

the future of Open Source

scott chacon



a short history of
open source

why
free software



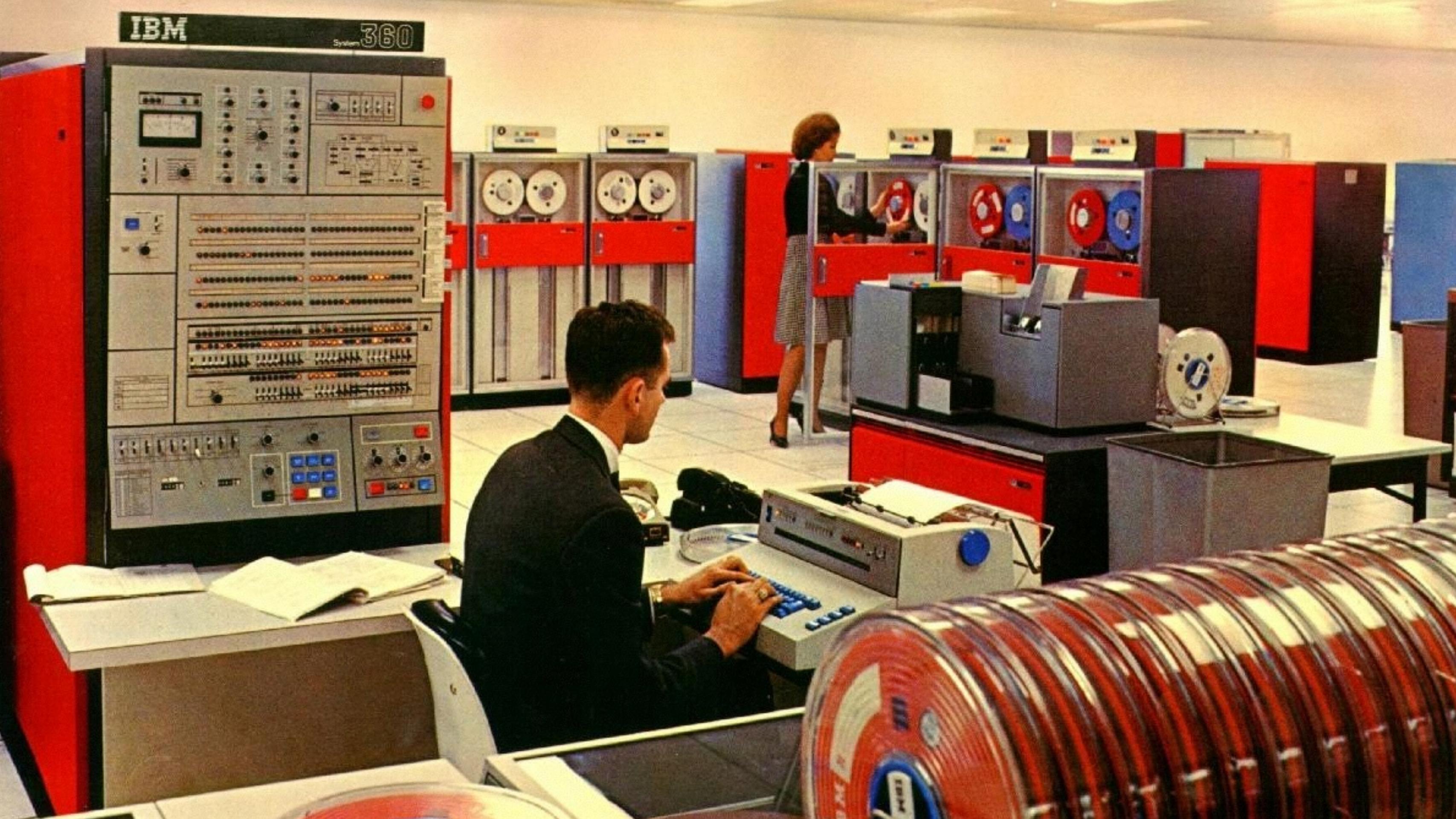
Rear Admiral **Grace M. Hopper**, USN, Ph.D.

the UNIVAC IA2 system



IBM

System 360



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



Richard Stallman
is in Town



the GNU Manifesto

“GNU, which stands for Gnu's Not Unix, is the name for the complete Unix-compatible software system which I am writing so that I can give it away free to everyone who can use it”

the GNU Manifesto

“Software sellers want to divide the users and conquer them, making each user agree not to share with others. I refuse to break solidarity with other users in this way. I cannot in good conscience sign a nondisclosure agreement or a software license agreement. For years I worked within the Artificial Intelligence Lab to resist such tendencies and other inhospitalities, but eventually they had gone too far: I could not remain in an institution where such things are done for me against my will.”



why
free software

why
open source

*"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.

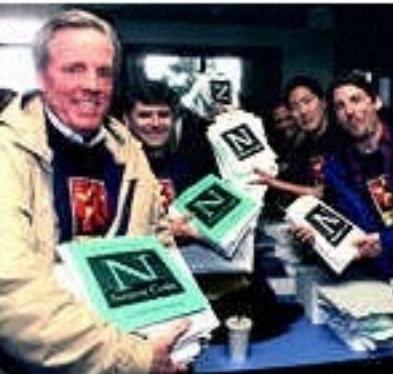
Release early. Release often.

With enough eyes, all bugs are shallow.

When you lose interest in a program, your last duty to it is to hand it off to a competent successor.

The next best thing to having good ideas is recognizing good ideas from your users.

Netscape sets source code free



Netscape CEO Jim Barksdale and the development team present the code

March 31, 1998 12:10 PM PST CNET News

After three months of anticipation, [Netscape Communications \(NSCP\)](#) today finally released the source code for its Communicator suite.

Netscape this morning unveiled the much-anticipated release with a teleconference featuring breathy executive statements touting the significance of the move. The company actually posted the approximately 8 megabytes of compressed Communicator 5.0 code at 10 a.m. PT to [Mozilla.org](#), the site Netscape has set up to be the central clearinghouse for source code-related information.

