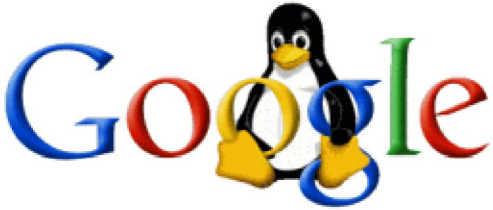
Linuxcon 2016



Open Source at Google

How We Use and Contribute to Open Source

Slides: http://goo.gl/NO0i2x
More Info: http://g.co/opensource

Marc MERLIN
merlin@google.com

The Early Days

Google's Commitment to Open Source

- Google wouldn't be around today without open source software.
- In its early days, Google had limited resources to contribute back to open source and most internal projects were not written with open-sourcing in mind… Nonetheless, we began contributing to open source projects in the early 2000's.
- Sadly, we were using a very old kernel and most of our patches against it were worthless to the community, so it took more time to upgrade to newer kernels and rewrite those patches against upstream for inclusion.

Google's Commitment to Open Source *(continued)*

- In 2004, Chris DiBona joined Google to head our open source efforts and was promoted to Director of Open Source. He created OSPO, our Open Source Program Office.
- He grew a team responsible for helping all Google employees releasing code as open source as well as ensuring that all teams and products are compliant with open source license requirements.
- You can find a partial list of open source projects Google has released at: https://developers.google.com/open-source/projects

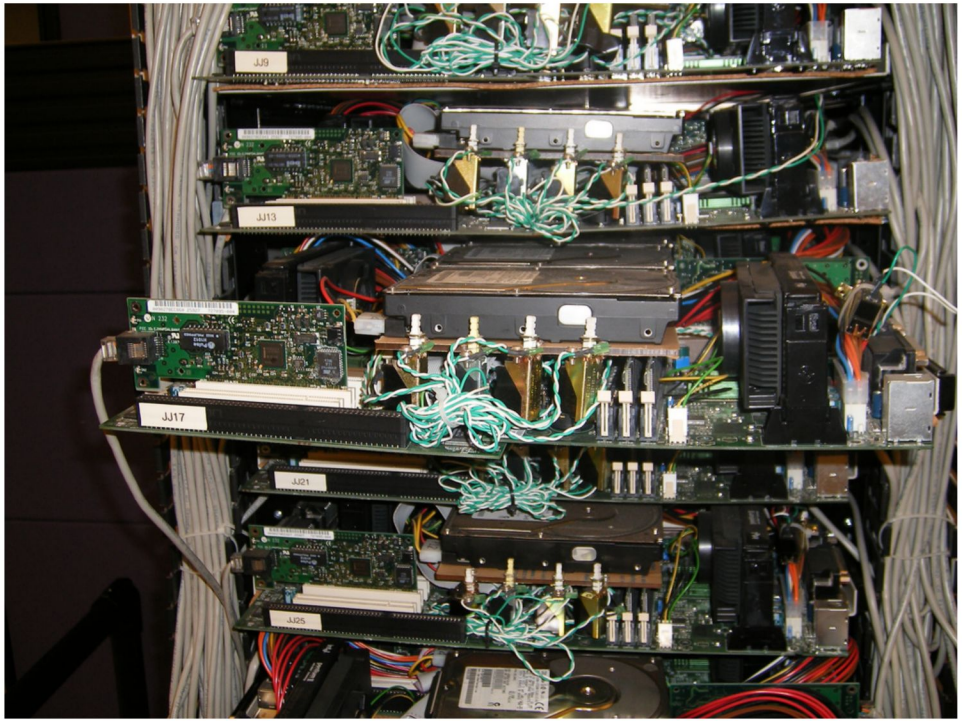
Open Source at Google, the Early Days

- In the late 1990's, Google was iterating very fast with few engineers, so a lot of the early code was written to be shared and linked across many projects for efficiency.
- However, this later made it difficult to release parts of our code as open source. Everything was intermingled.
- We were able to contribute many of our patches to the Linux kernel and other standalone projects, but this was harder for some of our other internal projects.

