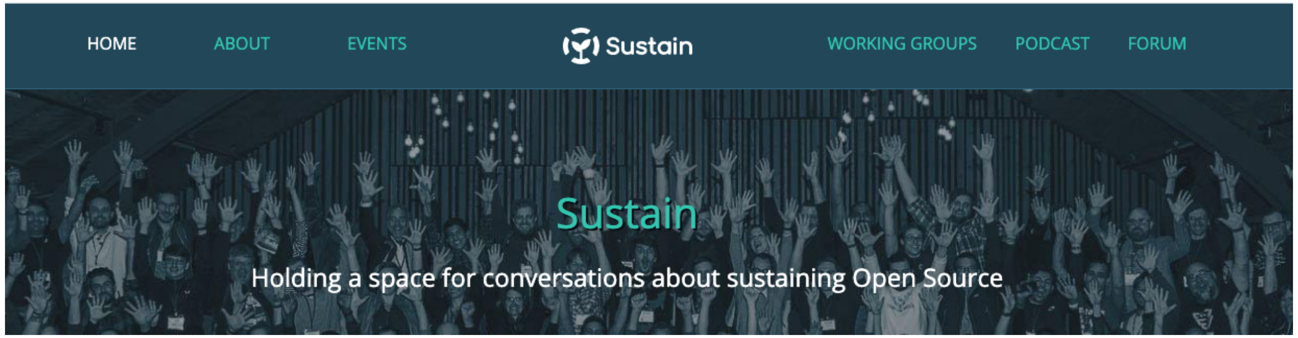
@richlitt Follows you

Software development, open source community, UX, writing, birds, all the things. Flâneur. He/his/him.

Wôbanakik | Montpelier, VT burntfen.com Joined February 2010

1,973 Following **3,821** Followers

Followed by Sustain Open Source, Qri, and 8 others you follow



Sustain

Holding a space for conversations about sustaining Open Source

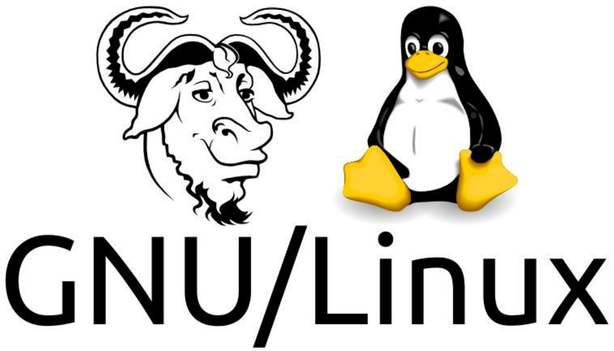
Learning Goals

- Understand the terminology “free software” and explain open source culture and principles.
- Express an educated opinion on the philosophical/political debate between open source and proprietary principles.
- Reason about the tradeoffs of the open source model on issues like quality and risk, both in general and in a proprietary context.

The culture

- “I’m doing a (free) operating system (just a hobby, won’t be big and professional like gnu) for 386(486) AT clones.”

-- Linus Torvalds

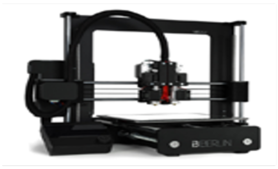


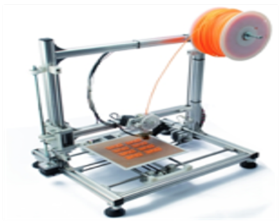
open source

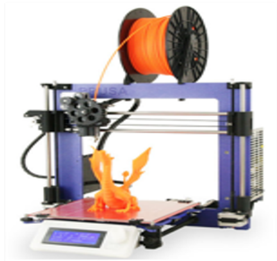
- Software
- Hardware
- What else?

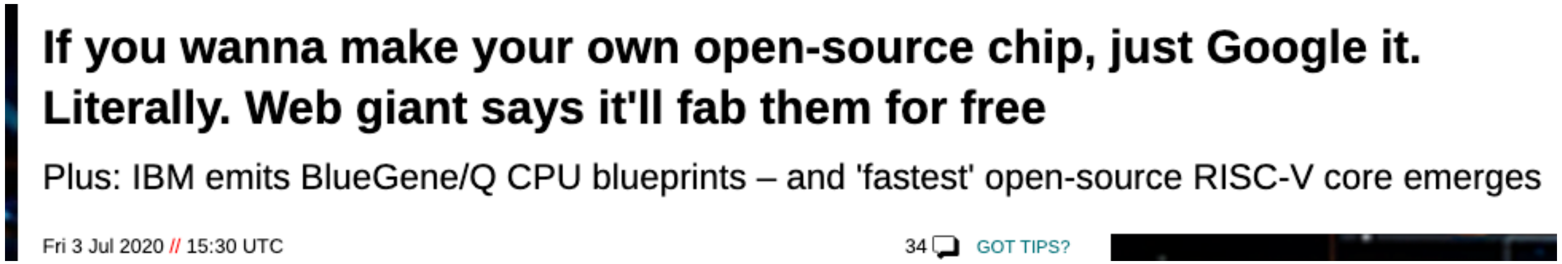
Raspberry Pi 4

Your tiny, dual-display, desktop computer









If you wanna make your own open-source chip, just Google it. Literally. Web giant says it'll fab them for free

Plus: IBM emits BlueGene/Q CPU blueprints – and 'fastest' open-source RISC-V core emerges

Fri 3 Jul 2020 // 15:30 UTC

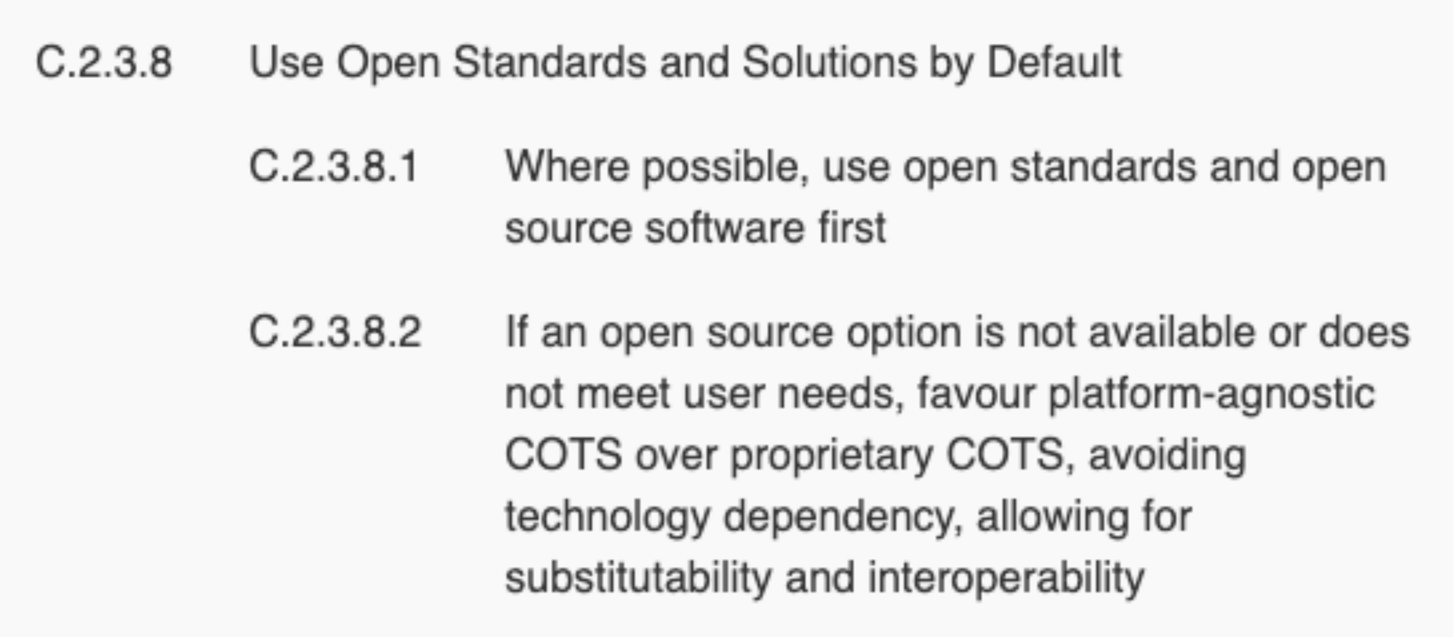
34 GOT TIPS?

OpenOffice.org
fedora
blender
Linux
mozilla
Firefox
php
MySQL
debian
Java
WordPress
Apache
ubuntu

Motivation to understand open source.

- Companies work on open source projects.
- Companies use open source projects.
- Companies are based around open source projects.
- Principles percolate throughout industry.
- Political/philosophical debate and being informed is healthy.