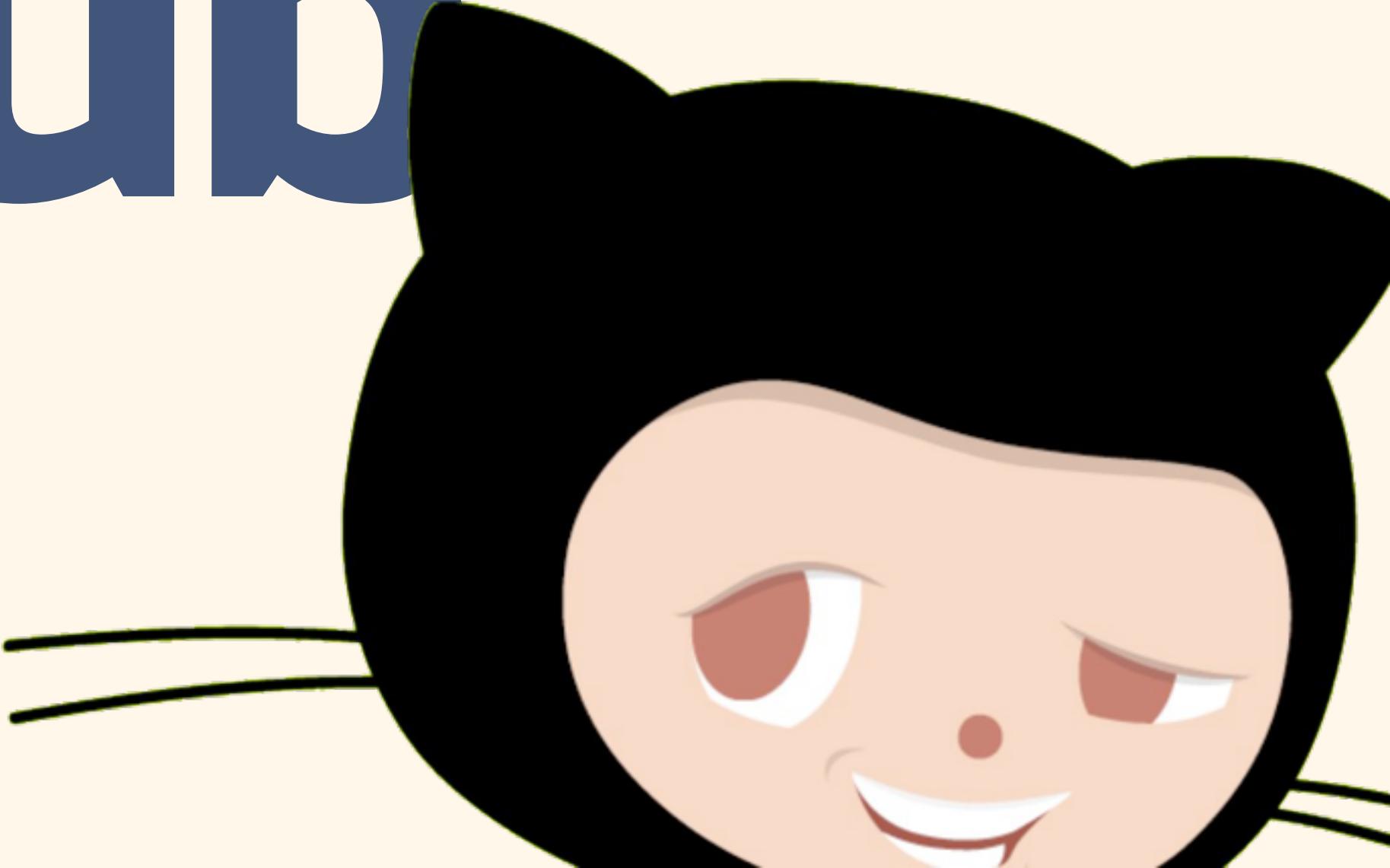
www.guruprasad.net

The conferees believed the **pragmatic, business-case grounds** that had motivated Netscape to release their code illustrated a valuable way to engage with potential software users and developers, and convince them to create and improve source code by participating in an engaged community. The conferees also believed that it would be useful to have a single label that identified this approach and **distinguished it from the philosophically- and politically-focused label "free software."**

