This is the preprint of an article accepted for publication by *New Media and Society*. It does not feature elements present in the final version [https://doi.org/10.1177/14614448221090474] such as copy-editing, formatting and layout.

Co-producing industrial public goods on GitHub: Selective firm cooperation, volunteer-employee labour and participation inequality

Mathieu O'Neil[1]
News and Media Research Centre
University of Canberra, Australia

Xiaolan Cai[2]
News and Media Research Centre
University of Canberra, Australia

Laure Muselli[3]
i3, Télécom Paris, Institut Polytechnique de Paris, France

Stefano Zacchiroli[4]
LTCI, Télécom Paris, Institut Polytechnique de Paris, France

[1] Corresponding author: mathieu.oneil@canberra.edu.au
[2] u3226998@uni.canberra.edu.au
[3] laure.muselli@telecom-paris.fr
[4] stefano.zacchiroli@telecom-paris.fr

Summary

The global economy's digital infrastructure is based on free and open source software. To analyse how firms indirectly collaborate via employee contributions to developer-run projects, we propose a formal definition of 'industrial public goods' – inter-firm cooperation, volunteer and paid labor overlap, and participation inequality. We confirm its empirical robustness by collecting networks of commits made by firm employees to active GitHub software repositories. Despite paid workers making more contributions, volunteers play a significant role. We find which firms contribute most, which projects benefit from firm investments, and identify distinct 'contribution territories' since the two central firms never co-contribute to top-20 repositories. We highlight the challenge posed by 'Big Tech' to the non-rival status of industrial public goods thanks to cloud-based systems which resist sharing, and suggest there may be 'contribution deserts' neglected by large IT firms, despite their importance for the open source ecosystem's sustainability and diversity.

1. Introduction

Free and open source software (FOSS) created in self-governed online projects is ubiquitous: a survey of 1,200 Information Technology (IT) professionals found that 92% of applications contained open source libraries (Szulik, 2018). IT firms are paying developers to produce FOSS, though significant areas of the FOSS universe retain a hobbyist community character, relying on volunteer labor. This workforce comprising employees and volunteers and these hybrid commercial and communal social worlds and value systems defy traditional categorisations. How can we define the production of open source software, which is at the heart of contemporary digital society? The concept of 'industrial public goods' has been advanced as a solution. Its origins lie with economists Paul M. Romer and Zvi Griliches' (1993) proposal to institute 'self-organizing industry investment boards' to fund non-rival goods (a resource is deemed non-rival if, for any level of production, the cost of providing it to an additional user is zero). This proposal was intended to address the classic collective action problem posed by non-rival goods: some actors may decide to 'free ride' and benefit from others producing non-rival goods, without contributing anything in return. Since then IT firms have indeed self-organized to participate in the production of non-rival FOSS. French sociologists (Jullien & Zimmermann, 2013; Alcaras, 2020) used Romer and Griliches' (1993) original proposal of inter-firm cooperation to analyse how firms actually co-produce free and open source software. This latter definition of 'industrial public goods' was based on traits such as developers appropriating the means of production and controlling technical expertise, the existence of an overlap between paid and volunteer work, and the value of non-commercial production being recognised. Building on this work, we suggest that firms now cooperate both directly (e.g., via firm-sponsored entities such as the Linux Foundation) and indirectly (via employee contributions to developer-run projects). While the concept of 'industrial public goods' seems to fit this indirect co-production of open source software, it has yet to be formalized and empirically validated. In this article, we map contributions to GitHub open source repositories as the means to explore the fitness of this concept. We suggest that industrial public goods satisfy three criteria: firms indirectly cooperate to produce non-rival goods; contributions are highly unequal; and there exists an overlap between paid and volunteer work. We first seek a clearer understanding of how much firms are engaging with free and open source software. Our first research question is:

RQ1 Which firms contribute the most to FOSS projects, and which projects most benefit from firm investments?

We present an empirical study of the network of connections between firms and projects within a subset of highly active repositories in the popular GitHub code sharing platform. We collect the email addresses of the committers (or contributors: a 'commit' is a source code change) to a selection of GitHub repositories and use corporate email address domains (e.g., @microsoft.com) as a proxy for paid employment by firms. In particular, we focus on the question of collaboration between nominally competing firms, and our second research question is:

RQ2 To what extent do firms collaborate?

Analysing contribution patterns evokes another trait, noted since the origins of FOSS (Hill et al., 1992): contributions to FOSS are often found to be unequal, that is to say concentrated in the hands of a few hyper-productive participants. We posit that the production of industrial public goods will be similarly unbalanced, and our third research question is therefore:

RQ3 To what extent are contribution patterns unequal?

Finally, we address the relationship between paid and unpaid work during open source software coproduction by focusing on the amount and timing of paid and volunteer contributions, and our fourth research question is:

RQ4 To what extent do individual firm employee and volunteer contribution patterns overlap?

We analyse our GitHub dataset using content- and network-analytical methodologies and find that our definition of 'industrial public goods' is validated by our data. We uncover firm industrial strategies in the shape of 'free riding' and of 'contribution territories' whereby contributions employees of the two most central firms in our network never co-occur in the top-20 repositories.

2. Background

2.1. Evolution of firm engagement in FOSS

The benefits of FOSS are now orthodoxic in the IT industry. How did FOSS come to occupy this position, given it was originally perceived as a major threat by firms because of licenses such as the General Public Licence or GPL ('copyleft'), which contradicted traditional understandings of intellectual property? This section summarises the evolution of IT firms in this respect. Microsoft is particularly notable, as its early antagonism to free software was well-known: in 1998 a leaked Microsoft report, the ‘Halloween memo’ recommended the following strategy: ‘OSS projects have been able to gain a foothold in many server applications because of the wide utility of highly commoditized, simple protocols. By extending these protocols and developing new protocols, we can deny OSS projects entry into the market.’ In other words, the memo advocated adopting open standards and protocols and then modifying them - possibly proposing new features, but mostly making Microsoft versions incompatible with free versions (Rosenberg, 1998), a model variously referred to as ‘embrace, extend, extinguish’ or ‘copy and corrupt.’ Twenty years later, in 2018, Microsoft joined the Open Innovation Network, a ‘defensive patent pool and community of patent non-aggression’ aiming to protect Linux, signifying that Microsoft renounced extracting value from violations of their patents that may occur in open source products.[5] That same year Microsoft also bought GitHub for US$7.5b. Still in 2018, IBM acquired Red Hat for US$34m, and Google adopted Debian as the base for its internal operating system. IBM’s engagement with FOSS in the early 2000s inaugurated a larger corporate movement to translate FOSS principles into a neoliberal language of market agility, consumer choice, and an ‘improved bottom line’ (Coleman & Hill, 2004). The adoption by IT firms of open source licenses subsequently enabled the full integration of FOSS into the IT industry. This constituted an evolution between what Sebastien Broca (2018) defined as Microsoft’s ‘informational capitalism’ – i.e., the firm seeks to protect the value of its closed proprietary systems – to Google’s more flexible ‘digital capitalism’ – i.e., the firm integrates the commons into its business model and prioritises mobile and cloud business models using big data and artificial intelligence (Rikap & Lundvall, 2020; see also Birkinbine, 2020; Lund & Zukerfeld, 2020). Another integrating factor was the creation of online platforms such as GitHub (2005) and Stack Overflow (2008). Stack Overflow has become a key resource for mentoring and advice whilst GitHub facilitates large-scale collaborative development. Finally open source’s adoption by firms was facilitated by the emergence of key mediating entities or ‘boundary organisations’ (O’Mahony and Bechky, 2008), chief amongst which is the Linux Foundation. This Foundation, originally created in 2000 to employ Linux’s creator Linus Torvalds in order to prevent him from being attached to a single firm which would thereby

[5] https://www.openinventionnetwork.com/

gain inordinate influence, has become a key hub of software interoperability and firm-project coproduction; its growth has been described as ‘extraordinary’ (Biddle, 2019).

2.2. Paying for FOSS: Research and mapping

Researchers have investigated developers being paid to produce free and open source software in various guises. The question was initially addressed via developer motivation, which can be intrinsic, such as self-fulfilment, or extrinsic, such as financial gain (Alexy & Leitner 2011; Krishnamurthy et al., 2014). Krishnamurthy (2006) noted that FOSS had a ‘diversity of project structures, diverse employment arrangements, co-existence of corporations and communities and co-existence of the creative and commercial elements’ (2006: 25). By co-producing code with communal projects, firms are engaging in external collaborations with an ‘unknown workforce’ without the usual guarantees – in terms of delivery dates for example – provided by contracts (Ågerfalk and Fitzgerald, 2008)

The employment of FOSS developers has also been addressed in terms of its impact on projects. For Mansell and Berdou (2010) workers being paid by firms to contribute to the commons does not affect the ‘cooperative spirit’ of communal projects. O’Neil et al. (2020) examined the institutional logics of firms and projects, arguing that the integration of the commercial logic of firms with the communal logic of the Debian project required rhetorical legitimation: firm employees invoked developer self-fulfilment, irrespective of whether the work was paid by a firm or not. Butler et al. (2018) examined the type of contribution made by firm employees to communal projects, distinguishing between bug reporting, feature requests, support (seeking or providing help) and other activities such as documentation or governance.