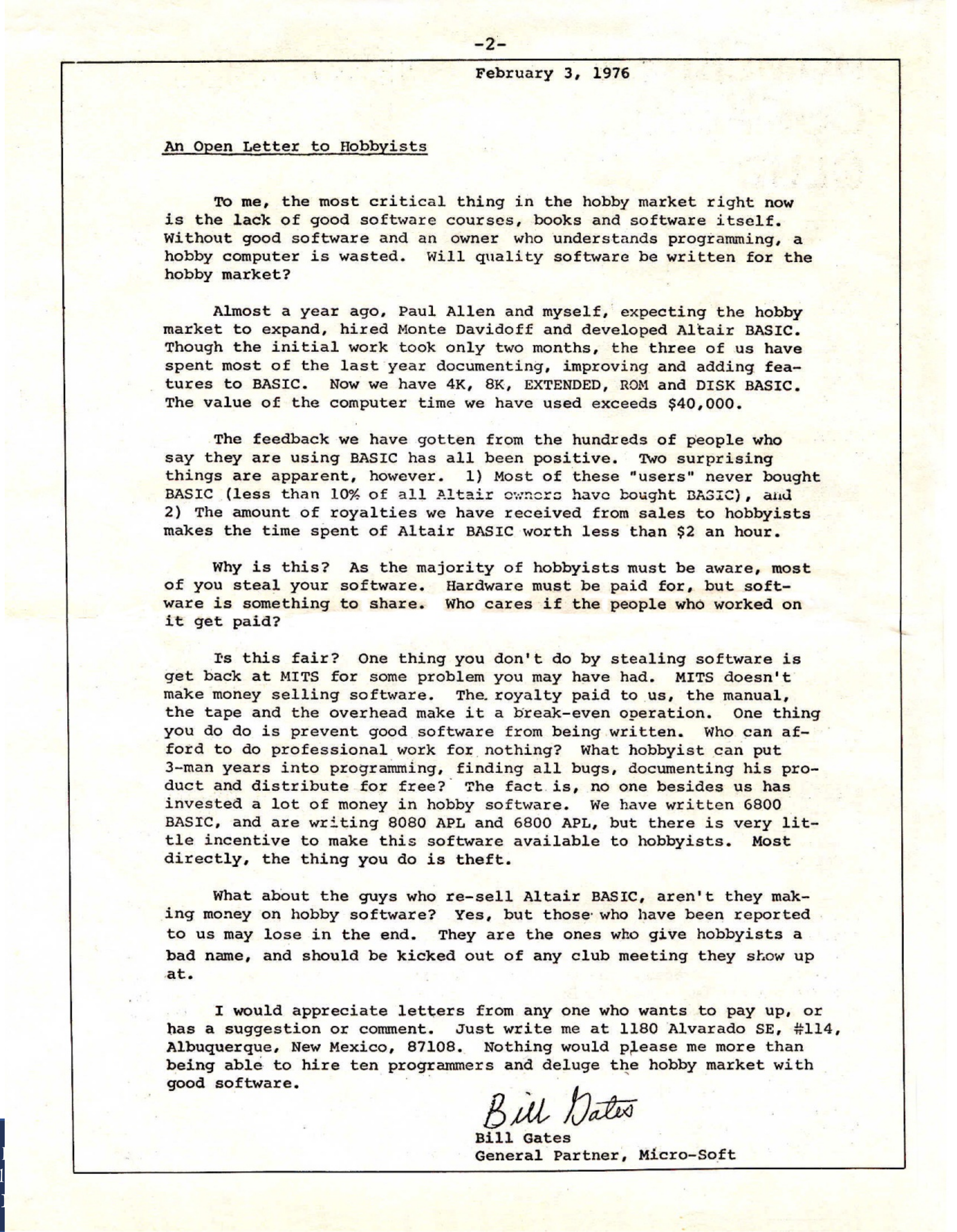
Canada Federal Government publishes a new IT directive that mandates the use of open source software first before considering proprietary software



C.2.3.8 Use Open Standards and Solutions by Default

C.2.3.8.1 Where possible, use open standards and open source software first

C.2.3.8.2 If an open source option is not available or does not meet user needs, favour platform-agnostic COTS over proprietary COTS, avoiding technology dependency, allowing for substitutability and interoperability



-2-

February 3, 1976

An Open Letter to Hobbyists

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

Almost a year ago, Paul Allen and myself, expecting the hobby market to expand, hired Monte Davidoff and developed Altair BASIC. Though the initial work took only two months, the three of us have spent most of the last year documenting, improving and adding features to BASIC. Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC. The value of the computer time we have used exceeds $40,000.

The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however. 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent of Altair BASIC worth less than $2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

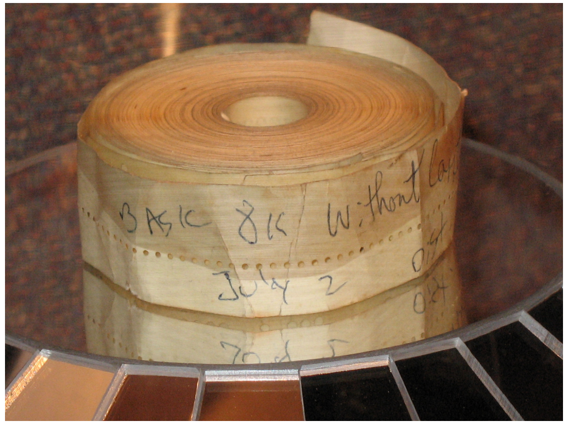
Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft.

What about the guys who re-sell Altair BASIC, aren't they making money on hobby software? Yes, but those who have been reported to us may lose in the end. They are the ones who give hobbyists a bad name, and should be kicked out of any club meeting they show up at.

I would appreciate letters from any one who wants to pay up, or has a suggestion or comment. Just write me at 1180 Alvarado SE, #114, Albuquerque, New Mexico, 87108. Nothing would please me more than being able to hire ten programmers and deluge the hobby market with good software.

Bill Gates

Bill Gates
General Partner, Micro-Soft



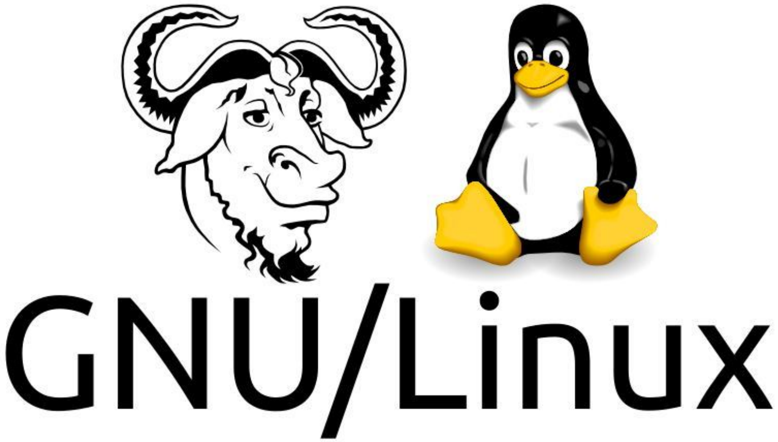
Altair BASIC，　1970

https://pt.wikipedia.org/wiki/Altair_BASIC

"Open source is an intellectual-property destroyer, I can't imagine something that could be worse than this for the software business and the intellectual-property business.”

-- former Windows chief Jim Allchin in 2001.

“Free as in free speech.”

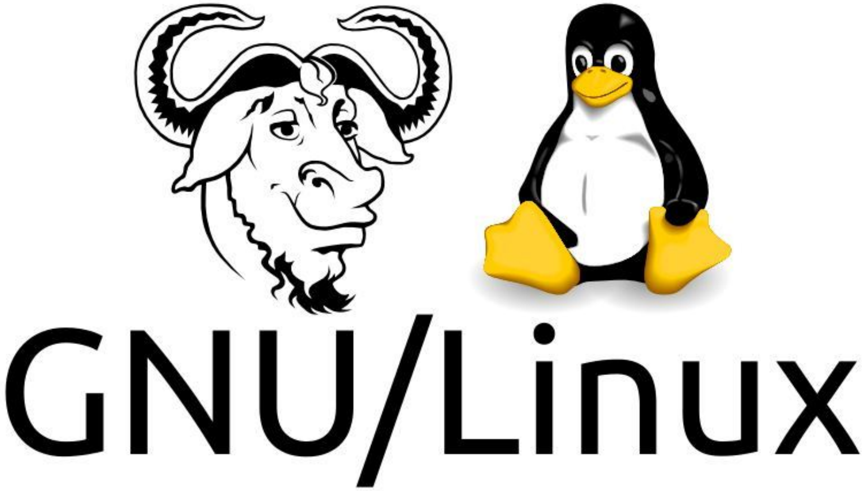


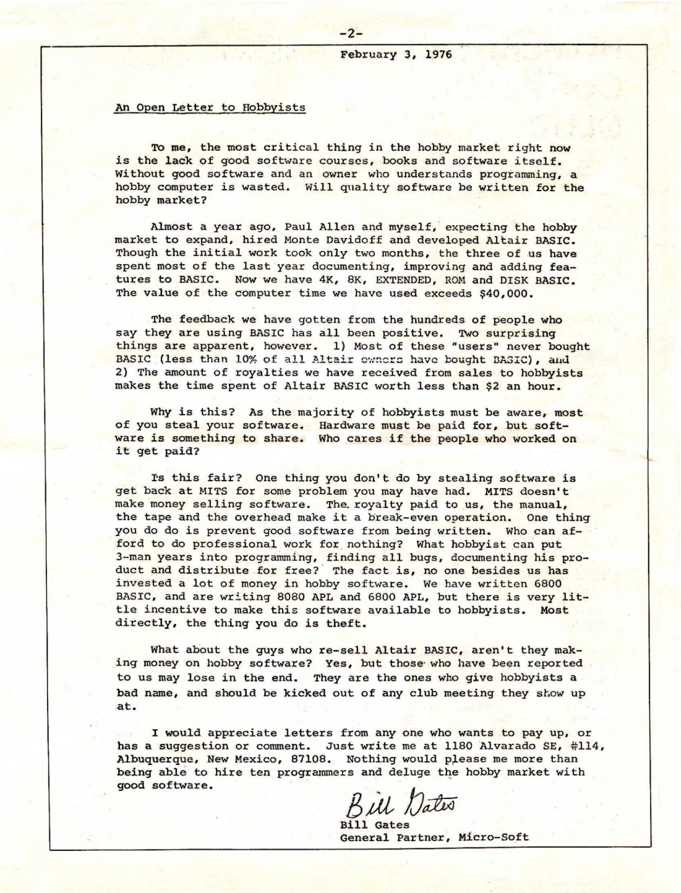
Richard Stallman

Stallman in 2019

Stallman launched the GNU Project, founded the Free Software Foundation, developed the GNU Compiler Collection and GNU Emacs, and wrote the GNU General Public License.