Open Source at Google, the Early Days *(continued)*

- For some projects like GFS, BigTable and MapReduce, we were able to publish papers describing the code and letting others implement it.
- We've released several thousand papers, including the one on MapReduce , which you can find here: http://research.google.com/pubs/papers.html

Our contributions to open source were sometimes as professional as our servers :)



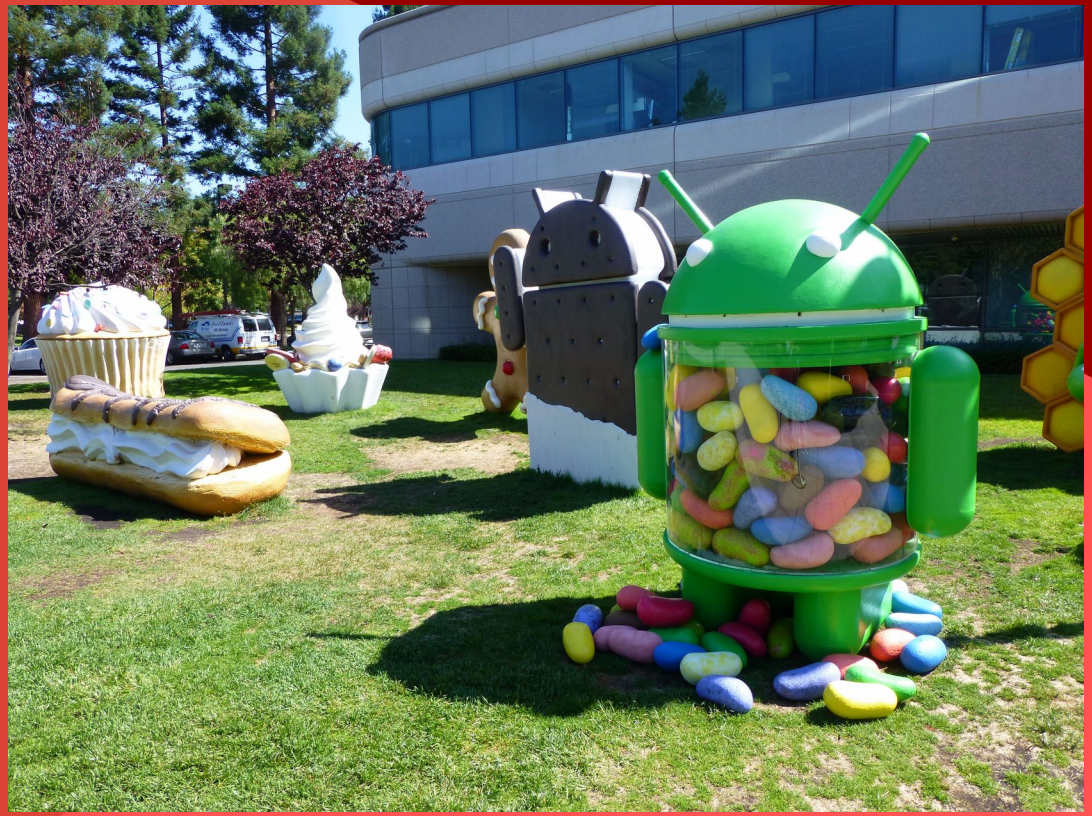
Contributions to the Linux Kernel

- Early on, we ran very old kernels and made quick and dirty patches that were of no interest to the kernel community.
- This was not a long term approach. We upgraded to newer kernels and began contributing patches, at first with fixes to existing code and later with new code like drivers, subsystems, or ideas that weren't taken as-is but re-implemented with similar features.
- You can find patches online going back to 2003 from Ross Biro, Chad Talbott, Frank Cusack, and Eric Uhrhane.

Contributions to the Linux Kernel *(continued)*

- We were often the first ones to find and fix obscure bugs that only happened at our scale. For instance, one rare IDE corruption bug we found in 2003 took over 1 million dollars in hardware to pin down and fix.
- LWN has details Google's kernel contributions up through 2009: https://lwn.net/Articles/357658/
- We release code whenever we can, though it takes time. Also, code and drivers for non released hardware often isn't useful to anyone else.