In terms of size, Mansell and Berdou’s (2010) examination of GNOME found that 75% of ‘core’ modules were maintained by developers affiliated to firms. Riehle et al. (2014) found that 23% of authors working on the Linux kernel were paid for their work, as were between 10 and 20% of developers in GNOME, Netbeans IDE, KDE and KVM. These authors based their analysis of the proportion of paid and volunteer work in projects on the time of commits to Linux - were these made during ‘business hours’ or not? - and estimated that approximately 50% were paid. As concerns the evolution of this relationship, they suggested that the number of paid developers remains stable overall as work during leisure time is higher in new projects, and work during office hours higher in more established projects. Claes et al. (2018) criticised

Rielhe et al.'s (2014) approach, arguing that considering any commit made on a weekday between 9AM and 5PM as paid work, and work outside this timeframe as voluntary work, 'completely ignores unemployment, flexible working hours and overwork' (Ibid., p. 437). Whilst this critique has validity, we contend that the time of commits is a valuable heuristic which should not be discarded. That being said, a limitation of Rielhe et al.'s (2014) analysis of paid labor is that they do not examine in detail which firms are contributing to FOSS development by paying employees. As noted by Eghbal (2016): 'With better metrics, we could describe the economic impact of digital infrastructure (…) Right now, it is impossible to say who is using an open source project unless that person or company discloses their usage' (129). Though we propose to explore contributions by firms to open source, whilst Eghbal is referring to firm use of open source (which can occur with zero contributions, aka 'free riding'), both these approaches agree that firm involvement in open source is under-documented. Indeed the literature on industrial public goods also calls for concrete, quantitative evidence about the scale of firm involvement (Alcaras, 2020). We address this question by collecting commit data automatically from GitHub, but instead of starting with the time of commits, we first analyse the professional email addresses of committers. We now present how our GitHub dataset was built.

3. Dataset and methodology: **GitHub commits by firm employees**

Created in 2008, GitHub is a code hosting platform based on the Git version control system which makes participation levels in FOSS projects highly visible. It has grown to become the most popular FOSS collaborative development platform, hosting more than 100 million repositories, becoming the 'cultural epicenter' for open source's explosive growth (Eghbal, 2016: 50).

Between early 2015 and mid-2019, the period of our study, approximately 10 billion commits were published on GitHub.[6] However these numbers are deceptive. Kalliamvakou et al. (2014) found that despite being nominally oriented towards 'social coding,' the number of committers per GitHub repository is highly skewed: 72% of repositories have one committer, 91% have 2 or less, and 95% have 3 or less. They conclude that 'perhaps the biggest threat to validity to any study that uses GitHub data indiscriminately is the bias towards personal use. While many repositories are being actively developed on GitHub, most of them are simply personal, inactive repositories' (Ibid., p. 100).

[6] Based on the numbers provided at https://octoverse.github.com/

For this reason, we targeted a restricted number of highly active repositories. By 'committers' we understand the original contributors ('commit authors' in Git terminology), not the maintainers who review contributions ('commit committers' in Git terminology). Our GitHub commits dataset was constituted between 1 January and 31 May 2019. 135 GitHub repositories were selected in three ways: (a) top 42 repositories by number of committers, commits and forks listed in the Open Source section of GitHub's annual 'Octoverse Report' (1,934,848 commits, Nov 2015-Oct 2018);[7] (b) top 45 repositories from the Technology section of the Stack Overflow (a popular Q&A site for software developers) Developer Survey (596,538 commits, Jan 2016-Dec 2018);[8] (c) finally, 48 most active repositories self-identifying as Open Source via GitHub tags by number of commits and stars were collected in May 2019 by the authors (293,304 commits, from their date of creation to 30.04.2019).[9]

We collected commit information from these 135 repositories via GitHub's REST API v3 with R, using packages such as *httr, httpuv,* and *jsonlite* in May 2019. In total, 113,614 committers with 26,459 unique email domains contributed 2,824,690 commits to these 135 repositories between 01.01.2015 and 31.05.2019. Using a professional email address whilst contributing to a FOSS project signals that a developer is being paid to produce FOSS (O'Neil et al., 2020). Committers with an email address such as @google.com or @microsoft.com were classified as working for a firm.[10] 'Foundation' comprises addresses such as @linux.org and @php.net. 'No affiliation' comprises email address domains such as @gmail.com and @users.noreply.github.com. 'Researchers' comprises @***.edu and similar addresses. 'Individual developer' comprises addresses related to committers' personal identities such as @***.nl.

GitHub accounts allow users to commit with different email addresses. We analysed this feature by using 'group by' (author and author email domain) and 'count' in the R package 'dplyr' functions to identify accounts using at least one firm email address and one non-firm email address to commit to the selected 135 repositories, regardless of whether they used Foundation and/or researcher and/or no-affiliation and/or individual email addresses.

4. Findings

[7] https://octoverse.github.com/

[8] https://insights.stackoverflow.com/survey/

[9] GitHub participants use 'stars' to bookmark interesting repositories for later reference (Lima et al., 2014).

[10] The full list of firm email address domains will be made available online. [Note to reviewers: see Appendices]

4.1. Which firms contribute the most to FOSS projects, and which projects most benefit from firm investments? (RQ1)

Our first metric for industrial public goods firm engagement in FOSS projects. Figure 1 provides a numerical count of the most active contributing firms and Table 2 a numerical count of the projects most-contributed to by firm employees. Microsoft was the lead firm contributor on GitHub during the 2015-2019 period (a strategy confirmed by their 2018 purchase of the platform).

[Figure 1 about here]

[Table 1 about here]

Table 1 presents a proportional breakdown of commits made by firm employees to repositories. For reasons of space, we only include the top 20. We note that Linux is by far the central node in this firm-project GitHub network. This is due to the historical importance of Linux, the first massively collaborative software project, and to the fact that a kernel is mandatory in both hardware machines and virtual machines in the cloud, making a high-quality open source kernel hard to avoid.

To determine whether firm contributions are significant, we collected commit Source lines of code (SLOC), a software metric used to measure the size of a computer program by counting the number of lines in the text of the program's source code. Table 2 provides detailed insights into the extent to which firms contribute to free and open source projects. Only SLOC for the top-20 projects with the largest proportion of commits made by firm employees were collected.

[Table 2 about here]

Overall, SLOCs contributed by firm employees are significant in volume, signifying that employees are not just gatekeepers harnessing volunteer effort: they do the actual work. In terms of specific projects, we note that Linux has significant non-firm contributions and that Kubernetes, which produces a popular open source solution for managing cloud infrastructure, has relatively few commits from firm employees (52%) but these contributions are more massive (82% of SLOC): volunteers contribute frequently, but to marginal sections of the product. Mono, though currently controlled by Microsoft, has strong community involvement, likely stemming from the project's origins as a community-driven implementation of Microsoft's .NET technology. Pytorch, a scientific project, has relatively few firm employee contributions, just like Tensorflow (Google's machine learning framework). Contributions from researchers in these projects may be standing in for firm employee contributions; they could form the object of dedicated investigation in future work.

Apart from NixOS (2), Homebrew (3), pytorch (18), and Apache Spark (19), which are community-managed projects, the top projects are either entirely developed by firms (via their employees) or managed by industrial consortia formed by firms with common market interests, whose boards are controlled by firms (e.g., by requiring high fee payments to join consortium boards of directors, the so called 'pay-to-play' governance model). With rare exceptions, in contrast to community-driven projects, technical governance in those consortia-managed projects is not in the hands of developers but derive top-down from industry interests.

That the technical development of single-firm projects is directly controlled by the strategic interest of the controlling firm can be seen in Table 1 in the case of spring-projects (Pivotal, 89%), or vscode and dotnet (Microsoft, 72% and 71%). Diversity in terms of *who-contributes-how-much* is a factor to be taken into account when evaluating to what extent a project is controlled by a single firm, a group of firms, or a much larger group of stakeholders - as is the case for the Linux Kernel. Linux has by far the highest number of commits overall, 73% of which are authored by firm employees, and yet employees of the leading firm (Intel) only contribute 0.07% of commits (see Table 1).

In order to delve deeper into firm contribution patterns, we list in Table 3 the projects which the top 10 firms contribute to.