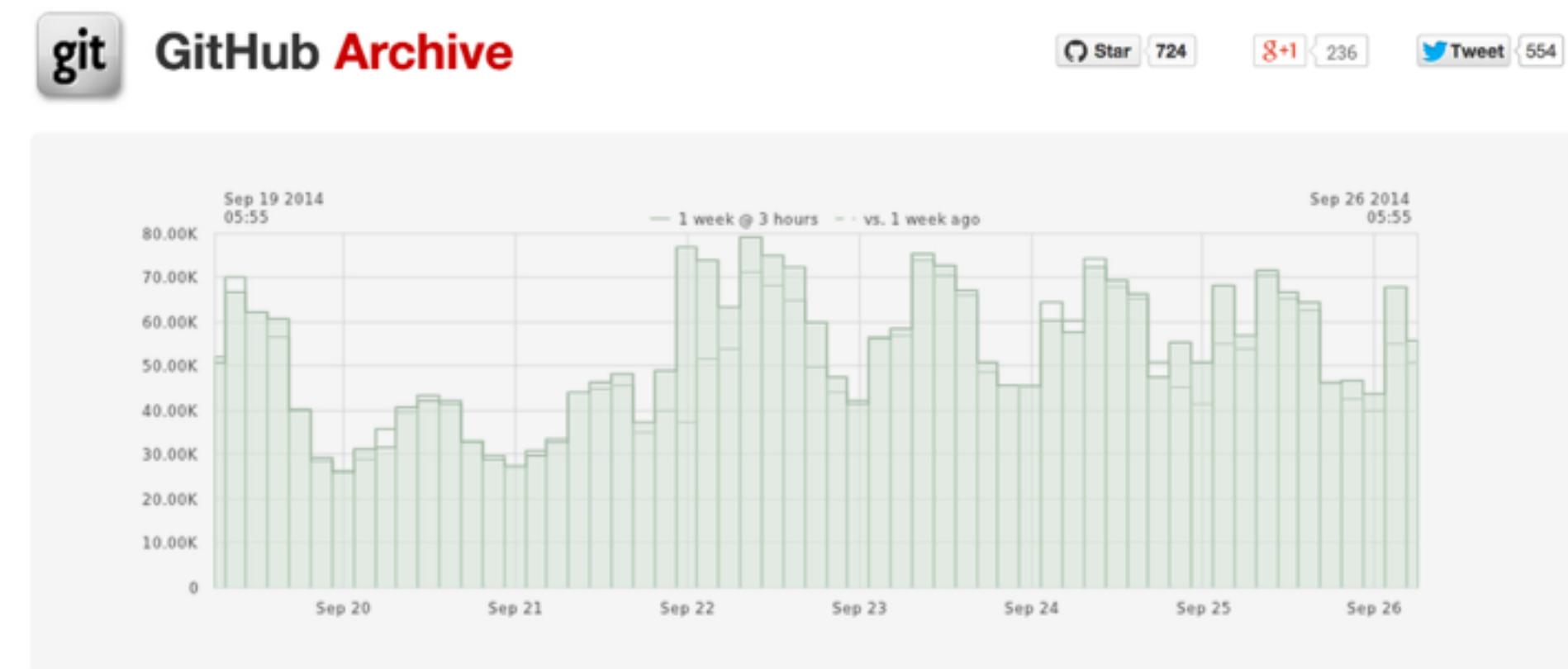
16 years

the current state of
open source

Git**H**ub



<http://www.githubarchive.org/>



Open-source developers all over the world are working on millions of projects: writing code & documentation, fixing & submitting bugs, and so forth. GitHub Archive is a project to **record** the public GitHub timeline, **archive it**, and **make it easily accessible** for further analysis.

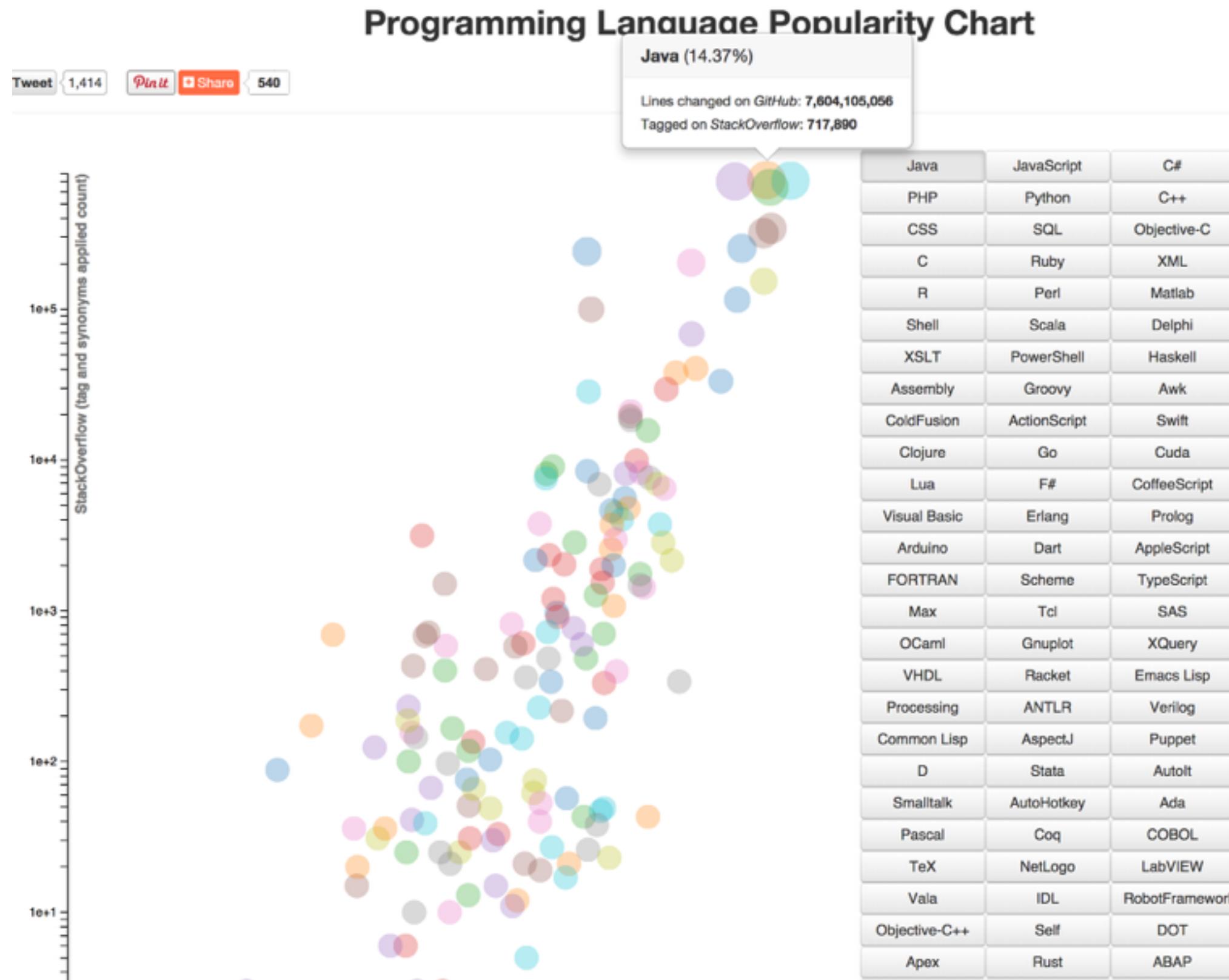
Looking for the [daily top new & watched repository reports](#)? [Sign up here](#).