Our Contributions to Open Source Today



Contributions to the Linux Kernel *(continued)*

- We've contributed over 5,000 patches to the Linux kernel (the exact figure is hard to get as some developers use personal email addresses when submitting patches).
- We also employ people like Andrew Morton to help people, even people outside of Google, review and contribute patches to the Linux kernel.
- They range from small fixes to full drivers, or subsystems like containers which are now used by much of the Linux community.
- For hardware we release that runs Chrome OS or Android, we work to open source every driver we can and pressure hardware vendors into releasing hardware specs and allowing open source drivers.

OSS at Google is a lot more than Android ...

... and Chrome

Google
Chrome
Themes
Youtube
Google
Creator
31
MINION
chromethemes.net

Releasing Google Code as Open Source

- We have 6 people working on compliance covering use of open source software within Google and releasing our own code as open source.
- Just like we have Andrew Morton for the Linux kernel, we have other people whose primary job is to work on external open source projects like Jeremy Allison for Samba and Junio Hamano for Git.
- We also encourage all our employees to work on open source projects with suitable licenses, either on company time, or personal time.
- We have a process to allow employees work on personal projects not related to Google.

Releasing Google Code as Open Source *(continued)*

- We have countless engineers working on code that gets open-sourced, who work on external open source projects, who publish research papers to share with others, and who give talks like this one.
- Much of the code to run our clusters, which was written a long time ago in a way that was hard to open source, is being rewritten as open source and released as part of our Google Cloud offering, such as Kubernetes.
- New code is written from the ground up to be open-sourced in an internal Git repository, reviewed for license compliance and bugs to fix before release, and then released to one of our Github organizations (we have over 100 organizations that release open source code).

Releasing Google Code as Open Source *(continued)*

- The Google org alone contains over 800 projects as of 2016/03, and we have more than 3,000 open source projects on Github across all of our orgs.
- A few examples of our 100+ top level orgs on Github:
  - https://github.com/chromium
  - https://github.com/google
  - https://github.com/GoogleCloudPlatform
  - https://github.com/golang
  - https://github.com/ganeti
  - https://github.com/googleapis
  - https://github.com/tensorflow
  - https://github.com/angular

Contributing to Open Source Projects

- We ask employees to send patches using their Google email address when possible just to help spread the word about how much code Google releases and contributes as patches.
- We also ask them, as Googlers, to only contribute to open source projects with a suitable OSI approved license (no public domain, no WTFPL, no Unlicense, etc), as well as refrain from working on AGPL projects since they are not safe for us to interact with.

Contributing to Open Source Projects *(continued)*

- Some projects lack a license and as a result aren't legally open source until their source tree is fixed. Usually we can get the external project owner to fix their source tree, but sometimes not.
- In cases where employees want to work on projects without a clear license or a license that does not work for us, we have a process to allow them to contribute patches on their own time and equipment.
- To accomplish the above goals, we ask our employees to send us patches for a quick review and they then get blanket approval to patch outside projects that aren't problematic and have proper licenses.