[Table 3 about here]

In general terms, firms providing software and/or cloud services contribute to repositories which they initiated, and firms producing hardware mainly contributed to Linux. For example, Microsoft has its own kernel (Windows) and, up to recently, did not use Linux much in-house, so has little technical interest in contributing to Linux – this is marginally less the case since the introduction of Microsoft's cloud product (Azure) which increasingly relies on Linux. Red Hat lies at the opposite end of the spectrum as its main activity consists in selling paid services to clients running Red Hat on servers (in most cases 'metal' servers, i.e., real hardware, instead of virtual servers). Hence Red Hat has an interest in making sure Linux, the kernel it uses, runs well on hardware; the main way to make sure Linux runs well is contributing to the Linux kernel, which explains this firm's high number of contributions. The quasi-entirety of Apple and Pivotal contributions go to single projects (swift/swift and spring/boot, respectively): 99% of the commits and SLOC their employees contributed went to these repositories. Despite adopting open source technologies early on, Apple is not helping to sustain top open source repositories. Table 3 enables us to understand which contributions can be said to correspond to a firm's industrial strategy (e.g., when commits and SLOCs number in the hundreds of thousands), and which are due to a developer's hobby-like activity (e.g., when commits and SLOCs number in the hundreds or dozens). We explore individual contribution patterns and firm cooperation in the following sections.

Before then, we need to consider another question: the role of automated agents or 'bots.' If firm contributions are being made by bots, this would not change the technical involvement of firms in projects, but it would reduce their financial engagement, and the significance of human employees would need to be downplayed. We searched for well-known bots used in software development (Erlenhov et al., 2019) in our GitHub dataset, and found commits made by two bots ('dependabot' and 'greenkeeper') which were used by non-firm-employees. These two bots submitted 237 commits representing 2,623 SLOC in total (amounting for 0.002% of total SLOC contributed by non-firm-employees). We searched for users with email addresses containing 'bot' and 'robot,' and, following Dey et al. (2020) we excluded false positives such as people's names (e.g., 'Abbot') and company names which contain 'bot' or 'robot.' This enabled us to identify 16 bots which generated 6,517,394 SLOC, representing 6.5% of SLOC submitted by firm-employees and 2.8% of total SLOC. In short, bots have a minimal impact on the present study.

4.2. To what extent do firms collaborate? (RQ2)

We pursue our investigation of the fitness of industrial public goods as an explanatory concept by examining how indirect firm cooperation (via contributions to developer-run projects) operates in practice. We list in Table 4 the top five firm contributors for the top twenty most active projects in our sample.

[Table 4 about here]

Microsoft and Google, the most central entities in our firm network (see below) are ubiquitous, being present in 13 of the top 20 repositories, often as the leading contributors. The contributions of Microsoft and Google employees co-occur with significant contributions by employees of other firms (e.g. Intel, Red Hat, IBM). However in the seven repositories where Microsoft is the dominant contributor, Google employee contributions are non-existent or close to zero, and the reverse is true of the six repositories where Google dominates: Microsoft employee contributions are always below 0.5%. This shows that these firms have clearly distinct 'contribution territories.'

As for the seven repositories with low contributions from both Microsoft and Google: this absence can be attributed to the presence of a third party competitor in the case of swift (Apple), pytorch (Facebook) and servo (Mozilla), or to the repository's niche character: nixpkgs is a GNU/Linux software distribution mainly popular in scientific contexts; godotengine/godot is a FOSS game engine, and neither Google nor Microsoft are heavily invested in the video game market (in any case players in that space have their own non-FOSS game engines); finally Elasticsearch is a software product used to index and search large amounts of documents, and Google have their own search solutions.

To further investigate firm cooperation patterns, we conducted network analyses of connections between firms and projects. Social network analysis quantitatively measures the behaviour of actors in networks as collections of nodes connected by ties. It also measures the degree to which networks are centralised and nodes are central. Two key measures of centrality are degree centrality, which measures a node's number of connections, and betweenness centrality, which measures how much a node is likely to be in a favoured position (to the extent that the node falls on the shortest path between other pairs of nodes in the network).

[Table 5 about here]

Table 5 compares the two forms of centrality for the top 20 firms. Despite contributing far less in terms of volume of commits than Microsoft employees (see Figure 1), Google employees occupy the top position. This can be explained by the distribution of Google employee commits, which is more evenly spread out than that of Microsoft employees, as can be seen in Table 3. Despite contributing less overall, Google employees are thus co-contributing with a large number of other firm employees.

Another way to map connections between firms is to measure top dyads or pairwise connections between firms. Table 6 shows the top-20 connections between firms committing to common repositories. As in Table 4, Google and Microsoft are never paired. Red Hat and Intel, two other top firm contributors, collaborate with each other and with other leading firms such as Google, AMD and Huawei, but not with Microsoft.

[Table 6 about here]

4.3. To what extent are contribution patterns unequal? (RQ3)

Our second metric is contribution inequality. To address it we map the connections between firms and repositories in the whole network. Figure 2 is a two-mode network (comprising two different types of actors: firms and repositories) which are connected according to the firms' employees contributions to repositories. For the reason outlined above, Google is more prominent than Microsoft. Similarly Linux not only receives the highest volume of contributions (see Table 1), but also has high betweenness centrality, signifying that it is receiving contributions from a great diversity of sources (as seen in Table 2). Figure 2 points to the existence of strong inequalities in terms of contributions.

[Figure 2 about here]

Table 7 provides a summary of the properties of the entire network. The number of ties directed towards projects (indegree) and ties issuing from firms (outdegree) are distributed into four quartiles, each containing an equal number of nodes (6,648). The firms and projects in the highest quartiles obtain by far the greatest number of connections: a minority of projects is attracting the overwhelming majority of commits and a minority of firms are overwhelmingly committing. Examining the GHTorrent dataset (a larger collection of GitHub repositories, users and events), Kalliamvakou et al (2014) also found that the most active 2.5% of projects accounted for the same number of commits as the remaining 97.5%.

[Table 7 about here]

We next analysed top email domains. The top-100 email domains with the greatest number of commits to the 135 repositories were classified in five categories (Table 8). There are 41 firm domains among the top-100, corresponding to 11,538 GitHub accounts, which are responsible for more than 1m commits. The other significant block of commits, of a similar size, is authored by unaffiliated email domains, corresponding to a much higher number of GitHub accounts (68,913): firm employees have a higher contribution rate. The single research domain is CERN.

[Table 8 about here]

4.4. To what extent do individual firm employee and volunteer contribution patterns overlap? (RQ4)

One of the means we use to address our third metric (overlap between paid and unpaid work) is to focus on a subset of committers who use both firm and non-firm emails. As we will see, these findings resonate with our contribution inequality metric. But to start off with, we analyse the time of commits of individual contribution. Figure 2 conforms to previous literature (Rielhe et al., 2014) which held that firm employee contributions dip during the weekend, whereas volunteer contributions remain constant, showing that our classification of employee and volunteer contributions is robust.[11]

[Figure 2 about here]

[11] We thank a reviewer for this point.

[Figure 3 about here]

Figure 3 shows how SLOC relates to time of commit. Just like the average number of commits (Figure 2), the median SLOC number dips for employees during the weekend, whereas it remains constant for volunteers.

We next analyzed the relationship between volunteer and firm email contributions. Each account on GitHub can be linked to multiple email addresses. Among the 113,614 developer accounts who contributed to the selected repositories on GitHub from 01.01.2015 to 30.04.2019, there were 11,943 (10.5%) who used more than one email address. Of those, 3,279 accounts (2.9% of total) were identified as using both a firm email address and a non-firm email address. These 3,279 accounts contributed 614,746 commits or 21.8% of the total number. In terms of source lines of code (SLOC), these 3,279 developers, or less than 3% of the total, contributed 24.4% of all SLOC: a striking instance of contribution inequality. More than half (369,529) the commits contributed by the 3,279 individuals who both used firm and non-firm email addresses were made using a firm email address, whilst 233,429 were made using a non-firm or personal email address, showing that volunteer contributions are significant (see Table 9).

[Table 9 about here]

[Figure 5 about here]

Figure 5 shows that firm involvement in FOSS is rising: while non-firm email address use remains constant, there is a clear increase in the numbers of firm email addresses between 2015-2016 and 2017-2019. Firm email addresses are more numerous than non-firm addresses during the work week, and there is a sharp decrease of firm email account use on the weekend. Non-firm email addresses use also decreases on the weekend, but much less than that of firm email addresses.