Stallman vs. Gates





-2-

February 3, 1976

<u>An Open Letter to Hobbyists</u>

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

Almost a year ago, Paul Allen and myself, expecting the hobby market to expand, hired Monte Davidoff and developed Altair BASIC. Though the initial work took only two months, the three of us have spent most of the last year documenting, improving and adding features to BASIC. Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC. The value of the computer time we have used exceeds $40,000.

The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however. 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent of Altair BASIC worth less than $2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

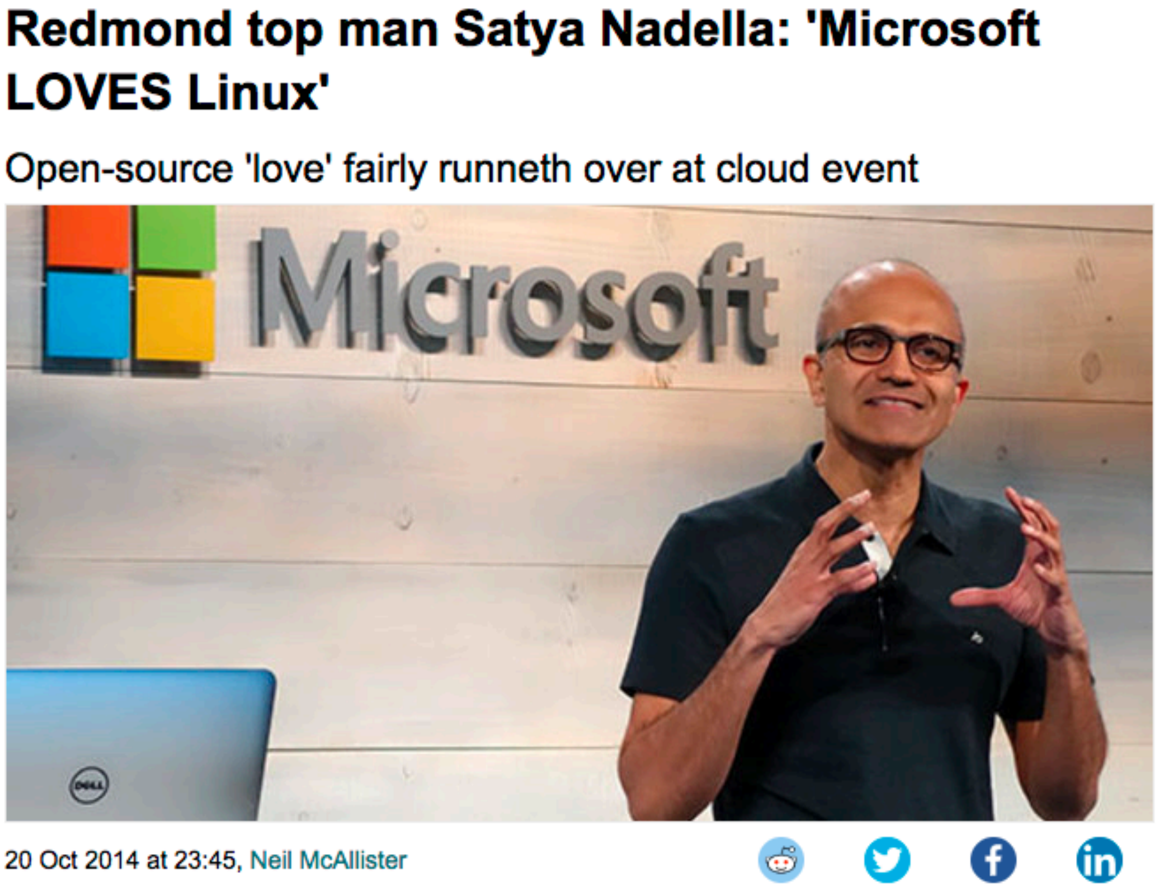
Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft.

What about the guys who re-sell Altair BASIC, aren't they making money on hobby software? Yes, but those who have been reported to us may lose in the end. They are the ones who give hobbyists a bad name, and should be kicked out of any club meeting they show up at.

I would appreciate letters from any one who wants to pay up, or has a suggestion or comment. Just write me at 1180 Alvarado SE, #114, Albuquerque, New Mexico, 87108. Nothing would please me more than being able to hire ten programmers and deluge the hobby market with good software.

Bill Gates
Bill Gates
General Partner, Micro-Soft

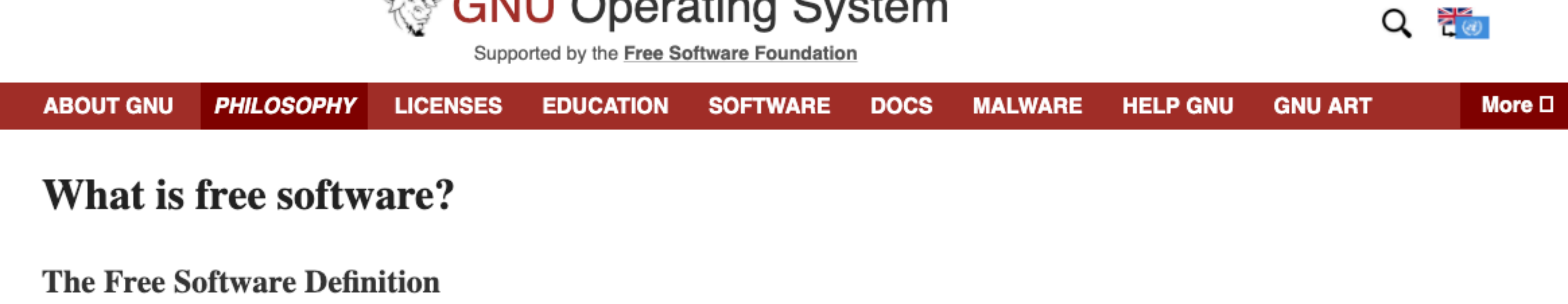
Hilarious irony



Redmond top man Satya Nadella: 'Microsoft LOVES Linux'

Open-source 'love' fairly runneth over at cloud event

20 Oct 2014 at 23:45, Neil McAllister



What is free software?

The Free Software Definition

freedom 0 : The freedom to run the program as you wish, for any purpose

freedom 1: The freedom to study how the program works, and change it so it does your computing as you wish

freedom 2: The freedom to redistribute copies so you can help others

freedom 3: The freedom to distribute copies of your modified versions to others

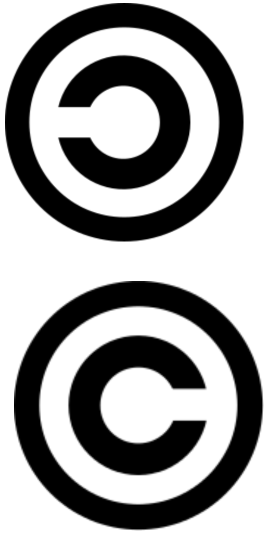
Licenses

- The MIT (Massachusetts Institute of Technology) License: This is a permissive license that places limited restrictions on software reuse.

- The GNU General Public License v2: This copyleft license gives users the freedom to run, study, and make improvements to the software.

- The Apache License v2: This is a permissive license that mandates preservation of the copyright notice and disclaimer.

- The BSD Licenses: They are a set of non-copyleft licenses that gives minimal restrictions on the use and redistribution of the software.