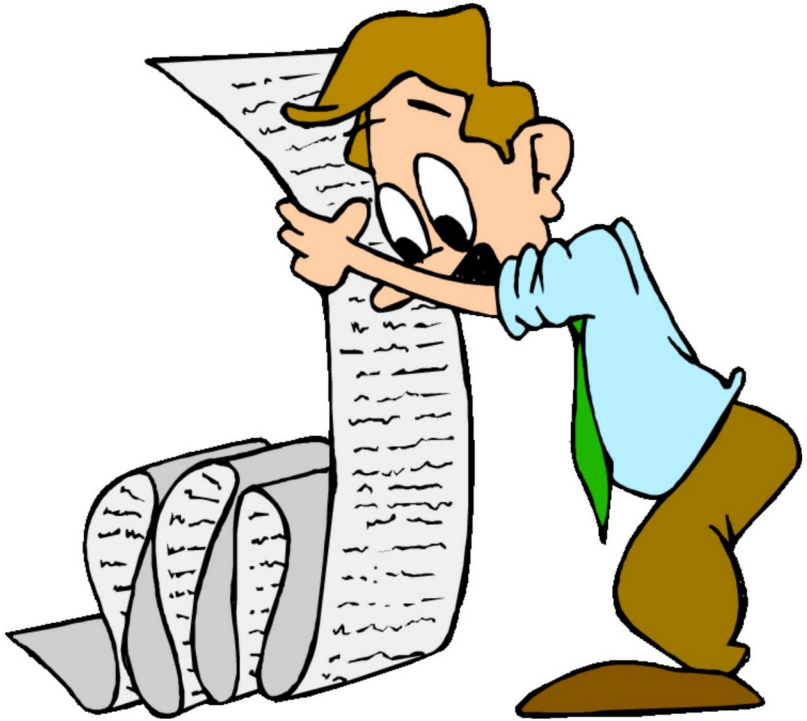
Using Open Source at Google

- We use open source just about everywhere. Being able to add features and fix bugs as quickly as we need them is a primary motivation, much more so than saving on licensing fees.
- All our production machines are running ProdNG, a stripped down Debian derivative: http://marc.merlins.org/linux/talks/ProdNG-LISA/ProdNG.pdf
- All our Linux desktops are running Goobuntu, a customized Ubuntu we maintain.
- Virtually all hardware we release is based on Linux:
  - Android (custom Linux distribution)
  - Chrome OS (based on Gentoo)
  - Chromecast, Android Auto, etc...

Partial list of 250+ events Google sponsors

- Akademy (KDE)
- Bioinformatics Open Source Conference
- ApacheCon EU
- AsiaBSDCon
- BADCamp
- BSDCan
- Calgary Open Source Systems Festival
- CiviCon (CiviCRM)
- Community Leadership Summit
- ConFoo.ca
- COSCUP
- DebConf
- DjangoCon US and EU
- Document Freedom Day
- dotGo
- DrupalCon NA and EU
- EclipseCon
- Embedded Linux Conf
- EuroBSDCon
- FiSL
- FOSDEM
- FOSS4G
- FOSSASIA
- FSCONS
- Gophercon
- GOSCON
- IndieWebCamp
- Libre Planet
- Linux.conf.au
- Linux Collab Summit
- LinuxCon NA/EU/JP
- LinuxFests and SCaLE
- Open Source Bridge
- OpenSym
- OSCON US and EU
- Penguicon
- PGCon
- PuppetConf
- PyCon US, AU and APAC
- Railsbridge
- SciPy
- State of The Map
- Strange Loop
- Graphical Web
- USENIX Annual Technical Conference (USENIX)
- VideoLAN Dev Days
- Wikimania

Partial list of 250+ events Google sponsors



Partial list of organizations Google supports

- Apache Software Foundation
- Blender Foundation
- COASTE program at CMU
- Creative Commons
- DAISY Consortium
- Document Foundation
- Eclipse Foundation
- FLOSS Manuals
- FreeBSD Foundation
- Freenet Project
- Free Software Foundation
- Free Software Foundation EU
- GNOME Foundation
- HackerDojo
- KDE
- Kernel.org
- Linux Foundation
- Core Infrastructure Initiative (CII)
- LLVM Foundation
- LWN.NET
- MetaBrainz
- MIT Innovation Lab
- NetBSD Foundation
- Network Time Foundation
- OASIS
- OpenBSD Foundation
- OpenHatch
- OpenID Foundation
- Open Invention Network
- Open Source Initiative
- OSU Open Source Lab
- Outreachy
- Python Software Foundation
- Simply Secure
- Software Freedom Conservancy
- Software Freedom Law Center
- Software in The Public Interest
- Standard C++ Foundation
- Women Who Code
- Xen Project
- TeX Users Group
- Tor Project

Partial list of open source work Google sponsored

- Accurate distributed system tracing infrastructure at PolyMTL
- Blender Open Movies
- Buildbot
- Clang
- CodeMirror
- C++ Standards
- DataNucleus
- EFF SSL Observatory
- Eigen
- Freenet
- FreeType
- GNURadio
- HTML5
- Kernel.org
- Libpurple
- LibreOffice
- Linux Foundation Documentation Project
- LuaJIT 2.0 to x86-64
- MAJAL
- Mercurial
- MetaBrainz projects
- M-Lab
- ModemManager
- Network Time Protocol
- Nigori
- Open Data Kit
- Ogg Theora/Vorbis decoder for ARM processors
- OpenROSA
- OpenSSL
- Open Usability
- PyPy
- R
- StartTLS.info
- SuiteSparse
- SWIG
- Tor
- Twisted
- Yaffs 2

… and many more through Google Summer of Code and Google Code-in.