[Figure 6 about here]

Similarly Figure 6 shows that for the 3,279 developers using both firm and personal email addresses SLOC have dramatically increased, but only for the firm email addresses used by these dual-email address developers (non-firm levels remain constant or decrease). Finally, we classified the relationship between firm and non-firm email use amongst the 3,279 dual-email address type developers. We defined as 'occasional' developers demonstrating less than 10% overlap between firm / non-firm email account use; as 'parallel' developers demonstrating constant and frequent overlap between firm and non-firm email account use; and as 'sequential' developers who changed from non-overlapping firm or non-firm email account use to the opposite non-overlapping firm or non-firm email account use. Table 10 lists these developer profiles. Among the 862 'sequential' profiles, 494 moved from non-firm to firm email addresses, whilst 347 moved from firm to non-firm email addresses, and the rest (18) experienced more than one main change (staying with the same email address type more than 180 days).

[Table 10 about here]

5. Discussion

5.1. Firms and Industrial Public Goods

The development of new media and digital technology have disrupted traditional understandings of business by individualising what used to be solely organisational industrial production processes. In the same way that individual bloggers and social media 'influencers' have challenged the monopoly of news media and marketing firms, individual computer engineers or hackers have to a large extent appropriated their means of production and of recognising technical expertise. This has led to the emergence of key non-firm entities such as FOSS software projects, the recognition of the value of these projects' output by firms, and a blurring of the once airtight barrier between paid and nonpaid work. We have mapped contributions to a sample of active open source repositories on GitHub and shown that dominant players in the IT software industry contribute the most in terms of volume (Microsoft) or diversity of recipients (Google), with other major industrial actors (e.g. Apple, IBM) contributing significantly less (RQ1).[12] Following Alcaras (2020),

[12] In 2019 the ten largest technology companies by revenue, according to the Fortune Global 500 list of technology sector (i.e., hardware and software) companies were: Apple, Samsung Electronics, Foxconn, Alphabet, Microsoft, Huawei, Dell Technologies, Hitachi, IBM, and Sony. See https://en.wikipedia.org/wiki/List_of_largest_technology_companies_by_revenue#2019_list

we defined FOSS as non-rival ‘industrial public goods’ and proposed a formal definition including indirect inter-firm cooperation, contribution inequality and the overlap between paid and volunteer work. Our first metric (indirect inter-firm cooperation) was confirmed as we found evidence of employees from multiple firms contributing to GitHub repositories (RQ2). However this cooperation was highly selective, with employees of the two most central firms, Google and Microsoft, never contributing to repositories where employees of their counterpart were the top contributors.

Large firms derive two main benefits from releasing some technology as open source: it allows these firms to showcase their technological credentials, helping them to attract talented developers in the highly competitive IT hiring market, and it also enables firms to benefit from the contributions of unpaid developers, without giving up their control of the technology. To that end, firms control who can commit to a given repository and who cannot (typically, only firm employees can commit, whereas others need to obtain approval via pull requests). This partly explains why firms remain the main contributor to a repository of a technology they initially released. The technology is therefore ‘open source’ in terms of the code license, but is not developed following an open or shared governance model whereby developers from different firms collectively decide its future direction (the Linux kernel is an obvious exception, as it features contributions from a very diverse group of firms). The mutualization of the development costs of non-rival industrial public goods can be defined as ‘coopetition’ in which firms both compete and collaborate with each other (Teixeira et al. 2015; Nguyen-Duc et al. 2019).

Contributions to FOSS being concentrated in the hands of a few hyper-productive participants has been noted since the origins of FOSS (Hill et al., 1992; Kuk, 2006), with more recent examples confirming this observation (Chełkowski et al., 2016). Whilst examining OS development patterns some researchers found evidence for the so-called ‘Pareto principle’ whereby less than 20% of users make 80% or more contributions (Goeminne & Mens, 2011; Akond Rahman et al., 2018); others found less support for this principle (see Yamashita et al., 2015; Gasparini et al., 2020). Contribution patterns in our network were highly unequal, both when considered in aggregate, by firm domains (see Tables 7 and 8) with a few key organisational players massively contributing, as well as in terms of individual developers (see Table 9). This confirmed our second metric (RQ3).

In terms of our final metric for the existence of industrial public goods, the connection between volunteer and paid work (RQ4), we found that although contributions made by non-firm email addresses are less numerous than those made using firm email addresses, they are still very significant. Developers being paid by a firm to produce code may be reluctant to use their professional email address when contributing to a project which is less directly relevant to their firm. This could also explain the very low number of contributions made by (for example) large IT firm employees to repositories which their firm does not support. We found that firm email contributions dip significantly during the weekend, in contrast to non-firm emails. We can only speculate as to why this is occurring: this could be understood as a manifestation of the negative connotations of working on a paid job during the weekend, implying one does not have a good work-life balance, in contrast to working for intrinsic, or non-monetary reasons as indicated by using a non-firm email. There is evidence of continuity between paid and volunteer work as some developers use predominantly one type of email address before switching to the other, with a slight majority of changes being from non-firm to firm email addresses. Clearer results concern the population of developers who commit using both firm and non-firm email accounts: firm email addresses are associated with significant increases in volume of commits and SLOCs.

We acknowledge that the highly-active nature of the repositories in our sample is not insignificant, and that focusing on a less active sample might produce different results. But for the time being, we interpret these results as supporting the definition of industrial public goods we have suggested: (a) firm cooperation, (b) employee and volunteer labor overlap, and (c) contribution inequality. Apart from its indirect quality, our analyses show that informality is also a feature of the production of industrial public goods: though a minority of contributors commit, being able to rely on a crowd of micro-contributors whose motivations are not monetary is key for this new kind of non-rival resource. The intermingling of volunteer and paid labor on GitHub has been examined previously (Rielhe et al. 2014; Claes et al. 2018), but we provided empirical evidence of its temporal evolution both in terms of the growing involvement of employees and of the interplay between volunteer and paid status. Finally our detailed network and content of employee contributions enable us to distinguish between hobby-type and professional contributions by firm employees. This represents a pioneering insight into the industrial strategies of firms whose employees' participation is highly selective, demonstrating clear 'contribution territories.' Selectively occupying territories enables firm employees to co-produce industrial public goods with the employees of ‘co-opeting’ firms and with unpaid volunteers.

5.2. Threats to openness

Public goods, whether industrial or not, are meant to be open: FOSS developers agree to renounce to their exclusive property rights over the resource they have created (Benkler, 2013). FOSS intellectual property (IP) licenses such as the GPL created a legal environment in which motivated contributors could entrust their work with individuals with whom they had no prior personal contact (Lee & Cole, 2003). In other words volunteer FOSS work was primarily done for the benefit of a community or project, and developers contributing without being paid challenged traditional understandings of work as linked to a firm. Whether this relationship exploited individual developers has been the object of much debate.[13]

It seems less controversial to suggest that increased firm involvement is likely to influence project decisions of all kinds, from technical to licensing ones. And indeed the open status of industrial public goods is now at risk. To understand why this is the case, it is necessary to briefly review the current IT industry landscape. The near-ubiquity of FOSS in a growing range of applications means all firms today, not just IT firms, are benefiting from the free labor of FOSS project contributors who are not their employees. With a few exceptions – such as the automotive sector for example – most non-IT firms which use FOSS components in ‘products and services’ (Butler et al., 2019) are reaping the benefits of FOSS without contributing in return. Foley (2013) calls this capture of outsourced intellectual rents ‘surplus-value appropriation.’ Non-IT firms are purchasing data storage and analysis services, based on FOSS, from ‘Big

[13] Does the commercial-communal coproduction of FOSS mean firms are unfairly benefiting from unpaid labor? Eghbal concluded her report on FOSS by asserting that ‘fundamentally, digital infrastructure has a free rider problem’ (2016: 106). This tension had been noted since the beginning of FOSS development, when firms were described as harvesting the altruism of volunteer developers (Haruvy et al., 2003), resulting in a relationship between altruistic individuals and selfish firms (Rossi & Bonaccorsi, 2005). Yet it could also be argued that far from being adversely impacted by volunteering their time, individuals benefit by giving away code, as others will then contribute also. The potential for this misinterpretation was identified by Chopra and Dexter (2007) when they discussed Terranova’s (2001) famous analysis of how users freely contribute online: ‘Terranova’s critique only points to the existence of labor uncompensated by traditional means such as money exchange: it trades on an old confusion by conflating “financially uncompensated” with “free”.’ Indeed, many developers derive enjoyment from contributing, and other unpaid benefits include lifelong learning, community support and validation. Further, the acquisition of a reusable set of skills leads to increased freedom in working wherever they choose (Eghbal, 2016). This is different from the traditional situation of proletarians who are dispossessed from the fruits of their labor, as here the exchange seems to be mutually profitable, even if the goods being swapped – economic profits for the firms, self-realisation and other advantages for volunteers – are different (Broca, 2018). However the question of developer exploitation must be left open for now, as determining whether an exploitative relationship exists would require empirical evidence of the concrete benefits involved: how much programmers benefit from participating in FOSS vs how much firms benefit, for example.