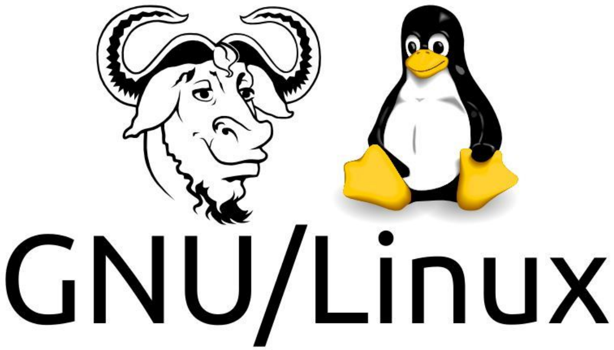
Open Source Licenses

Software	Percentage
MIT License	24%
GNU General Public License (GPL) 2.0	23%
Apache License 2.0	16%
GNU General Public License (GPL) 3.0	9%
BSD License 2.0 (3-clause, New or Revised) License	6%
GNU Lessor General Public License (LGPL) 2.1	5%
Artistic License (Perl)	4%
GNU Lesser General Public License (LGPL) 3.0	2%
Microsoft Public License	2%
Eclipse Public License	2%

List from: https://www.blackducksoftware.com/resources/data/top-20-open-source-licenses

Free Software vs Open Source

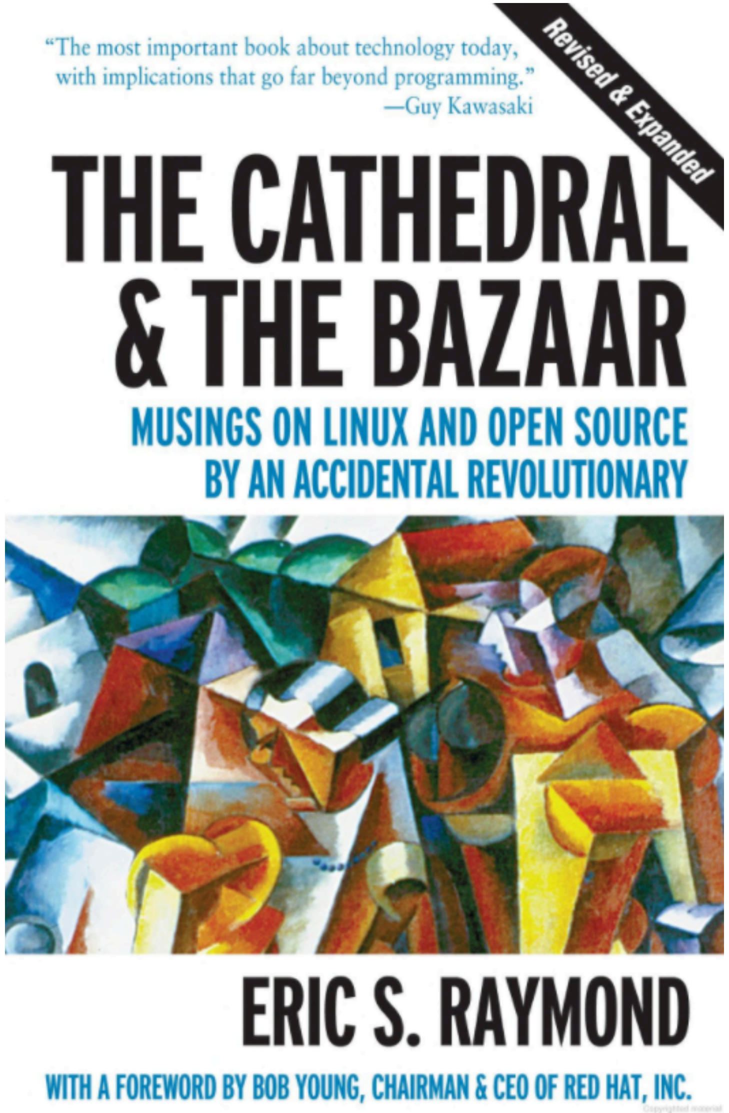
- Free software origins (70-80s ~Stallman)
    - Political goal
    - Software part of free speech
        - free exchange, free modification
        - proprietary software is unethical
        - security, trust
    - GNU project, Linux, GPL license
- Open source (1998 ~ O'Reilly)
    - Rebranding without political legacy
    - Emphasis on internet and large dev./user involvement
    - Openness toward proprietary software/coexist
    - (Think: Netscape becoming Mozilla)



The Cathedral and the Bazaar

- Centralized vs. decentralized
- Planned vs. unplanned

- Fetchmail


"The most important book about technology today, with implications that go far beyond programming." —Guy Kawasaki
Revised & Expanded
THE CATHEDRAL & THE BAZAAR
MUSINGS ON LINUX AND OPEN SOURCE BY AN ACCIDENTAL REVOLUTIONARY
ERIC S. RAYMOND
WITH A FOREWORD BY BOB YOUNG, CHAIRMAN & CEO OF RED HAT, INC.

Roles

- Leader

  Develops initial system

  Does what nobody else does

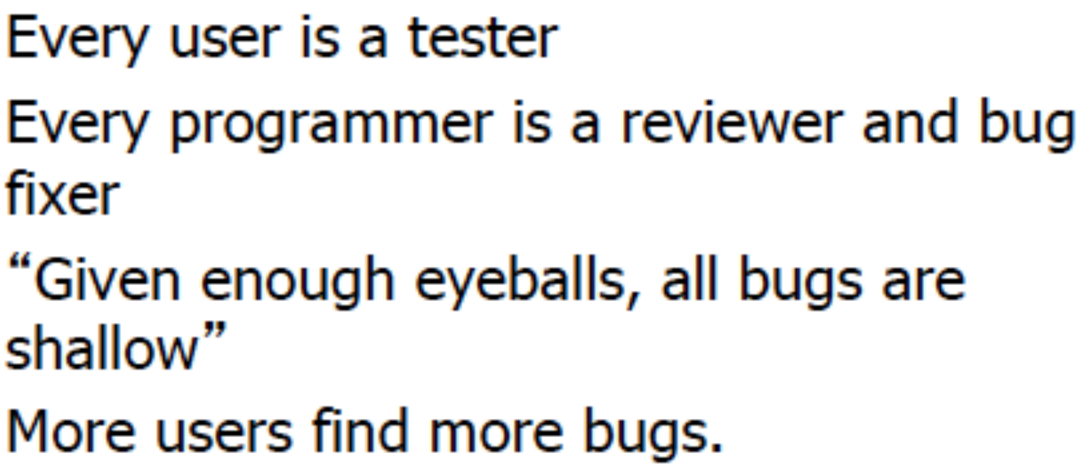
  Makes final decisions

- User/programmer
  - Dose most of the work

Requirements

- Who decides what features get added?
  - Programmers
    - who want to use the feature (scratch an itch)
    - who are persuaded to add it
- Must be a way to distribute changes for a feature (Version Control)
- Must be way to talk about desired features (mailing lists, forums)

Testing



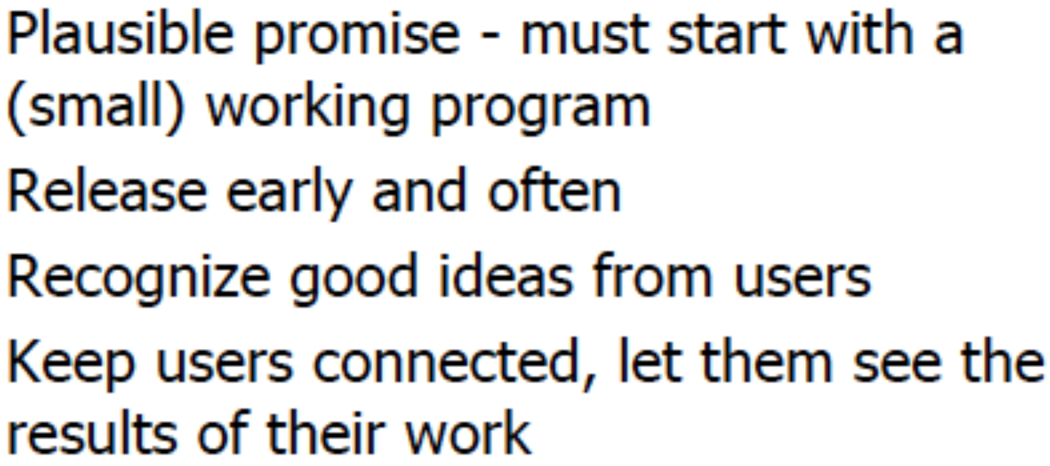
Every user is a tester

Every programmer is a reviewer and bug fixer

“Given enough eyeballs, all bugs are shallow”

More users find more bugs.

Life-cycle



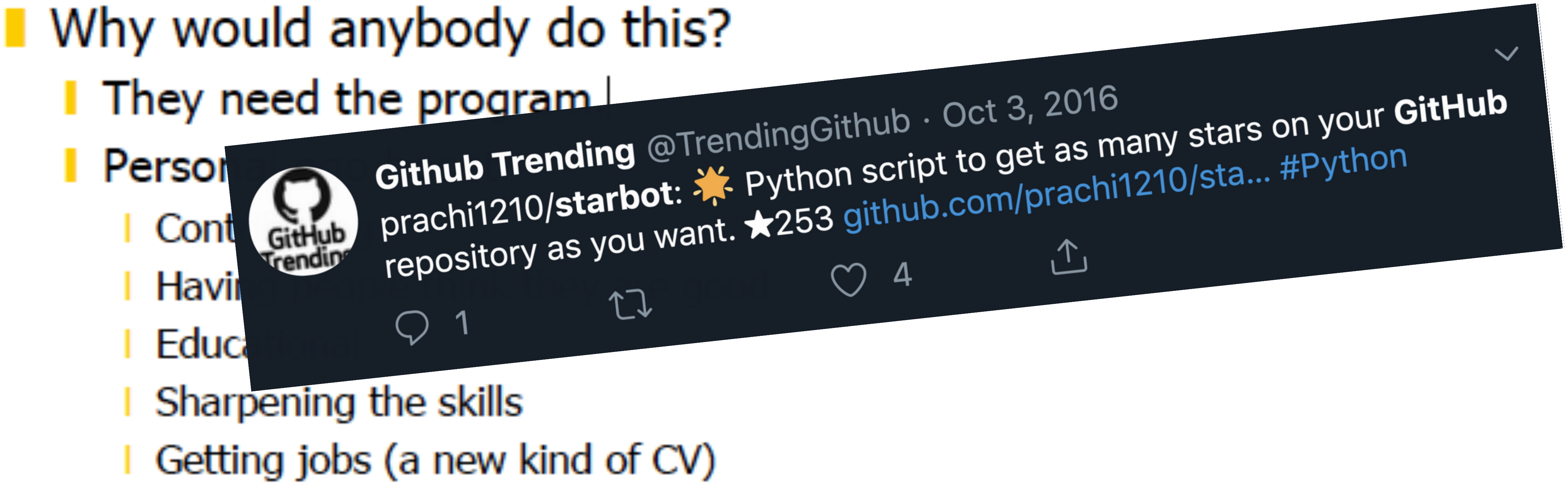
Plausible promise - must start with a (small) working program