GitHub provides [18 event types](#), which range from new commits and fork events, to opening new tickets, commenting, and adding members to a project. The activity is aggregated in hourly archives, which you can access with any HTTP client:

Query	Command
Activity for April 11, 2012, 3PM UTC	<code>wget http://data.githubarchive.org/2012-04-11-15.json.gz</code>
Activity for April 11, 2012	<code>wget http://data.githubarchive.org/2012-04-11-{0..23}.json.gz</code>
Activity for April 2012	<code>wget http://data.githubarchive.org/2012-04-{01..31}-{0..23}.json.gz</code>

<http://ghtorrent.org/>





Third Annual GitHub Data Challenge

July 22, 2014

derekgr

General

Edit



New blog post

Featured

All Posts

New Features

Engineering

Enterprise

Meetups

New Hires

Watercooler

Subscribe

The Details

Entries are generally visualizations, prose descriptions of data analyses, or both. We love innovative entries, so an "entry" is defined somewhat loosely.

There are only three rules:

1. To enter, you must fill out our [submission form](#) by **midnight PDT on August 25th, 2014**.
2. Your entry needs to use publicly available GitHub data from any number of available sources described below.
3. Show your work! Whatever you submit needs associated code or documentation describing what data you used and how you processed it. Some examples of what we're looking for include code (and instructions to use it) in a GitHub repository, an academic write-up of your analysis, or an informal prose write-up. If you're not linking to a repository, you should submit a [Gist](#) with your documentation.

After the submission deadline on August 25th, GitHub employees will review and vote on all entries to pick the three top winners. We'll send out notifications to those top three by mid-September.

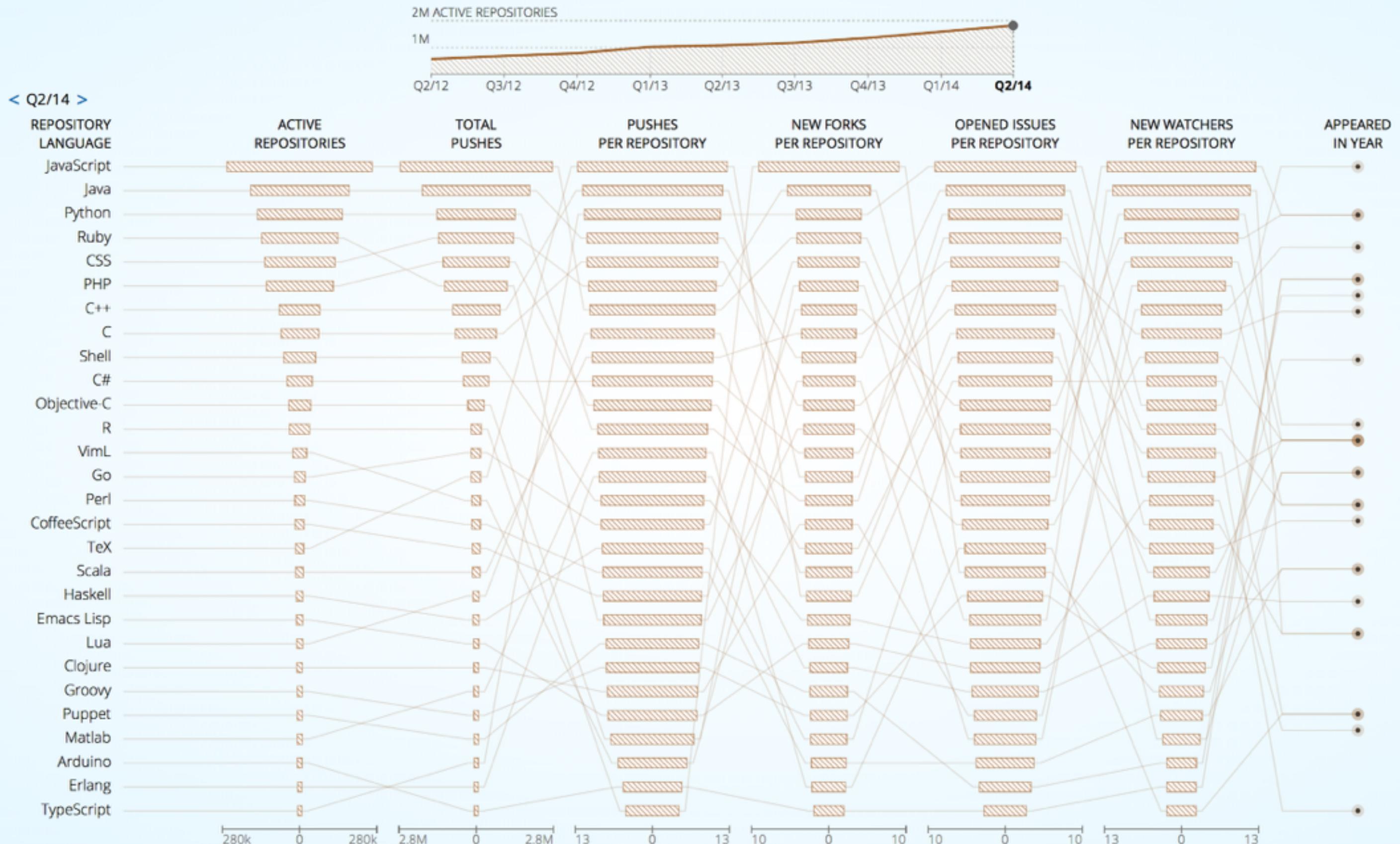
Data Sources

GitHub activity data is available from several publicly-available sources. Here are a few links to get you started:

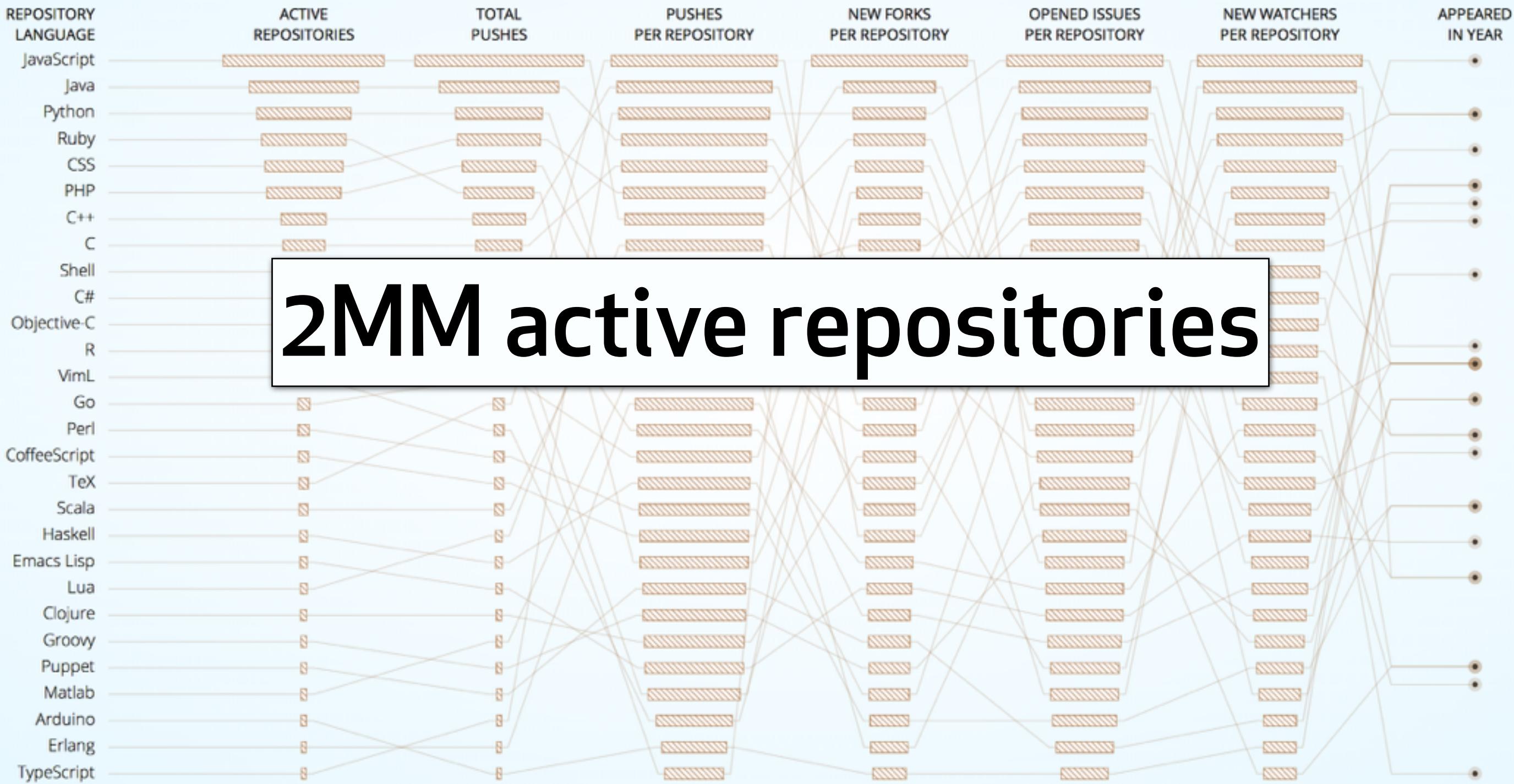
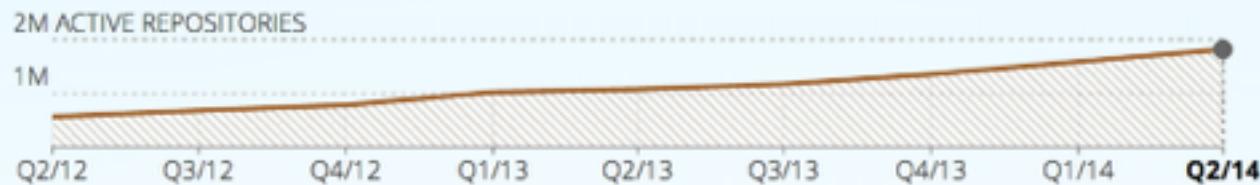
- Our very own [API](#).
- The [GitHub Archive](#), providing historical archives of our public timeline data.
- [Google BigQuery](#), where GitHub's public timeline is a featured public dataset: see the

GitHut

A SMALL PLACE TO DISCOVER LANGUAGES IN GITHUB



< Q2/14 >



280k 0 2.8M 0 2.8M 13 0 13 10 10 0 10 10 13 0 13 10 13 0 13

The GitHub Open Source Development Process

Kevin Peterson
kevin@kevinp.me

Abstract

Open Source Software (OSS) projects, or software projects with publicly available source code, are realizing ever more important roles in both personal and business computing. As such, shifts in the way in which OSS is developed could have impacts on both the quantity and the quality of OSS projects.

The development process by which these projects are produced is generally unstructured compared to commercial software, but many projects do exhibit general development patterns. GitHub, a popular OSS code hosting website, along with Git, the site's Source Code Management (SCM) tool of choice, may have the potential to fundamentally change this process by facilitating new patterns and opportunities for developers.

By analyzing a subset of GitHub repositories, this report will show how GitHub has influenced some intrinsic aspects of traditional OSS development, such as developer hierarchies and issue close velocity. We find that many of the traditional aspects of OSS development remain, such as most project development being done by a small group of core developers. Other traditional assumptions about OSS developer hierarchy, such as a large number of Issue Reporters compared to Committers, seems unsupported by the GitHub data. We conclude that GitHub represents an evolution of the OSS development process, and not necessarily a large shift.