Other Contributions to Open Source

- When SourceForge was becoming questionable as a repository for open source software, we helped fill the gap with code.google.com.
- After many years of good service, and with Github becoming a popular central platform, we decided to turn down code.google.com and help people migrate to Github.
- A lot of Google code is on Github, but we host some of our own big trees so as not to overload Github, such as: https://android.googlesource.com/

Other Contributions to Open Source *(continued)*

- Open source peer bonuses: We allow employees to nominate open source people outside of Google for a cash bonus for their work in open source.
- TODO Group: http://todogroup.org/
  - Google is a member of a group of companies that are committed to open source and want to collaborate on practices, tools and how to run successful open source projects and programs. This helps companies, even those who aren't members, to be better open source citizens.

Other Contributions to Open Source *(continued)*

- We have programs to expose students to open source. These bring new contributors into open source, connects students with mentors, and gives students real-world software development experience.
- Google Code-in: http://g.co/gci
  - Students ages 13 to 17 take on bite-sized tasks from accepted open source projects and compete for prizes and recognition.
  - Designed so that people of all skill levels can participate.
  - 6 years running, over 3,213 students from 99 countries have completed 17,271 tasks. In 2015 there were 14 organizations that created thousands of tasks from students to choose from.

Other Contributions to Open Source *(continued)*

- Google Summer of Code: http://g.co/gsoc
  - University students and open source projects apply to participate. Students propose what they'd like to contribute to a given open source project.
  - Accepted students are given a stipend so they can focus on development full-time, and paired with mentors who help as the students code over a period of 12 weeks.
  - 12 years running, over 12,000 students from 103 countries have contributed over 50 million lines of code to more than 515 open source projects. This year we accepted 1,206 students who are working with 178 open source projects.

Other Contributions to Open Source *(continued)*

- Financial contributions to the Software Freedom Conservancy
  - Open source is a two way street, we spend resources to be a good citizen and make an effort to lead by example.
  - We do our best to play by the rules, by the letter, and in spirit too by releasing Git history, not just a giant tarball, and having source available on the day of release.
  - We therefore support the SFC in their work to get more companies to comply with the letter and spirit of open source licenses.
- Financial contributions to Linux Foundation:
  - We're happy to help pay for Linus' salary
  - As well as other LF staff
  - And the many outreach efforts LF does
- Financial contributions to other organizations like the FSF.

Working with Licenses and CLAs

License Compliance

- We store all external open source code in a third_party hierarchy, along with the licenses for each project. We only allow the use of OSS under licenses we can comply with.
- Use of external open source is not allowed unless it is put in that third_party tree.
- This makes it easier to ensure we are only using software with licenses that we can abide by.
- This also allows us to generate a list of all licenses used by any project we build when they are released externally.

License Compliance *(continued)*

- Big projects like Chrome OS, Chrome (the browser) and Android have specific code that compiles a list of all open source licenses and copyrights used by the project, and outputs a combined license file with all licenses and copyrights.
- Example from Chrome OS (about/credits): http://marc.merlins.org/mirrors/about-credits.html
- Many do not know that MIT and BSD licensed software requires its users to display a copyright notice for them, along with the name/entity that holds the copyright. This means you cannot just give a list of packages and say they are BSD licensed, you need to display a separate copyright line with name for each of them.

License Compliance *(continued)*

- On the flip side, GPL software requires you to display the GPL license text in a way that can be shared amongst all packages that use it.
- This means that generating a combined license file is more work with MIT and BSD software than GPL.
- License file generator that is part of the Chromium OS source tree generates an up-to-date license file for each build of Chromium OS/Chrome OS.
- We make sure to list copyrights of licenses like BSD or MIT inside JavaScript code used on our web pages.
- Before we acquire a company, we check whether they are in compliance with open source licenses and get them into compliance, or don't acquire them.

This should not be
construed as
unlicensed
practice of law.

Licenses we usually use (incomplete description of each)

- Apache 2: This is our default license, offers explicit patent license to the user.
- MIT/BSD: Basic OSS licenses. Users of BSD/MIT code are required to reprint the author's name and copyright notice, they are not required to share the code.
- GPLv2: Does not require copyright attribution statements. Does require sharing the code, may or may not offer patent license.
- GPLv3: Adds explicit patent license to GPLv2, also requires developer switch on hardware (OK for Chrome OS, maybe not for Android or self driving cars).