Release early and often

Recognize good ideas from users

Keep users connected, let them see the results of their work

Rewards



- Why would anybody do this?
  - They need the program
  - Perso
    - Cont
    - Havi
    - Educ
    - Sharpening the skills
    - Getting jobs (a new kind of CV)

Github Trending @TrendingGithub · Oct 3, 2016

prachi1210/**starbot**: 🌟 Python script to get as many stars on your **GitHub** repository as you want. ★253 github.com/prachi1210/sta... #Python

Costs

- Need a leader
  - a lot of work over a long time
  - must communicate
  - an organizer as much as a designer

How do open source companies make money?

How do open source programs make money?

- Red Hat – revenues of about $2 Billion last year and is worth approximately $15 Billion.
- Mozilla – has revenues of $300 Million annually
- Apache Software Foundation – recent revenue of $1 Million

https://docs.google.com/spreadsheets/d/17nKMpi_Dh5slCqzLS
FBoWMxNvWiwt2R-t4e_l7LPLhU/edit#gid=0

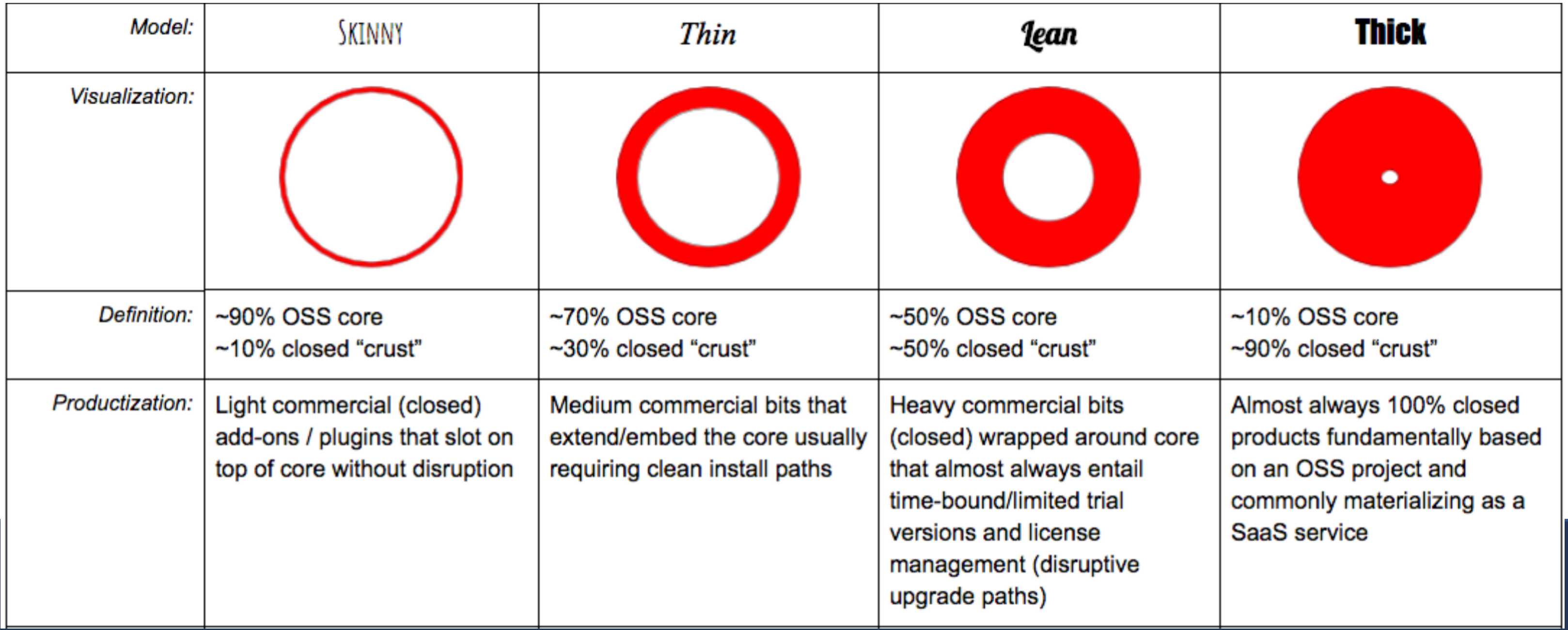
Open Source Business Models

- Support
- Hosting
- Open-core
- Restrictive Licensing
- Hybrid Licensing

Red Hat

databricks

Acquia Cloud



Model:	SKINNY	Thin	Lean	Thick
Visualization:				
Definition:	~90% OSS core ~10% closed "crust"	~70% OSS core ~30% closed "crust"	~50% OSS core ~50% closed "crust"	~10% OSS core ~90% closed "crust"
Productization:	Light commercial (closed) add-ons / plugins that slot on top of core without disruption	Medium commercial bits that extend/embed the core usually requiring clean install paths	Heavy commercial bits (closed) wrapped around core that almost always entail time-bound/limited trial versions and license management (disruptive upgrade paths)	Almost always 100% closed products fundamentally based on an OSS project and commonly materializing as a SaaS service

Dual License Business Model

- Released as GPL which requires a company using the open source product to open source it's application
- Or companies can pay $2,000 to $10,000 annually to receive a copy of MySQL with a more business friendly license

Risk: Incompatible Licenses

- Sun open sourced OpenOffice, but when Sun was acquired by Oracle, Oracle temporarily stopped the project.
- Many of the community contributors forked OpenOffice and rename it as LibreOffice
- Oracle eventually released OpenOffice to Apache
- LibreOffice changed the project license so LibreOffice can copy changes from OpenOffice but OpenOffice cannot do the same due to license conflicts