Tech' firms, sometimes referred to as 'GAFAM' (Google, Amazon, Facebook, Apple, and Microsoft). These large IT firms contribute to FOSS projects 'in ways that may not always be apparent from public sources, such as employing core project developers, making donations, and joining project steering committees in order to advance strategic interests' (Butler et al., 2019). Yet this firm support has entailed a considerable cost: several of the dominant 'GAFAM,' with Amazon Web Services as the lead offender, are using cloud computing and Software as a Service built with open source software to transform this scientific and technological knowledge, which is intended to be shared, into closed assets.[14] These assets are then sold as privately owned software business solutions to non-IT sectors such as the automotive, entertainment, and energy industries. Birkinbine's (2020) advancement of the notion of 'incorporation' over that of 'enclosure' (which typically refers to the imposition of higher excludability on the common resource) thus refers to a pre-SaaS era when value was being extracted without restricting the community's access to their collective goods: cloud-driven enclosures do represent a clear and present threat to openness.

Rikap and Lundvall (2020) position the involvement in FOSS of 'GAFAM' as typical of their widespread 'predatory' behaviour. Examples include releasing libraries to open source in order to set standards, enabling start-ups to integrate their applications into GAFAM platforms and – in case they are acquired – facilitating their integration into the platform. Predatory GAFAM behaviour also takes the form of the re-appropriation of co-authored research: 78,3% of Microsoft's 17,405 publications between 2014 and 2019 were co-authored with university researchers; during the same period Microsoft applied and was granted 76,109 patents, 0,2% of which were co-owned (Rikap & Lundvall, 2020).

Several interlocking reasons explain why the open source community has been unable to withstand this expropriation: the use of aggressive tactics by Big Tech companies to reject SaaS resistant licenses such as Affero; the fact many developers are themselves employed by Big Tech firms; the rhetorical strategies deployed by firms which aim to transform open source developers' values (Authors, 2021b); finally this is also due to the way FOSS developers view themselves and their activity. Part of the computer hacker philosophy is to view one's activity as a calling, not a profession, with Eghbal (2016) suggesting for

[14] In a 'traditional' mode, a software program is downloaded and executed by customers on their own hardware. In a SaaS mode, the program is never transferred onto the customers' machines, but is executed remotely on the provider's hardware and used online. With SaaS, service prevails over use: a subscription to a service is bought, rather than a user licensing agreement being accepted for software copied onto the user's computer. This creates a SaaS 'loophole' in the FOSS principle, as the service provider is no longer obliged to offer access to the code (Authors, 2021).

example that money is a ‘taboo’ topic in projects. The ethics of reciprocity, transparency and openness which animate FOSS have traditionally been understood in terms of the need to protect the ‘four freedoms’ (to access, copy, and modify software, as well as to release modified versions). Software should be considered as an asset upon which users have certain *rights* – not as the product of work that deserves monetary retribution, or as a now-fragile resource that needs to be cared for (Authors, 2021a). Will Big Tech firms’ increasing expropriation of collective resources and subversion of developer values generate a reaction from FOSS developers? At the least a wide-ranging public debate in the FOSS community would perhaps clarify whether the term ‘free and open source’ still has any meaning when its core principles of openness and sharing are being systematically flouted.

Conclusion

In order to analyse the impact of free and open source software in the digital economy, we proposed a definition of how industrial public goods are produced (encompassing metrics such as inter-firm indirect cooperation, participation inequality, and coexistence of volunteer and paid labor) and we provided empirical evidence of this definition’s validity. We mapped a firm-project FOSS coproduction network on GitHub, which was dominated by a minority of key players, such as Linux and Microsoft. We found diverse firm contributory models, including quasi-monopolies, indirect cooperation between firms, and extreme fragmentation in the case of Linux. Despite paid workers making the majority of contributions, unpaid workers played a significant role, indicating the informal nature of these hybrid work arrangements. Our mapping of exclusive firm ‘contribution territories’ led us to define indirect firm cooperation as ‘selective’ and raises the question of the existence of ‘contribution deserts’ – projects and code that are neglected by large IT firms and can only rely on volunteer labor, despite being important for the sustainability and diversity of the open source ecosystem. Further research would help determine the extent of these deserts and whether preserving these types of industrial public goods would warrant support from public authorities.

References

Ågerfalk, J., & Fitzgerald, B. (2008). Outsourcing to an unknown workforce: Exploring opensourcing as a global sourcing strategy. *MIS Quarterly*, 32(2), 385-409. https://doi.org/10.2307/25148845

Akond Rahman, A., Krishna, R., Sobran, A. & Menzies, T. (2018). We Don’t Need Another Hero?: The Impact of ‘Heroes’ on Software Development. ICSE-SEIP’18. IEEE, 245–253. https://doi.org/10.1145/3183519.3183549

Alcaras, G. (2020) Des biens industriels publics. Genèse de l’insertion des logiciels libres dans la Silicon Valley, Sociologie du travail, 62(3). https://doi.org/10.4000/sdt.33283

Alexy, O., & Leitner, M. (2011). A fistful of dollars: Are financial rewards a suitable management practice for distributed models of innovation? *European Management Review*, 8, 165-185. https://doi.org/10.1111/j.1740-4762.2011.01017.x

Benkler, Y. (2013). Practical anarchism: Peer mutualism, market power, and the fallible State. *Politics & Society*, *41*, 213–251.

Biddle, C.B., (2019). Linux Foundation is Eating the World. *Journal of Open Law, Technology & Society*, 11(1), 57-73. https://doi.org/105033/jolts.v11i1.137

Birkinbine, B. (2020). *Incorporating the digital commons. Corporate involvement in free and open source software*. University of Westminster Press.

Bonaccorsi, A., Rossi, C., 2004. Altruistic individuals, selfish firms? The structure of motivation in open source software. *First Monday*, 9(1). http://dx.doi.org/10.2139/ssrn.433620

Broca, S. (2018). Du modèle du logiciel libre au modèle productif des communs. Les licences pair à pair contre le free software? *Working Paper 9*, Projet EnCommuns.

Butler, S. et al. (2019). On company contributions to community open source software projects. *IEEE Transactions on Software Engineering*. https://doi.org/10.1109/TSE.2019.2919305

Chełkowski T., Gloor P., & Jemielniak D. (2016) Inequalities in open source software development: Analysis of contributor’s commits in Apache Software Foundation projects. *PLoS ONE* 11(4): e0152976.

Claes, M., Mäntylä, M., Kuutila, M., & Farooq, U. (2018). Towards identifying paid open source developers - A case study with Mozilla developers. *Proceedings of the 15th International Conference on Mining Software Repositories*. Sweden, 437-441. https://arxiv.org/abs/1804.02153v1

Coleman, B., & Hill, M. (2004). How Free Became Open and Everything Else under the Sun: Introduction. M*/C Journal*, 7(3).

Dey, T., et al. (2020). Detecting and characterizing bots that commit code. arXiv preprint arXiv:2003.03172

Eghbal, N. (2016) *Roads and bridges: The unseen labor behind our digital infrastructure*. Ford Foundation. https://www.fordfoundation.org/media/2976/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure.pdf

Erlenhov, L, et al. (2019). Current and future bots in software development. IEEE/ACM 1st International Workshop on Bots in Software Engineering (BotSE). IEEE, 2019. https://doi.org/10.1109/BotSE.2019.00009

Foley, Duncan K. (2013). Rethinking financial capitalism and the 'information' economy. *Review of Radical Political Economics*, 45(3): 257–268.

Gasparini, M., Clarisó, R., Brambilla, M., & Cabot, J. (2020). Participation inequality and the 90-9-1 principle in open source. *OpenSym 2020*. ACM, New York, NY, USA, Article 6, 1–7. https://doi.org/10.1145/3412569.3412582

Goeminne, M. & Mens, T. (2011). Evidence for the Pareto principle in open source software activity. 5th International Workshop on Software Quality and Maintainability. 74–82.

Hill, W., Hollan J., Wroblewski, D., McCandless T., 1992, Edit wear and read wear, Proceedings of the SIGCHI conference on Human factors in computing systems – CHI '92. 3–9.

Jullien, N., Zimmermann, J.-B. (2013) Le logiciel libre : un renouveau du modèle industriel coopératif de l'informatique. In Paloque-Berges, C., Masutti, C. (Eds), *Histoires et cultures du Libre. Des logiciels partagés aux licences échangées*, pp. 135-164. Paris : Framasoft.