Keywords. Git, GitHub, Open Source

1 Introduction

Open Source Software (OSS) has fundamentally changed how we view the Software Development Process [25]. OSS projects are not only viable, but successful and thriving.

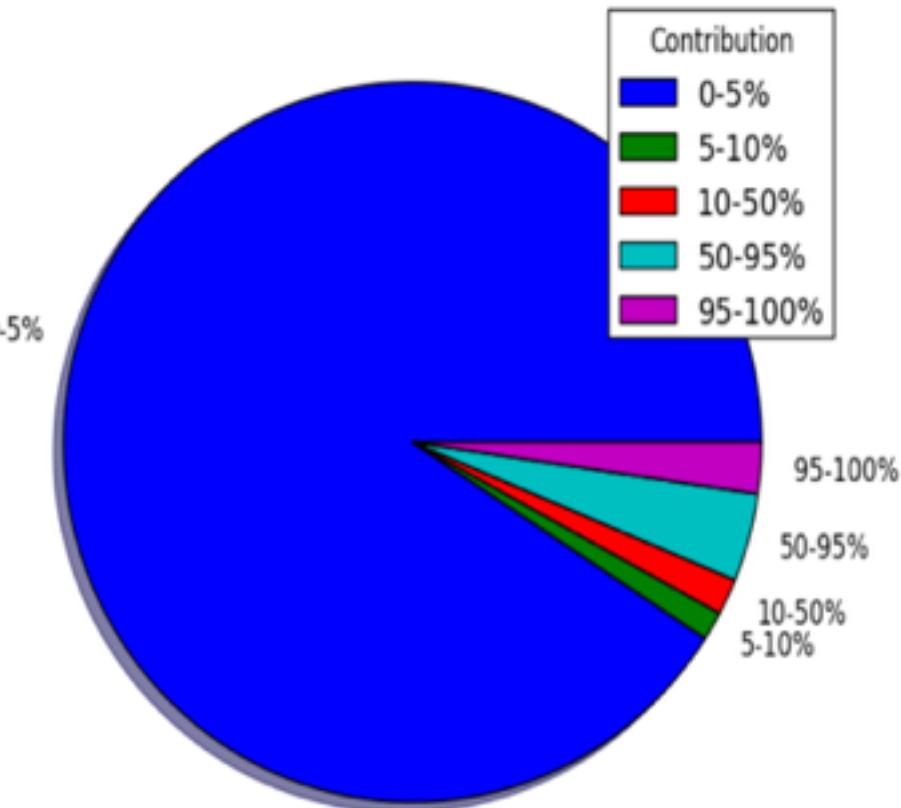
A case study of the Apache Server project [20] showed that a dedicated community can produce software that rivals or exceeds commercial offerings. Furthermore, it showed that even in the unstructured context of OSS, certain structures, hierarchies, and codes of conduct emerge. Contrast the Apache Server project to a successful project on GitHub. Although the spirit and intent may be the same, the tool set is drastically different. The intent of this work is to explore how GitHub facilitates this process, and also how it may be causing it to evolve.

GitHub¹ is a popular code hosting website that uses Git Source Code Management (SCM).² Along with hosting software repositories, GitHub incorporates the social aspects of development. GitHub users have community-visible profiles, and user actions can be tracked and followed by other community members. This type of social integration of a user's identity and actions is unique to GitHub [11].

Git itself, the SCM of GitHub, lends itself to the collaborative nature of GitHub by allowing for development to take place in a more distributed manner than previously available in other SCM systems [27]. Git also allows for a variety of work flows [8]. These work flows may be tailored to the individual needs of a project. The Linux Kernel, for example, uses a Dictator and Lieutenants Workflow [24], which is hierarchical in nature. Other work flows may be more distributed, or resemble more traditional centralized SCM systems. Although work flow techniques aren't explicitly addressed in this work, it is important to note that work flow change may contribute to process change.

To better understand quantitatively the GitHub process, three hypotheses are introduced. The goal of these hypotheses is to bridge the gap between the more qualitative research of GitHub [11, 4] and the statistical analysis of data from a random sampling