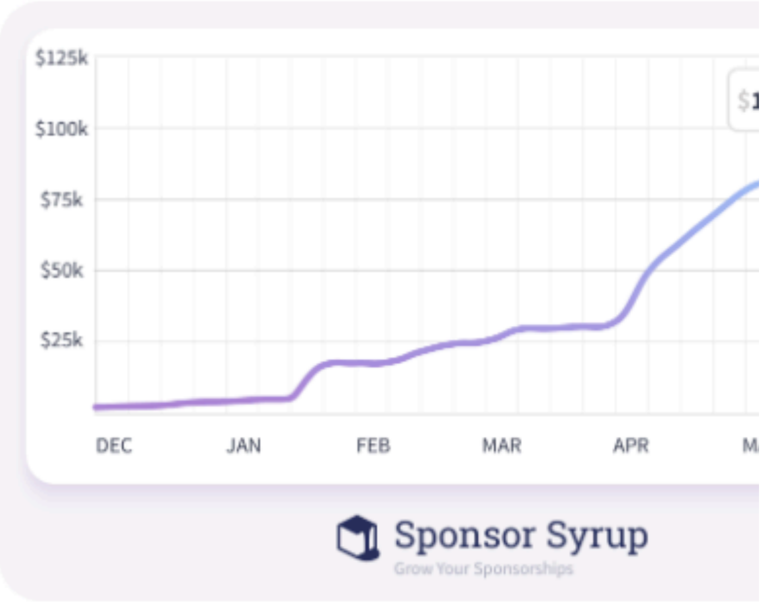
Other Open Source Business Models

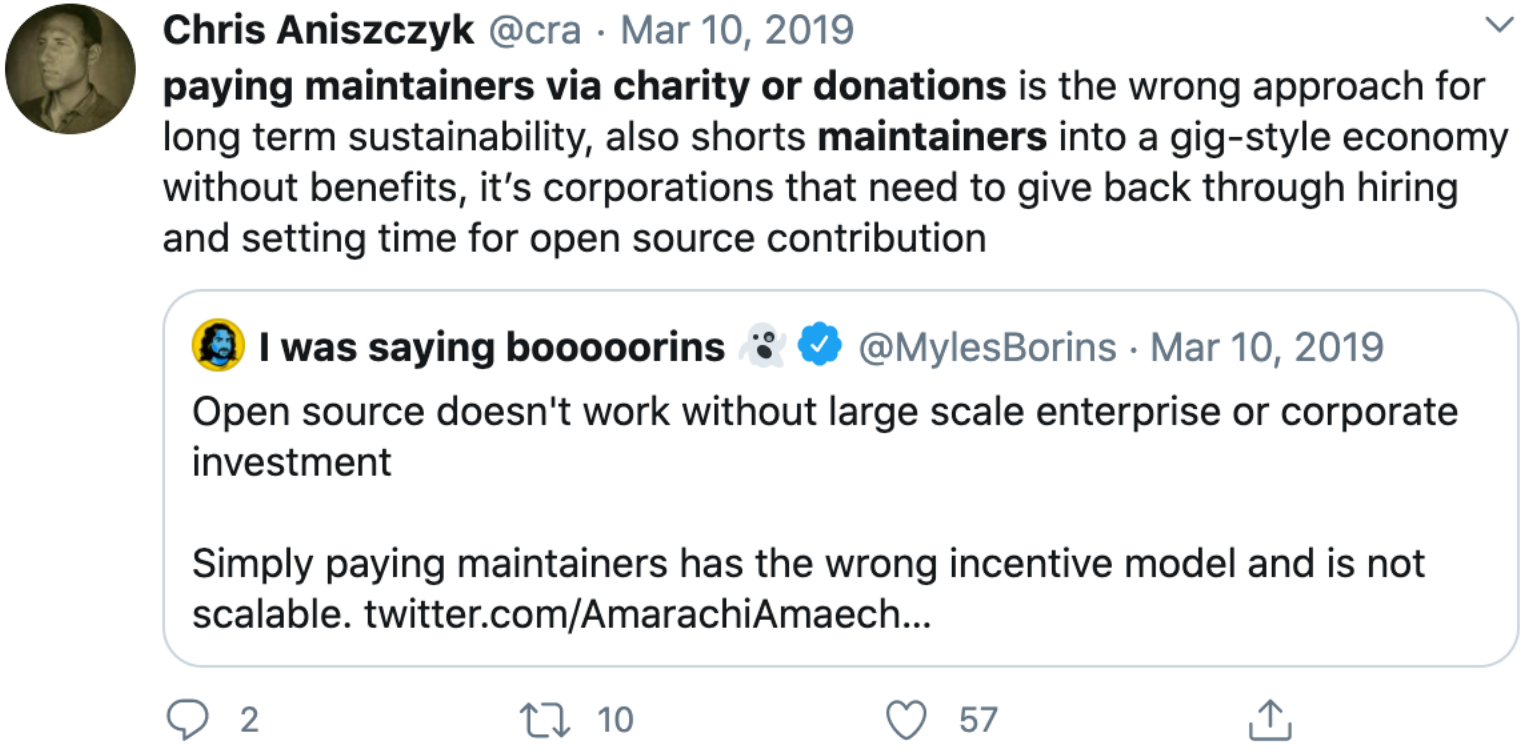
- Companies dedicate resources to projects which help them and the community
  - Apache receives donations
- Selling merchandise – Canonical (Ubuntu)
- Selling advertising or customer traffic – Mozilla

Sustaining Open Source is critically important

I Just Hit $100k/yr On GitHub Sponsors! 🎉❤️ (How I Did It)

Jun 2020





Chris Aniszczyk @cra · Mar 10, 2019

paying maintainers via charity or donations is the wrong approach for long term sustainability, also shorts **maintainers** into a gig-style economy without benefits, it's corporations that need to give back through hiring and setting time for open source contribution

> **I was saying boooorins** 👻 @MylesBorins · Mar 10, 2019
>
> Open source doesn't work without large scale enterprise or corporate investment
>
> Simply paying maintainers has the wrong incentive model and is not scalable. twitter.com/AmarachiAmaech...

2 10 57

Caleb Porzio
https://hubs.ly/H0rR4_X0

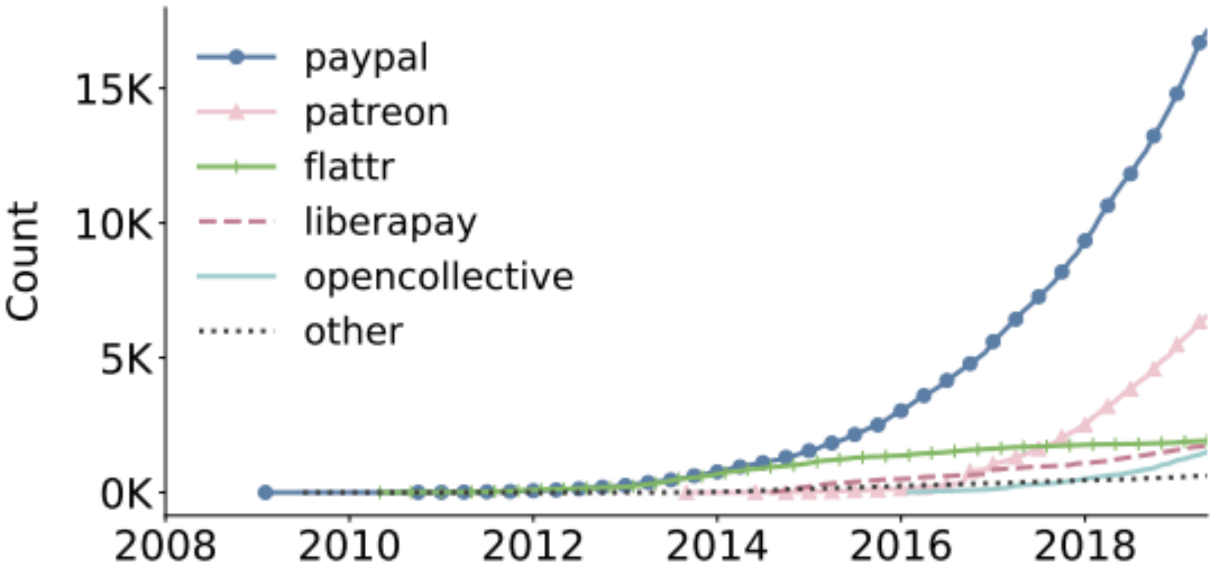
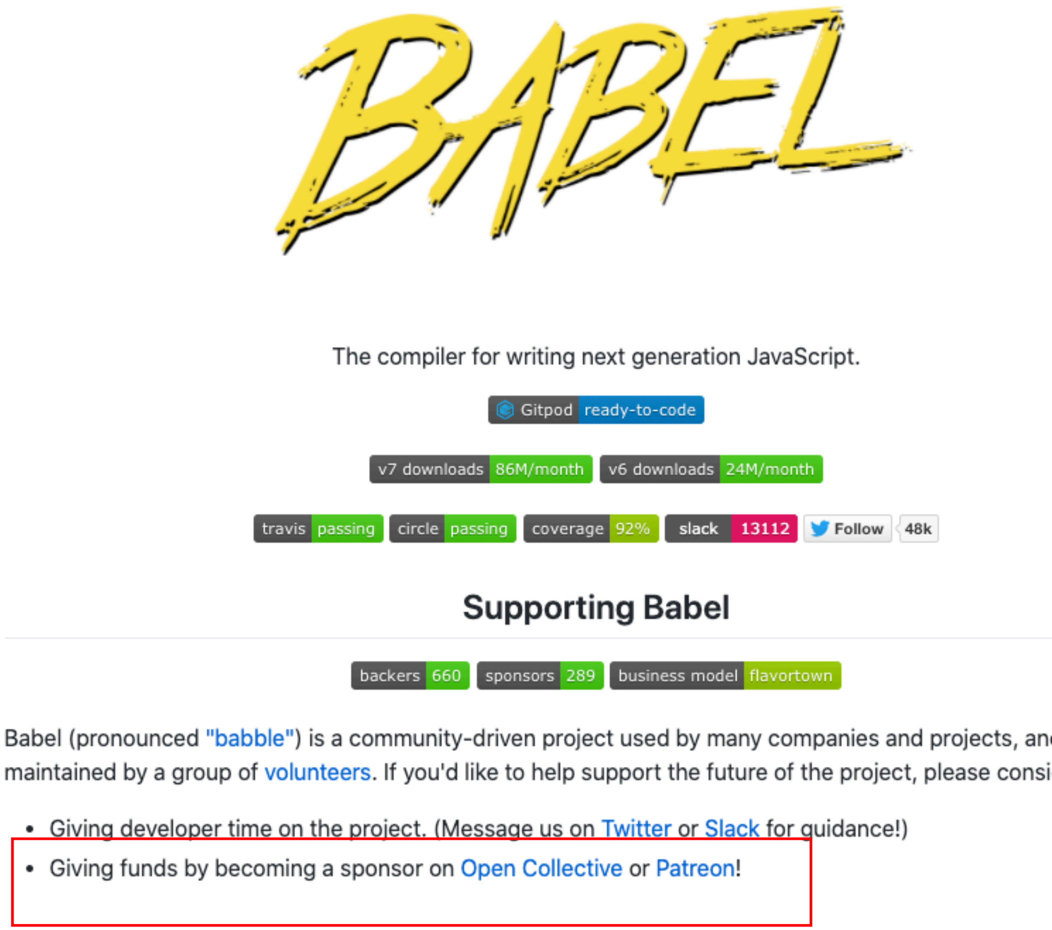


Figure 1: Adoption of donation platforms over time on GitHub (number of new non-fork repositories per month).

How to Not Get Rich: An Empirical Study of Donations in Open Source.
Overney, C., Meinicke, J., Kästner, C., and Vasilescu, B.
International Conference on Software Engineering, *ICSE*, ACM (2020).