Kalliamvakou, E., Gousios, G., Blincoe, K., Singer, L., German, D., & Damian, D. (2014), The promises and perils of mining GitHub. *Proceedings of the 11th Working Conference on Mining Software Repositories*. India, 92-101. https://doi.org/10.1145/2597073.2597074

Krishnamurthy, S. (2006). On the intrinsic and extrinsic motivation of open source developers. *Knowledge, Technology & Policy*, 18(4), 17–39. https://doi.org/10.1007/s12130-006-1002-x

Krishnamurthy, S., Ou, S., & Tripathi, A. (2014). Acceptance of monetary rewards in open source software development. *Research Policy*, 43(4), 632-644. https://doi.org/10.1016/j.respol.2013.10.007

Kuk, G. (2006). Strategic interaction and knowledge sharing in the KDE developer mailing list. *Management Science*, 52(7), 1031–1042.

Lima, A., Rossi, L., & Musolesi, M. (2014). Coding together at scale: GitHub as a collaborative social network. Eighth International AAAI Conference on Weblogs and Social Media. https://arxiv.org/abs/1407.2535

Lee, G. & Cole R. (2003). From a firm-based to a community-based model of knowledge creation. *Organization Science*, 14, 633-649

Lund, A., & Zukerfeld, M. (2020). *Corporate capitalism's use of openness*. Basingstoke: Palgrave.

Mansell, R., & Berdou, E. (2010). Political economy, the internet and FL/OSS development. In J. Hunsinger, M. Allen, & L. Klastrup, L. (Eds.), *International Handbook of Internet Research* (pp. 341–362). Springer. https://doi.org/10.1007/978-1-4020-9789-8_21

Maurel, L. (2019). Common labour rights and right to work in the commons. *Journal of Peer Production*, #13. http://peerproduction.net/issues/issue-13-open/news-from-nowhere/common- labor-rights-and-right-to-work-in-the-commons/

Nguyen-Duc, A., Cruzes, D., Terje, S., & Abrahamsson, P. (2019). Do software firms collaborate or compete? A model of coopetition in community-initiated OSS projects. *e-Informatica Software Engineering Journal*, 13(1), 37-62. https://doi.org/10.5277/e-Inf190102

O'Mahony, S., & Bechky, B.A. (2008). Boundary organizations: Enabling collaboration among unexpected allies. *Administrative Science Quarterly*, 53, 422-459. https://doi.org/10.2189/asqu.53.3.422

O'Neil, M., Muselli, L., Raissi, M., & Zacchiroli, S. (2020). 'Open Source has won and lost the war': Legitimising commercial-communal hybridization in a F/OSS project. *New Media & Society*. https://doi.org/10.1177/1461444820907022

Riehle, D., Riemer, P., Kolassa, C., & Schmidt, M. (2014). Paid vs. volunteer work in Open Source. *47th Hawaii International Conference on System Sciences* (HICSS), USA, 3286-3295. https://doi.org/10.1109/HICSS.2014.407

Rikap, C., & Lundvall, B-A. (2020). Big Tech, knowledge predation and the implications for development. *Innovation and Development*. https://doi.org/10.1080/2157930X.2020.1855825

Romer, P. & Griliches, Z. (1993). Implementing a national technology strategy with self-organizing investment boards, *Brookings Papers on Economic Activity, Microeconomics*, vol. 1993, n° 2, p. 345-399. DOI: 10.2307/2534742

Rosenberg, S. (1998, Nov. 5) Let's get this straight: Microsoft's Halloween scare, *Salon*. https://www.salon.com/1998/11/04/straight_39/

Szulik, K. (2018, April 12) Open source is everywhere: survey results part 1. *Tidelift*. https://blog.tidelift.com/open-source-is-everywhere-survey-results-part-1

Teixeira, J., Robles, G., & González-Barahona, J. M. (2015). Lessons learned from applying social network analysis on an industrial free/libre/open source software ecosystem. *Journal of Internet Services and Applications*, 6(1). https://doi.org/10.1186/s13174-015-0028-2

Yamashita, K., McIntosh, S., Kamei, Y., Hassan, A.E., & Ubayashi, N. (2015). Revisiting the applicability of the Pareto Principle to core development teams in open source software projects. IWPSE 15, Bergamo, Italy. ACM, 46–55. https://doi.org/10.1145/2804360.2804366

Table 1. Number and proportion of firm-employee commits in top-20 projects (by number of commits), 01.01.2015 to 30.04.2019

Project	Total	Firm-employee commits	Proportion firm-employee	Leading firm by commits	Percentage of leading firm commits
torvalds/Linux	340710	247864	0.73	Intel	0.07
NixOS/nixpkgs	125205	63042	0.50	LogicBlox	0.01
Homebrew/homebrew-core	108709	54352	0.50	Charcoal Design	0.001
apple/swift	67197	40351	0.60	Apple	0.56
kubernetes/kubernetes	74201	40041	0.54	Google	0.30
Microsoft/vscode	49418	37366	0.76	Microsoft	0.72
tensorflow/tensorflow	56656	29515	0.52	Google	0.46
dotnet/corefx	32884	25660	0.78	Microsoft	0.71
DefinitelyTyped	54801	17920	0.33	Microsoft	0.10
aspnet/AspNetCore	34946	16486	0.47	Microsoft	0.39
spring-projects/spring-boot	17855	16357	0.92	Pivotal	0.89
ansible/ansible	31544	16252	0.52	Red Hat	0.06
elastic/elasticsearch	33983	16051	0.47	Elastic	0.23
rust-lang/rust	57790	15897	0.28	Red hat	0.003
facebook/react-native	16908	12704	0.75	Facebook	0.63
moby/moby	24472	11743	0.48	Docker	0.16
home-assistant/home-assistant	18876	10756	0.57	Affolter Engineering	0.04
pytorch/pytorch	17717	10492	0.59	Facebook	0.49
apache/spark	15180	8584	0.57	Databricks	0.30
storybooks/storybook	18968	7297	0.38	Dependencies	0.12

Table 2. Percentage of commits by firm employees, number and percentage of SLOC by firm employees and non-firm employees in top-20 repositories (by highest proportion of firm commits), 01.01.2015 to 30.04.2019

Project	Percentage of commits by employees	SLOC by employees	SLOC by non-employees	Percentage of SLOC by employees
spring-projects/spring-boot	91.58%	1162062	137322	89.43%
Microsoft/vscode	75.38%	6213205	1048348	85.56%
react/native	75.05%	2452807	302470	89.02%
dotnet/corefx	70.07%	12976731	2207941	85.46%
apple/swift	64.28%	3336874	390357	89.53%
torvalds/linux	62.31%	7792547	3670596	67.98%
pytorch/pytorch	57.71%	2156521	1686623	56.11%
apache/spark	56.15%	1603642	889692	64.32%
moby/moby	55.69%	2234220	1878238	54.33%
kubernetes/kubernetes	53.32%	12105955	2580226	82.43%
tensorflow/tensorflow	53.06%	7631597	6967787	52.27%
aspnet/AspNetCore	48.00%	6728611	4615271	59.31%
storybooks/storybook	43.93%	1454999	1363101	51.63%
hashicorp/vagrant	42.96%	65825	92383	41.61%
elixir-lang/elixir	40.00%	182327	197479	48.01%
helm/charts	39.53%	140820	331542	29.81%
angular/angular	38.61%	1521341	2366043	39.14%
opencv/opencv	38.06%	1991537	1024937	66.02%
ansible/ansible	37.00%	1441504	2048983	41.30%
mono/mono	36.96%	1509989	2654873	36.26%

Table 3. Contribution from Top-10 firm contributors by number of commits to top-20 repositories (descending by number of commits)