¹<https://github.com/>
²<http://git-scm.com/>

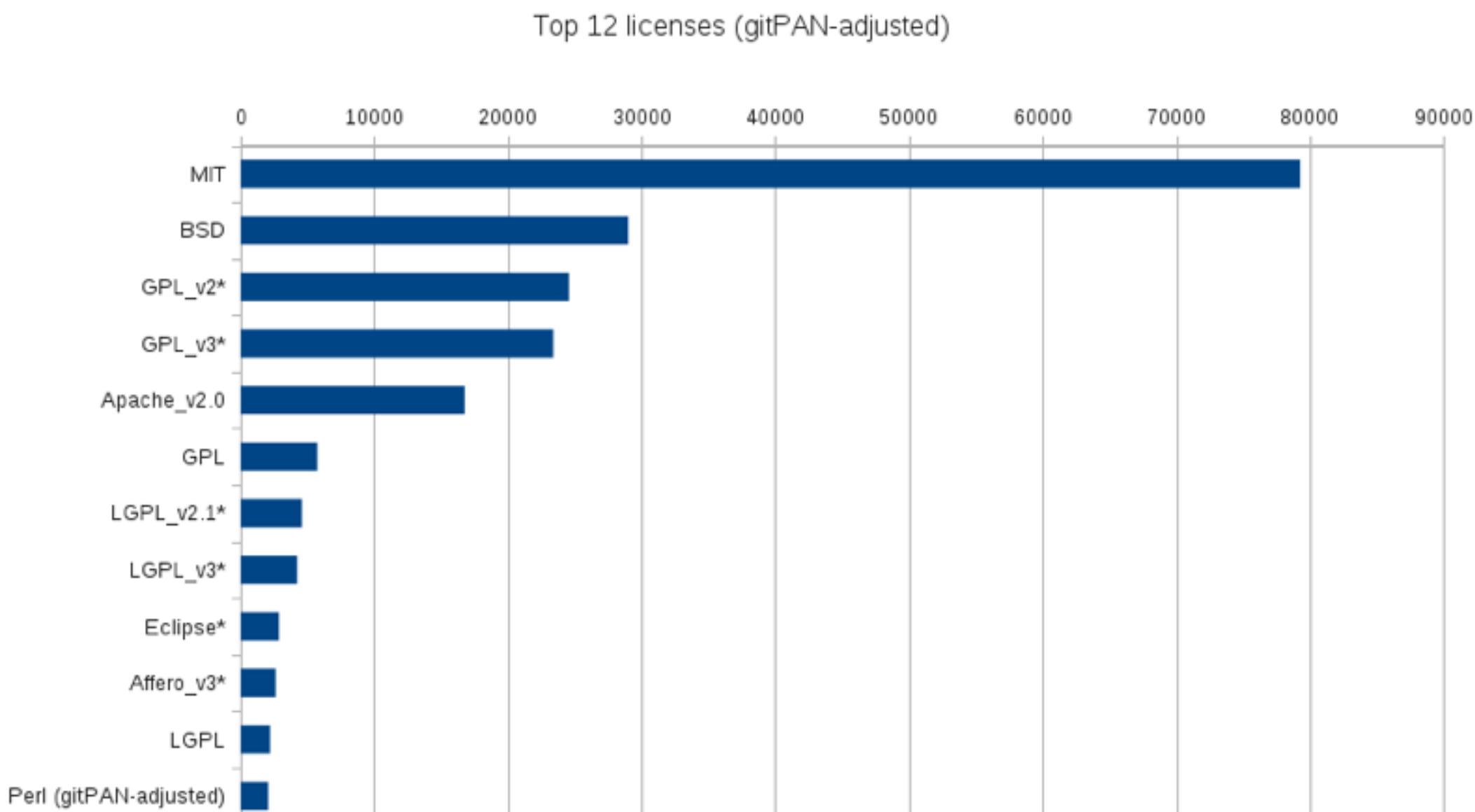


proliferation of
licenses

Licensing of Software on Github: A Quantitative Analysis



Aaron Williamson
Senior Staff Counsel
Software Freedom Law Center



**monkchips**

@monkchips

[Follow](#)

younger devs today are about POSS - Post open source software. fuck the license and governance, just commit to github.

[Reply](#) [Retweet](#) [Favorite](#) [More](#)RETWEETS
60FAVORITES
39

8:34 AM - 17 Sep 2012



monkchips @monkchips · 17 Sep 2012

younger devs today are about POSS - Post open source software. fuck the license and governance, just commit to github.

Reply Retweet Favorite More



Simon Phipps

@webmink



Follow

@monkchips ... and a whole new generation of lawyers thank them for it.