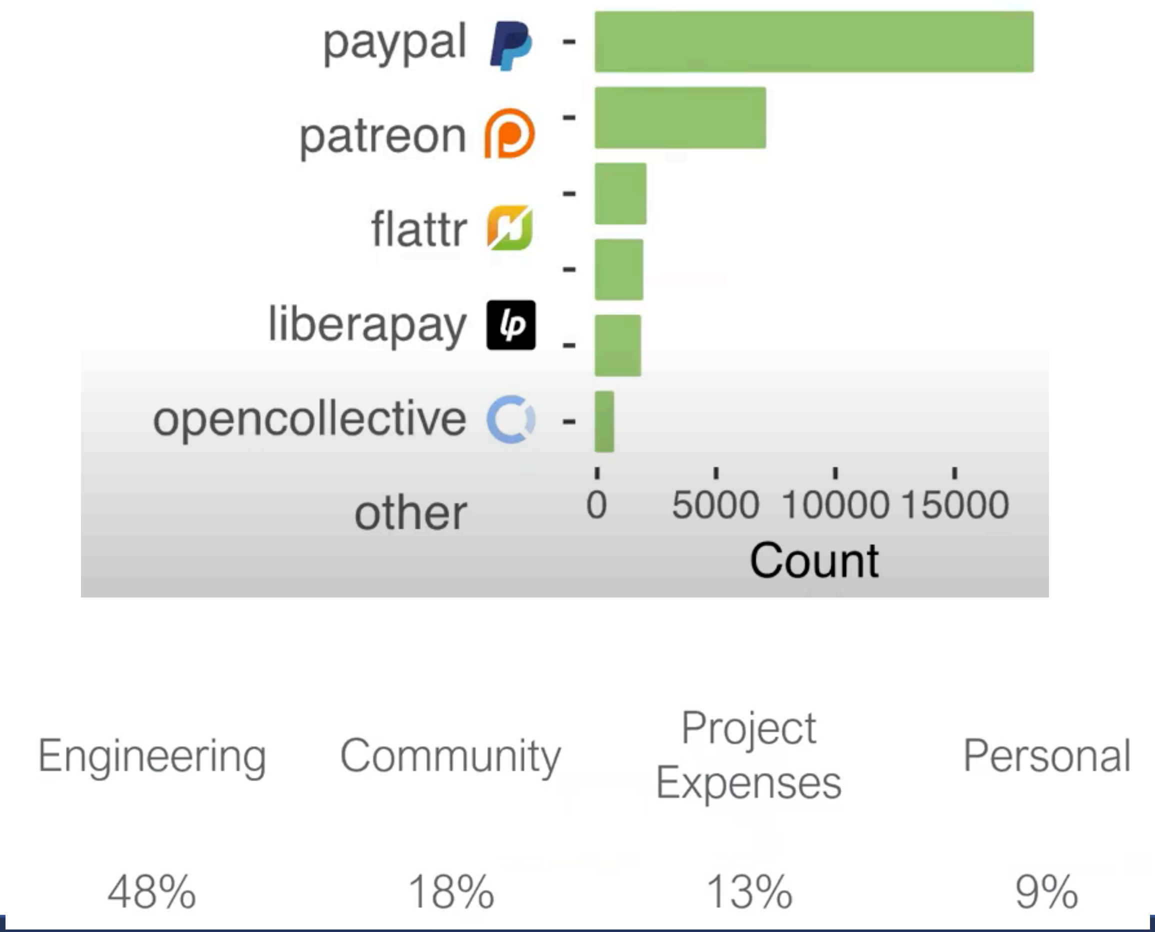
The compiler for writing next generation JavaScript.

Supporting Babel

Babel (pronounced "babble") is a community-driven project used by many companies and projects, an
maintained by a group of volunteers. If you'd like to help support the future of the project, please consi

- Giving developer time on the project. (Message us on Twitter or Slack for guidance!)
- Giving funds by becoming a sponsor on Open Collective or Patreon!

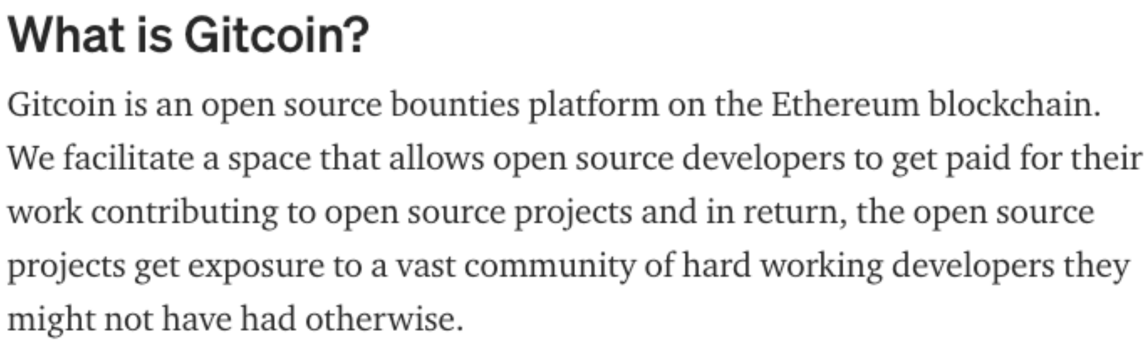
0.04% of repos on GitHub ask for donations



Engineering	Community	Project Expenses	Personal
48%	18%	13%	9%

- “I will be able to dedicate more time to the development and improvement of existing components and plugins.”
- “I can spend less time thinking about private monetization channels (e.g., taking on support/consulting contracts) and instead work more on content that benefits the entire community, e.g., more educational blog posts, videos and even books!”
- “The money from this Patreon keeps the servers for my projects running.”
- "The $3,000 per month will be put toward my living expenses and the student loan bills that I will need to start paying off during the project.”

GITCOIN



What is Gitcoin?

Gitcoin is an open source bounties platform on the Ethereum blockchain. We facilitate a space that allows open source developers to get paid for their work contributing to open source projects and in return, the open source projects get exposure to a vast community of hard working developers they might not have had otherwise.

(aka .. @owocki)

Summoner of Bots (All the things)

Favorite Gitcoin Feature

The Community

Favorite Food

Avocado Toast

Guest lecture 11/4

Post your questions on Quercus!

https://twitter.com/owocki

Quality?!

“There are no technical requirements for the plugins aside from them being able to be installed on a fresh Eclipse platform. We leave it to the community to find and report bugs related to technical features and conflicts.”

--Eclipse Marketplace, Dec 2014

Open Source Famous Phrases

Linus’s Law - Many eyes make all bugs shallow

Collaboration over Competition

…is open source code of higher quality?

- How would we be able to tell?

Open SSL/Heartbleed.

https://heartbleed.com/

- In 2013, OpenSSL made $2,000 in donations (and some from other sources)
- One full time programmer
- Heartbleed (2014): Vulnerability was found that effected about 17.5% of web servers (half a million)
- Used by Yahoo, Twitter, Google
- Who is responsible?

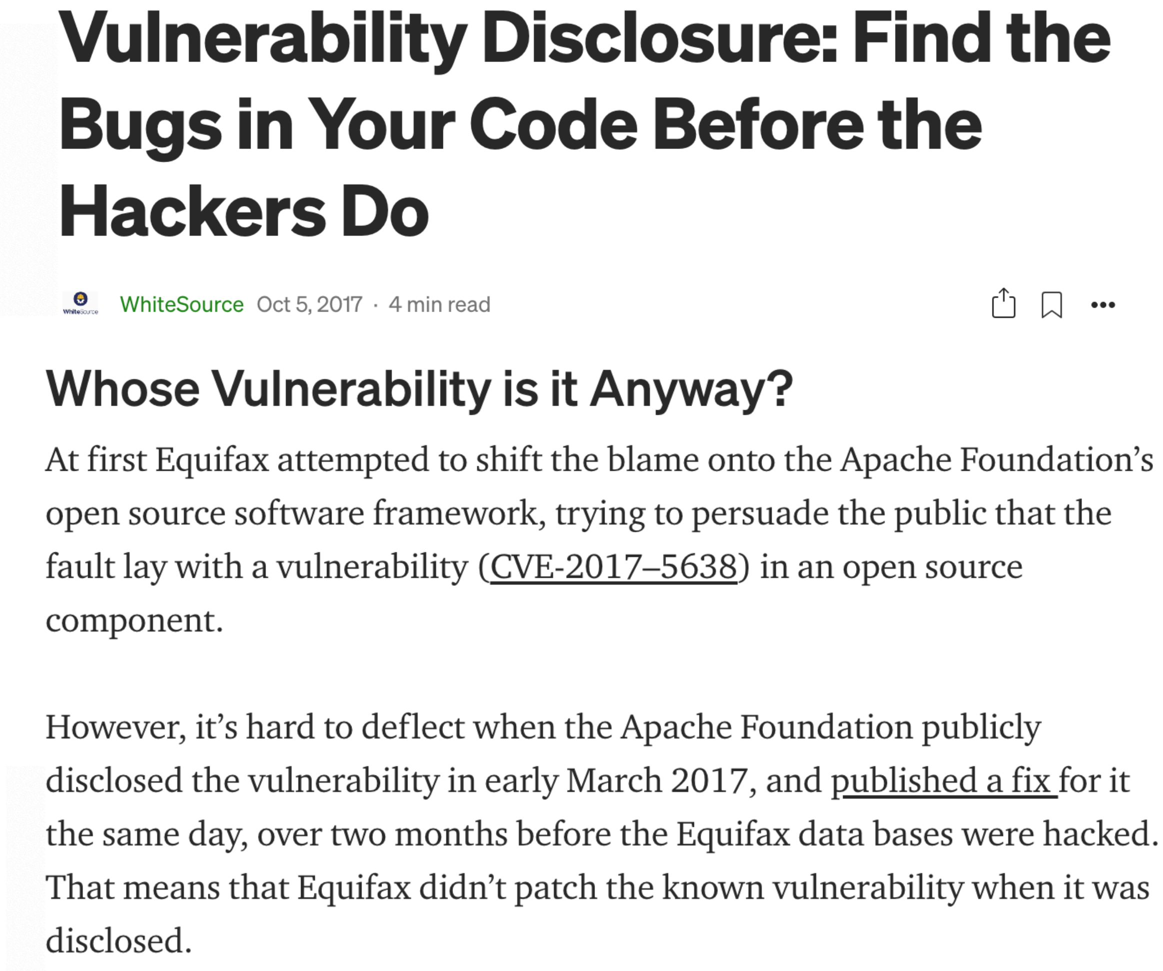
Case Study: OpenSSL

- When HeartBleed occurred, Google reported the bug and later submitted a patch
- After the HeartBleed bug, more than 17 companies agreed to each contribute $100,000 annually for 3 year to the Core Infrastructure Initiative.
- Core Infrastructure Initiative distributes funds to needy but important projects