Firm	Repository	Count	Percent. of firm commits	Percent. in repository	Count SLOC	Percent. of firm SLOC	Percentage in repository
Microsoft 438,220 commits 46,069,335 SLOC	MicrosoftDocs/azure-docs	342371	78.13%	80.33%	7371387	16.00%	84.02%
	Microsoft/vscode	32232	7.36%	72.91%	6103554	13.25%	84.04%
	dotnet/corefx	23284	5.31%	57.99%	12219477	26.52%	80.47%
	microsoft/TypeScript	21117	4.82%	73.07%	9680553	21.01%	78.19%
	aspnet/AspNetCore	14651	3.34%	39.45%	6011216	13.05%	52.99%
	DefinitelyTyped/ DefinitelyTyped	5510	1.26%	10.09%	2208651	4.79%	11.91%
	PowerShell/PowerShell	4037	0.92%	52.29%	459178	1.00%	45.71%
	dotnet/core	1772	0.40%	83.30%	545960	1.19%	81.23%
	mono/mono	1072	0.24%	5.81%	766648	1.66%	18.41%
	moby/moby	913	0.21%	5.55%	113752	0.25%	2.77%
	microsoft/vscode-docs	808	0.18%	0.13%	263593	0.57%	67.64%
	torvalds/Linux	715	0.16%	0.30%	21949	0.05%	0.19%
	kubernetes/Kubernetes	510	0.12%	0.76%	155720	0.34%	1.06%
	electron / electron	308	0.07%	2.83%	19051	0.04%	1.66%
	ansible/ansible	277	0.06%	0.89%	44721	0.10%	1.28%
	nodejs/node	202	0.05%	0.62%	32475	0.07%	0.04%
	facebook/react-native	119	0.03%	0.72%	7880	0.02%	0.00%
	tensorflow/tensorflow	118	0.03%	0.21%	5691	0.01%	0.29%
	Homebrew/homebrew-core	52	0.01%	0.05%	2027	0.00%	0.31%
	pytorch/pytorch	48	0.01%	0.24%	4074	0.01%	0.30%
	apache/hadoop	44	0.01%	0.00%	31.71	0.00%	0.06%
	JetBrains/intellij-community	187071	83.44%	84.05%	19863453	61.75%	84.83%
	JetBrains/kotlin	36004	16.06%	86.50%	11806284	36.70%	90.94%
	storybooks/storybook	1044	0.47%	5.58%	497705	1.55%	17.66%
	gradle/gradle	33	0.01%	0.00%	107	0.00%	0.00%
	Homebrew/homebrew-core	9	0.00%	0.01%	48	0.00%	0.00%
	mono/mono	8	0.00%	0.02%	49	0.00%	0.00%
	DefinitelyTyped/ DefinitelyTyped	5	0.00%	0.01%	312	0.00%	0.00%
	golang/go	4	0.00%	0.02%	15	0.00%	0.00%
	angular/angular	3	0.00%	0.04%	11	0.00%	0.00%
	npm/cli	2	0.00%	0.02%	67	0.00%	0.00%
	angular/angular.js	1	0.00%	0.02%	2	0.00%	0.00%
	ansible/ansible	1	0.00%	0.00%	2	0.00%	0.00%
JetBrains 126,485 commits 32,167,992 SLOC	hashicorp/vagrant	1	0.00%	0.02%	1	0.00%	0.00%
	meteor/meteor	1	0.00%	0.00%	27	0.00%	0.00%
	servo/servo	1	0.00%	0.00%	1	0.00%	0.00%
	spring-projects/spring-boot	1	0.00%	0.00%	16	0.00%	0.00%
Google 72,387 commits 26,496,745 SLOC	tensorflow/tensorflow	25675	35.47%	46.73%	6927173	26.14%	47.45%
	kubernetes/kubernetes	24044	33.22%	29.83%	8370118	31.59%	56.99%
	golang/go	4989	6.89%	19.35%	738619	2.79%	15.03%
	torvalds/linux	4586	6.34%	1.81%	112409	0.42%	0.98%
	flutter/flutter	3942	5.45%	28.24%	696954	2.63%	34.93%
	angular/angular	2194	3.03%	16.67%	687066	2.59%	17.67%

	istio/istio	1970	2.72%	15.28%	4579407	17.28%	59.91%
	tensorflow/models	1526	2.11%	43.79%	2820098	10.64%	62.08%
	angular/material	851	1.18%	7.72%	38323	0.14%	5.17%
	firebase/firebase-tools	483	0.67%	30.46%	180009	0.68%	62.92%
	flutter/plugins	445	0.61%	37.93%	325222	1.23%	51.52%
	nodejs/node	349	0.48%	1.43%	193560	0.73%	1.85%
	rust-lang/rust	253	0.35%	0.43%	22124	0.08%	0.53%
	remacs/remacs	237	0.33%	1.28%	3840	0.01%	0.46%
	ansible/ansible	236	0.33%	0.71%	107852	0.41%	3.09%
	angular/angular.js	224	0.31%	0.77%	956	0.00%	0.16%
	JetBrains/kotlin	216	0.30%	0.25%	337399	1.27%	2.60%
	DefinitelyTyped/ DefinitelyTyped	208	0.29%	0.37%	212055	0.80%	1.14%
	JetBrains/intellij-community	166	0.23%	0.14%	10656	0.04%	0.05%
	etcd-io/etcd	156	0.22%	1.40%	67054	0.25%	2.83%
Red Hat 54,788 commits 19,033,024 SLOC	openshift/origin	24467	44.66%	55.85%	10830172	56.90%	80.93%
	torvalds/linux	16150	29.48%	5.25%	585041	3.07%	5.10%
	kubernetes/kubernetes	8092	14.77%	9.90%	2188159	11.50%	14.90%
	ansible/ansible	2269	4.14%	6.35%	231634	1.22%	6.64%
	remacs/remacs	2155	3.93%	0.04%	25	0.00%	0.00%
	moby/moby	955	1.74%	1.31%	19935	0.10%	0.48%
	rust-lang/rust	170	0.31%	0.30%	10334	0.05%	0.75%
	istio/istio	93	0.17%	3.81%	64556	0.34%	0.84%
	Homebrew/homebrew-core	71	0.13%	0.01%	70	0.00%	0.01%
	apache/spark	49	0.09%	0.05%	145	0.00%	0.01%
	etcd-io/etcd	43	0.08%	0.41%	17680	0.09%	0.75%
	nodejs/node	40	0.07%	0.18%	1244	0.01%	0.01%
	jhipster/generator-jhipster	32	0.06%	0.11%	5624	0.03%	0.24%
	hashicorp/vagrant	26	0.05%	0.32%	194	0.00%	0.12%
	rails/rails	25	0.05%	0.09%	387	0.00%	0.06%
	NixOS/nixpkgs	20	0.04%	0.02%	98	0.00%	0.00%
	golang/go	13	0.02%	0.05%	597	0.00%	0.01%
	dotnet/corefx	11	0.02%	0.08%	1378	0.01%	0.01%
	spring-projects/spring-boot	10	0.02%	0.05%	2237	0.01%	0.17%
	django/django	10	0.02%	0.08%	169	0.00%	0.02%
Apple 37,520 commits 3,222,820 SLOC	apple/swift	37448	99.81%	55.87%	3214474	99.74%	86.24%
	apache/spark	44	0.12%	0.35%	5965	0.19%	0.24%
	jlord/patchwork	6	0.02%	0.00%	1	0.00%	0.00%
	torvalds/linux	5	0.01%	0.00%	16	0.00%	0.00%
	apache/hadoop	3	0.01%	0.02%	1356	0.04%	0.03%
	pingcap/tidb	3	0.01%	0.04%	18	0.00%	0.00%
	NixOS/nixpkgs	2	0.01%	0.01%	758	0.02%	0.02%
	DefinitelyTyped/ DefinitelyTyped	1	0.00%	0.00%	109	0.00%	0.00%
	facebook/react	1	0.00%	0.00%	72	0.00%	0.00%
	Homebrew/homebrew-core	1	0.00%	0.00%	4	0.00%	0.00%
	remacs/remacs	1	0.00%	0.00%	34	0.00%	0.00%
	nodejs/node	1	0.00%	0.00%	67	0.00%	0.00%
	opencv/opencv	1	0.00%	0.00%	27	0.00%	0.00%
	rails/rails	1	0.00%	0.00%	4	0.00%	0.00%
	rust-lang/rust	1	0.00%	0.00%	17	0.00%	0.00%