We can’t use or contribute to WTFPL licensed code

- WTFPL is a poorly written license not reviewed by lawyers.
- The Wikipedia entry explains why licenses need to list the rights given to the user: https://en.wikipedia.org/wiki/WTFPL#Effectiveness_as_license_or_waiver
- Stating “do anything you want” may not have proper standing in court.
- WTFPL does not disclaim warranties, putting the author and any contributor, at risk of being sued by users: https://en.wikipedia.org/wiki/WTFPL#Characteristics
- As a result, we don't work with WTFPL code and encourage people to use Apache 2, MIT or BSD if they want a simple license without restrictions.

We can’t use/contribute to Unlicense licensed code

- Similar to WTFPL, Unlicense is another attempt at removing necessary legalese, making the license non-binding in many cases.
- More information:
  http://programmers.stackexchange.com/questions/147111/what-is-wrong-with-the-unlicense

We can’t use or contribute to CC0 licensed code

- Section 2 of CC0 includes a waiver which has contributors give up all rights to their code.
- We want to be able to retain rights to the code we wrote even if we contribute it to another project.
- More information: https://creativecommons.org/publicdomain/zero/1.0/legalcode

Dedication to the Public Domain

- Public domain is not an open source license. It offers no protections to us or disclaimer of warranties.
- More generally, in the US, things can fall in the public domain after several decades, but In the 1970's Congress stopped people from being able to irrevocably transfer copyright, which includes by implication the public domain. Congress did this to protect authors, but it has had several (possibly) unintended consequences for the free software movement.
- The result is that people may retain their copyright even if they try to give it up.
- This is why public domain doesn't exist in most cases in the US.
- More information: http://scholarship.law.uc.edu/cgi/viewcontent.cgi?article=1146&context=fac_pubs

We can’t work with AGPL licensed code

- AGPL is a proper open source license and is meant to prevent a company from using your code and building a proprietary product on top of it. That’s a reasonable intention.
- Unfortunately the terms are impossible for us to comply with it and therefore stop us from using or contributing to AGPL code.
- If you use AGPL code, even unmodified, on a website, you must credit it and give a download link to the unmodified source. If you’re using that software to display your site, it’s fair and reasonably easy to do. If a portion of your web page was generated by an AGPL grep, filling your web page with links and licenses for all tools that were involved in creating/displaying it, can be harder.

We can’t work with AGPL licensed code *(continued)*

- The cost of ensuring that we don't have AGPL code involved in any way in one of our externally facing products is too high compared to the price of finding an alternative or writing our own under a less restrictive license.
- As a result, we do not have any AGPL code in our source tree, nor do we feel it’s safe to talk to AGPL code, even unmodified. Chris DiBona explains this in more details here: http://www.theregister.co.uk/2011/03/31/google_on_open_source_licenses/
- Google has a policy to upstream all internal patches by default, but we can't use AGPL software or contribute to it.

Working with CLAs

- CLAs (Contributor License Agreements) are often seen as a dirty three letter acronym :)
- They require that someone contributing code to a repository signs an agreement for their code to be accepted.
- A CLA can say anything, some companies or projects have terms in their CLA that you may not find acceptable.
- Google employees cannot sign a CLA unless it has first been reviewed by our open source lawyers.
- Plenty of CLAs are fine and we approve them. With some companies, we have company-level CLA signing, so our employees do not need to agree individually.

Working with CLAs *(continued)*

- Google also requires contributors to sign a CLA. It's an unmodified copy of the Apache CLA that allows Google to relicense your code if need be as well as gives Google patent grants to the code you are contributing (contributing to an Apache 2 licensed project gives a patent grant to the project, too).
- You retain ownership to your code, you are just granting a license to Google.
- Projects that do not use a CLA that gives you the right to sublicense are in trouble if they ever need to relicense their project, even from WTFPL to Apache 2, because they need to contact every single person who contributed code and get their permission. With a CLA, this becomes a non-issue.
- More information: https://www.apache.org/licenses/icla.txt and https://cla.developers.google.com/clas

If you have legal questions...

質問がありますか？

Q&A

Marc Merlin
merlin@google.com

More Information
http://g.co/opensource
http://goo.gl/NO0i2x