Bug Bounties

- Facebook, Google, Yahoo, Microsoft, and other companies have rewards for finding bugs and reporting them
- Usually $100 or more for simple bugs and higher rewards for more serious bugs
- Bounties can save the company from malicious exploits, which can cost the company much more.
  - Ponemon Institute reports average cost of $3.79 million per company data breech (2014)

Options traders are betting that Equifax's stock will drop further following last week's announcement of a security breach. Reuters / Brendan McDermid



Vulnerability Disclosure: Find the Bugs in Your Code Before the Hackers Do

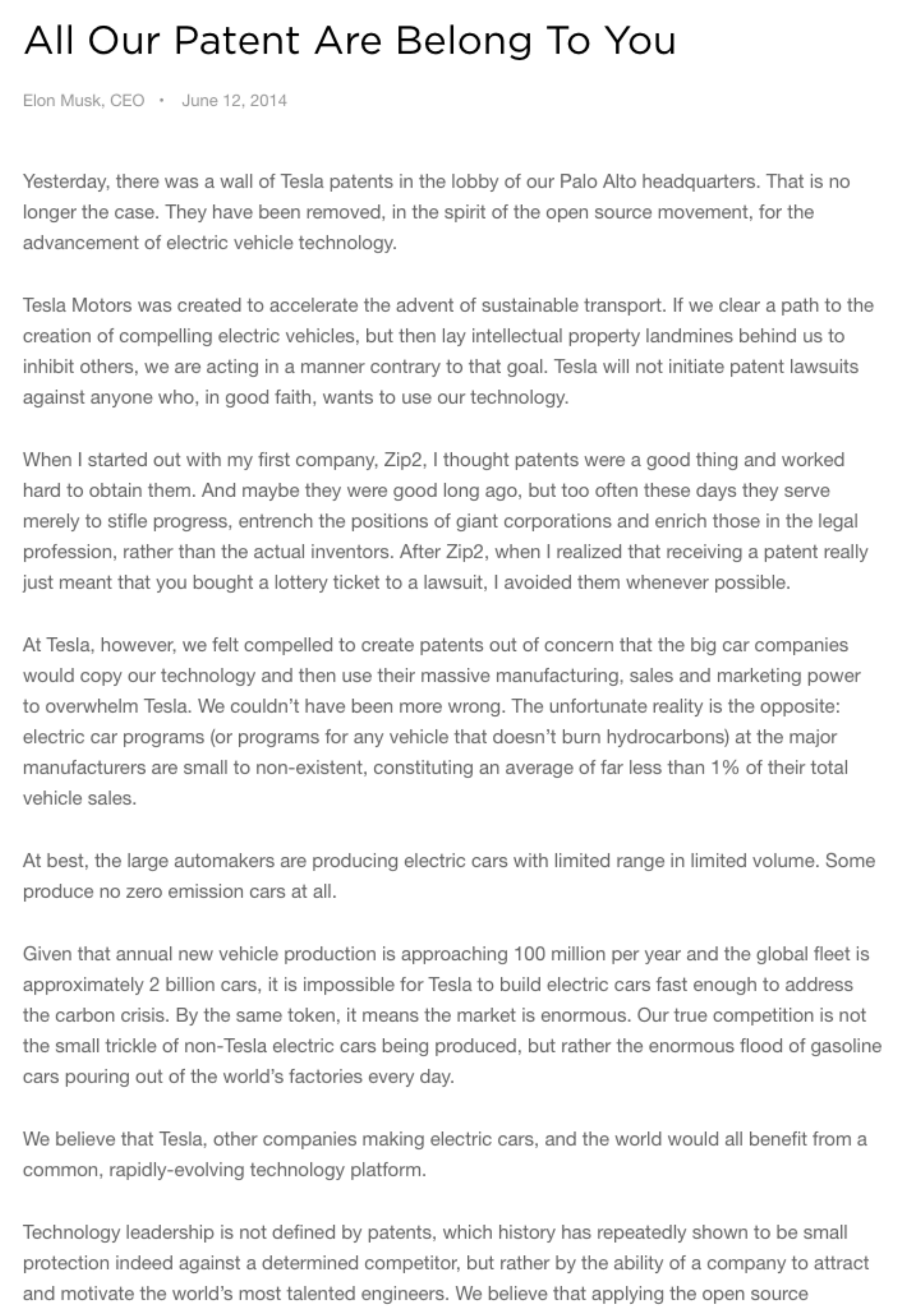
WhiteSource Oct 5, 2017 · 4 min read

Whose Vulnerability is it Anyway?

At first Equifax attempted to shift the blame onto the Apache Foundation's open source software framework, trying to persuade the public that the fault lay with a vulnerability (CVE-2017–5638) in an open source component.

However, it's hard to deflect when the Apache Foundation publicly disclosed the vulnerability in early March 2017, and published a fix for it the same day, over two months before the Equifax data bases were hacked. That means that Equifax didn't patch the known vulnerability when it was disclosed.

https://medium.com/@WhiteSourceSoft/vulnerability-disclosure-find-the-bugs-in-your-code-before-the-hackers-do-5c752e84142e



All Our Patent Are Belong To You

Elon Musk, CEO • June 12, 2014

Yesterday, there was a wall of Tesla patents in the lobby of our Palo Alto headquarters. That is no longer the case. They have been removed, in the spirit of the open source movement, for the advancement of electric vehicle technology.

Tesla Motors was created to accelerate the advent of sustainable transport. If we clear a path to the creation of compelling electric vehicles, but then lay intellectual property landmines behind us to inhibit others, we are acting in a manner contrary to that goal. Tesla will not initiate patent lawsuits against anyone who, in good faith, wants to use our technology.

When I started out with my first company, Zip2, I thought patents were a good thing and worked hard to obtain them. And maybe they were good long ago, but too often these days they serve merely to stifle progress, entrench the positions of giant corporations and enrich those in the legal profession, rather than the actual inventors. After Zip2, when I realized that receiving a patent really just meant that you bought a lottery ticket to a lawsuit, I avoided them whenever possible.

At Tesla, however, we felt compelled to create patents out of concern that the big car companies would copy our technology and then use their massive manufacturing, sales and marketing power to overwhelm Tesla. We couldn't have been more wrong. The unfortunate reality is the opposite: electric car programs (or programs for any vehicle that doesn't burn hydrocarbons) at the major manufacturers are small to non-existent, constituting an average of far less than 1% of their total vehicle sales.

At best, the large automakers are producing electric cars with limited range in limited volume. Some produce no zero emission cars at all.

Given that annual new vehicle production is approaching 100 million per year and the global fleet is approximately 2 billion cars, it is impossible for Tesla to build electric cars fast enough to address the carbon crisis. By the same token, it means the market is enormous. Our true competition is not the small trickle of non-Tesla electric cars being produced, but rather the enormous flood of gasoline cars pouring out of the world's factories every day.

We believe that Tesla, other companies making electric cars, and the world would all benefit from a common, rapidly-evolving technology platform.

Technology leadership is not defined by patents, which history has repeatedly shown to be small protection indeed against a determined competitor, but rather by the ability of a company to attract and motivate the world's most talented engineers. We believe that applying the open source

"in the spirit of the open source movement, for the advancement of electric vehicle technology".
-- Elon Musk

"Encouraging a healthy mix of collaboration and competition between market players — rather than diverting resources to heavy-handed intellectual property (IP) enforcement — would be the best way forward for what was still a nascent industry."

https://www.tesla.com/blog/all-our-patent-are-belong-you

Proprietary methods to gain community benefits

- Release early, release often; Continuous or small updates instead of big version changes
- “Many eyes make all bugs shallow”
- Recognize good ideas from your users.
- Collaboration over competition
- Promote users to report bugs and monitor new releases (easier if using software as a service)
- Allow users to write mods for the product (usually in a controlled way) or promote feature requests

Open Source Reality

- Aggressive collaborative tool use
  - version control, CI, issue tracker, reviews, …
- Careful management of people
- Process rigor
- Often aimed at expert users
- Intellectual property
- Often industry supported
- Often addressing common assets

similar to industrial practices