Reply Retweet Favorite More

RETWEET

1

FAVORITE

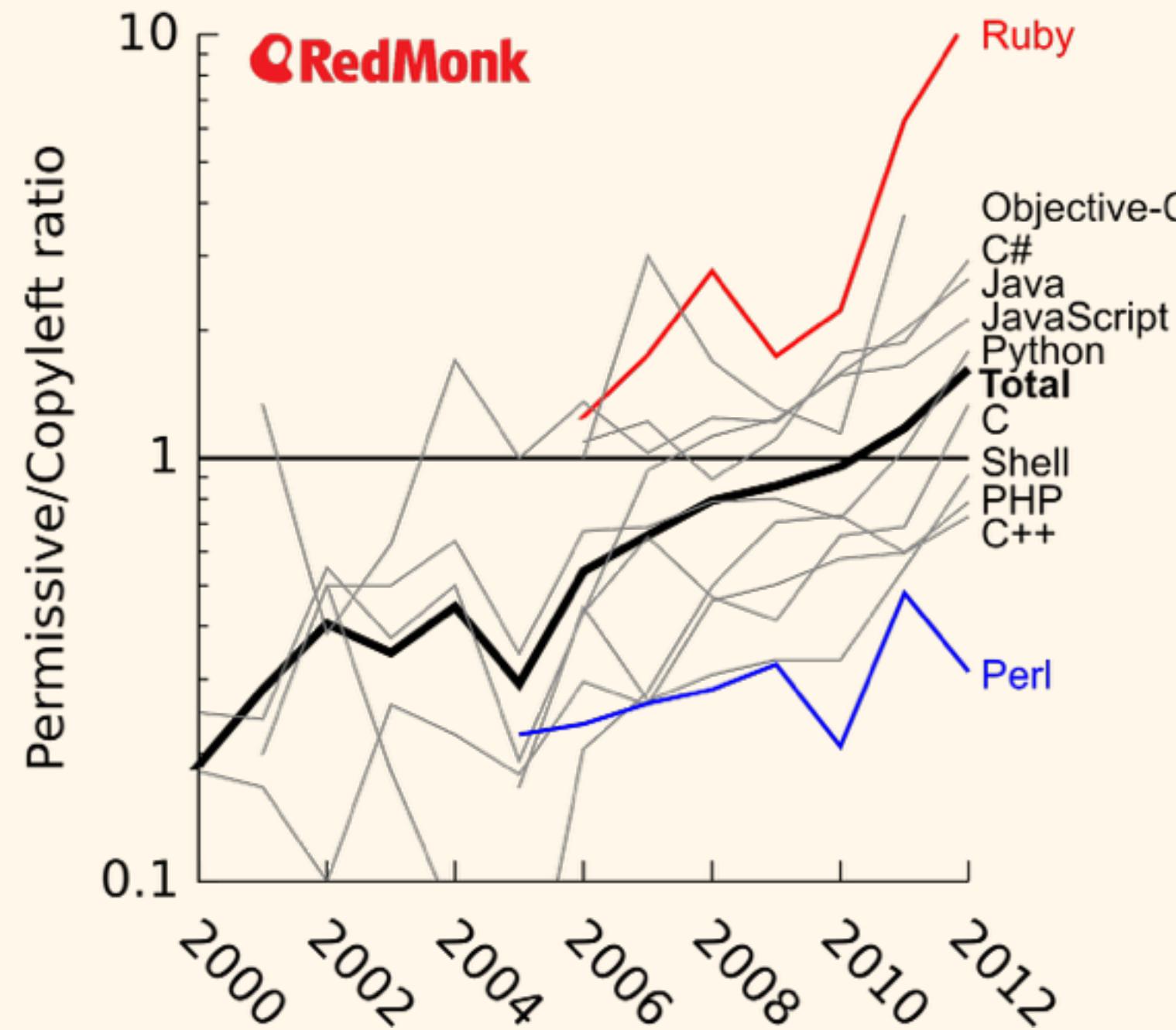
1



3:16 PM - 17 Sep 2012

decline of the
gpl

The shift toward permissive licensing



the involvement of
corporations



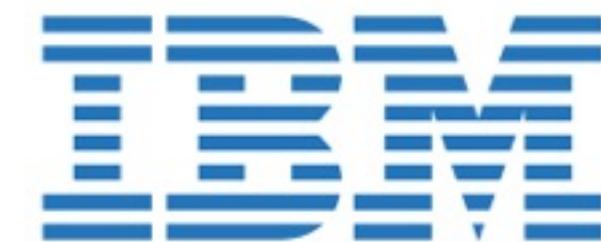
box



facebook



Google™



Microsoft

@WalmartLabs

Corporate Linux Kernel Contributors

Company	Changes	Total
None	12,550	13.6%
Red Hat	9,483	10.2%
Intel	8,108	8.8%
Texas Instruments	3,814	4.1%
Linaro	3,791	4.1%
SUSE	3,212	3.5%
Unknown	3,032	3.3%
IBM	2,858	3.1%
Samsung	2,415	2.6%
Google	2,255	2.4%
Vision Engraving Systems	2,107	2.3%
Consultants	1,529	1.7%
Wolfson Microelectronics	1,516	1.6%
Oracle	1,248	1.3%
Broadcom	1,205	1.3%

Company	Changes	Total
NVidia	1,192	1.3%
Freescale	1,127	1.2%
Ingics Technology	1,075	1.2%
Renesas Electronics	1,010	1.1%
Qualcomm	965	1.0%
Cisco	871	0.9%
The Linux Foundation	840	0.9%
Academics	831	0.9%
AMD	820	0.9%
Inktank Storage	709	0.8%
NetApp	707	0.8%
LINBIT	705	0.8%
Fujitsu	694	0.7%
Parallels	684	0.7%
ARM	664	0.7%

```
(linux) $ git shortlog -se origin/master | grep microsoft
  1 Dexuan Cui <decui@microsoft.com>
135 Haiyang Zhang <haiyangz@microsoft.com>
  40 Hank Janssen <hjanssen@microsoft.com>
  791 K. Y. Srinivasan <kys@microsoft.com>
   14 KY Srinivasan <kys@microsoft.com>
    1 Mike Sterling <mike.sterling@microsoft.com>
```

libgit2

the business of
open source

why do businesses open source

they want to *use* open source

finding / attracting developers

working across fields (ruby, libgit2)

engage community in their other products



box



Square

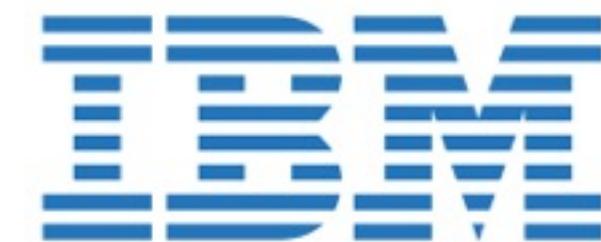


facebook



Linkedin

Google™



Dropbox



Microsoft

@WalmartLabs

the (possibly) certain future of
open source

corporate
open source



box



Square



facebook



Linkedin

Google™



Dropbox



Microsoft

@WalmartLabs

open source
workflow

proprietary
workflow

workflow

more remote work, autonomy

fewer cost of living issues (like SF)

fewer meetings, less email - more things with URLs

the death of
copyleft

Would you buy a car with the
hood welded shut?





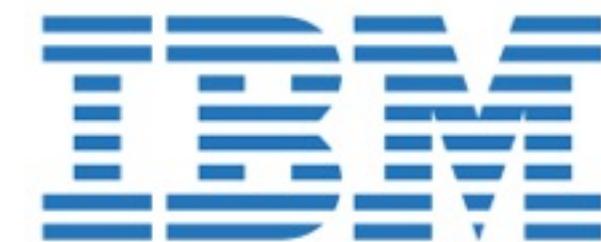
box



facebook



Google™



Microsoft

@WalmartLabs

What does copyleft do?



freedom from fear

what do we want from
open source

what do we want from open source

freedom from fear

the ability to improve and learn from cutting edge software

collaborate with people in other companies on commodity software

ability to deeply engage with users

what do we mean by
open source

what do we mean by **open source**

the availability of the source code?

the right to use it for anything?

the right to contribute back and improve it?

freedom
isn't enough

community source

community source

clear and permissible license and CLA

clear contributing guidelines (use github flow unless good reason)

be responsive, help people contribute

prepared to give up the project if you can't do this

“free as in
speech”

“free as in
we’re listening”

what can
you do?

"We see how politics, instead of being a dirty word, could be what it meant in the original Greek: the engagement of all citizens in the decisions that affect their lives."

Andrew Rasiej, PDF Opening Remarks, 2005

thank you

@chacon