	WebAssembly/design	1	0.00%	0.07%	6	0.00%	0.02%
Intel 27,613 commits 2,003,070 SLOC	torvalds/linux	23445	84.91%	6.79%	787366	39.31%	6.87%
	opencv/opencv	1459	5.28%	12.38%	670911	33.49%	22.24%
	tensorflow/tensorflow	944	3.42%	1.46%	154268	7.70%	1.06%
	apache/spark	676	2.45%	2.22%	41284	2.06%	1.66%
	kubernetes/kubernetes	319	1.15%	0.46%	23549	1.18%	0.16%
	golang/go	243	0.88%	1.12%	111923	5.59%	2.28%
	apache/hadoop	206	0.75%	1.61%	2478	0.12%	0.04%
	nodejs/node	112	0.40%	0.34%	3029	0.15%	0.03%
	pytorch/pytorch	81	0.29%	0.39%	16105	0.80%	0.42%
	dotnet/corefx	44	0.16%	0.39%	176585	8.82%	1.16%
	ansible/ansible	18	0.07%	0.05%	1319	0.07%	0.04%
	WebAssembly/design	13	0.05%	0.77%	93	0.00%	0.37%
	moby/moby	7	0.02%	0.02%	352	0.02%	0.01%
	helm/charts	6	0.02%	0.06%	80	0.00%	0.02%
	tensorflow/models	6	0.02%	0.11%	122	0.01%	0.00%
	Homebrew/homebrew-core	5	0.02%	0.03%	15	0.00%	0.01%
	DefinitelyTyped/ DefinitelyTyped	3	0.01%	0.02%	448	0.02%	0.01%
	etcd-io/etcd	3	0.01%	0.02%	121	0.01%	0.00%
	mono/mono	3	0.01%	0.02%	491	0.02%	0.00%
	servo/servo	3	0.01%	0.02%	490	0.02%	0.01%
Facebook 23,752commits 4,810,358 SLOC	facebook/react-native	10186	42.88%	63.03%	2333458	48.51%	84.69%
	pytorch/pytorch	8884	37.40%	48.23%	1915022	39.81%	49.83%
	facebook/react	1996	8.40%	15.13%	145712	3.03%	9.83%
	torvalds/linux	1774	7.47%	0.67%	81745	1.70%	0.71%
	atom/atom	272	1.15%	0.87%	5951	0.12%	0.31%
	apple/swift	149	0.63%	0.64%	12117	0.25%	0.33%
	apache/spark	104	0.44%	0.63%	14441	0.30%	0.58%
	prettier/prettier	90	0.38%	1.40%	175077	3.64%	11.56%
	chef/chef	81	0.34%	0.39%	122272	2.54%	13.01%
	Homebrew/homebrew-core	74	0.31%	0.04%	381	0.01%	0.06%
	facebook/create-react-app	51	0.21%	2.27%	2922	0.06%	0.51%
	mongodb/mongo	46	0.20%	0.19%	134	0.00%	0.00%
	rust-lang/rust	11	0.05%	0.01%	134	0.00%	0.00%
	tensorflow/tensorflow	8	0.03%	0.01%	519	0.01%	0.00%
	JetBrains/intellij-community	4	0.02%	0.00%	80	0.00%	0.00%
	symfony/symfony	4	0.02%	0.01%	122	0.00%	0.01%
	DefinitelyTyped/ DefinitelyTyped	3	0.01%	0.01%	15	0.00%	0.00%
	hashicorp/vagrant	2	0.01%	0.01%	36	0.00%	0.00%
	helm/charts	2	0.01%	0.00%	47	0.00%	0.01%
	Microsoft/vscode	2	0.01%	0.00%	74	0.00%	0.00%
Pivotal 15,984 commits 1,178,250 SLOC	spring-projects/spring-boot	15885	99.38%	88.84%	1126560	95.61%	86.70%
	atom/atom	31	0.19%	0.19%	1554	0.13%	0.08%
	kubernetes/kubernetes	28	0.17%	0.10%	3470	0.29%	0.01%
	helm/charts	8	0.05%	0.04%	60	0.01%	0.07%
	Homebrew/homebrew-core	5	0.03%	0.03%	12	0.00%	0.73%
	etcd-io/etcd	4	0.02%	0.01%	1841	0.16%	0.01%
	golang/go	4	0.02%	0.04%	100	0.01%	0.00%
	apache/spark	3	0.02%	0.01%	16	0.00%	0.00%

	DefinitelyTyped/ DefinitelyTyped	3	0.02%	0.02%	28	0.00%	0.00%
	twbs/bootstrap	3	0.02%	0.00%	267	0.02%	0.00%
	gradle/gradle	2	0.01%	0.00%	29	0.00%	0.00%
	istio/istio	2	0.01%	0.00%	42230	3.58%	0.01%
	elixir-lang/elixir	1	0.01%	0.01%	62	0.01%	0.01%
	JetBrains/kotlin	1	0.01%	0.00%	16	0.00%	0.00%
	jlord/patchwork	1	0.01%	0.00%	15	0.00%	0.00%
	Microsoft/vscode	1	0.01%	0.01%	62	0.01%	0.00%
	MicrosoftDocs/azure-docs	1	0.01%	0.01%	6	0.00%	0.00%
	moby/moby	1	0.01%	0.01%	31	0.00%	0.00%
	mui-org/material-ui	1	0.01%	0.01%	4	0.00%	0.00%
GitHub 15,324 commits 1,739,004 SLOC	atom/atom	9674	63.13%	51.23%	1210213	69.59%	63.97%
	twbs/bootstrap	1459	9.52%	12.61%	216569	12.45%	14.02%
	electron /electron	1225	7.99%	13.26%	89445	5.14%	7.79%
	github/VisualStudio	1156	7.54%	29.32%	147298	8.47%	10.35%
	integrations/slack	795	5.19%	61.29%	49996	2.87%	98.66%
	jlord/patchwork	522	3.41%	0.27%	1479	0.09%	0.55%
	github/gitignore	278	1.82%	9.80%	247	0.01%	2.17%
	MicrosoftDocs/azure-docs	46	0.30%	0.01%	1149	0.07%	0.00%
	Homebrew/homebrew-core	39	0.25%	0.07%	537	0.03%	0.08%
	rails/rails	37	0.24%	0.32%	550	0.03%	0.08%
	home-assistant/home-assistant	18	0.12%	0.20%	2193	0.13%	0.12%
	DefinitelyTyped/ DefinitelyTyped	14	0.09%	0.05%	3081	0.18%	0.02%
	pandas-dev/pandas	8	0.05%	0.02%	113	0.01%	0.01%
	notepad-plus-plus/notepad-plus-plus	7	0.05%	2.09%	885	0.05%	0.33%
	rust-lang/rust	6	0.04%	0.04%	506	0.03%	0.01%
	opencv/opencv	5	0.04%	0.05%	147	0.01%	0.00%
	dotnet/core	5	0.03%	0.28%	27	0.00%	0.00%
	remacs/remacs	4	0.03%	0.10%	1676	0.10%	0.20%
	servo/servo	4	0.03%	0.03%	3808	0.22%	0.01%
	aspnet/AspNetCore	3	0.02%	0.02%	2353	0.14%	0.02%
IBM 12,543 commits 1,340,013 SLOC	torvalds/linux	7330	58.44%	0.02%	787366	58.43%	5.13%
	istio/istio	962	7.67%	3.85%	109235	8.11%	1.43%
	moby/moby	836	6.67%	2.03%	28689	2.13%	0.70%
	apache/spark	697	5.56%	2.47%	45265	3.36%	1.82%
	nodejs/node	575	4.58%	0.32%	100386	7.45%	0.89%
	eclipse/eclipse	473	3.77%	7.09%	498	0.04%	2.52%
	kubernetes/kubernetes	440	3.51%	0.20%	10925	0.81%	0.07%
	golang/go	420	3.35%	2.04%	249983	18.55%	5.13%
	tensorflow/tensorflow	212	1.69%	0.30%	6517	0.48%	0.04%
	MariaDB/server	207	1.65%	2.34%	213	0.02%	0.96%
	apple/swift	95	0.76%	0.02%	593	0.04%	0.02%
	etcd-io/etcd	66	0.52%	0.32%	1097	0.08%	0.05%
	pytorch/pytorch	49	0.39%	0.26%	556	0.04%	0.01%
	jlord/patchwork	28	0.23%	0.01%	76	0.01%	0.16%
	mono/mono	20	0.16%	0.03%	5725	0.42%	0.14%
	DefinitelyTyped/ DefinitelyTyped	19	0.15%	0.00%	31	0.00%	0.00%
	ansible/ansible	16	0.13%	0.03%	279	0.02%	0.01%

rails/rails	15	0.12%	0.12%	23	0.00%	0.00%
chef/chef	11	0.09%	0.02%	26	0.00%	0.00%
Homebrew/homebrew-core	11	0.09%	0.00%	26	0.00%	0.00%

Table 4. Top-5 firm contributors in top-20 repositories by SLOC

Repository	Firm email domain	SLOC
Angular/angular Microsoft 2 commits – 0.00% SLOC	google.com	687066
	amadeus.com	53569
	ninja-squad.com	11626
	epam.com	3254
	ippon.fr	2562
AspNet/Core No Google commits	microsoft.com	6011216
	illyriad.co.uk	68712
	github.com	2353
	telenet.be	2166
	wanadoo.fr	1960
Azure/docs No Google commits	microsoft.com	7371387
	southworks.net	303636
	axelerate.com	6962
	contentmaster.com	5944
	averesys.com	4113
DefinitelyTyped/DefinitelyTyped Google 200 commits – 1.21% SLOC	microsoft.com	2208651
	syncfusion.com	1079309
	etrog.net.il	951976
	yworks.com	650260
	esri.com	238206
dotnet/corefx No Google commits	microsoft.com	12219477
	xamarin.com	176585
	intel.com	176585
	indcomp.co.uk	103926
	samsung.com	20881
elasticsearch/elasticsearch No Google commits	elastic.co	1088909
	elasticsearch.com	469911
	softwire.com	1943
	carrotsearch.com	1159
	microsoft.com	947
go/go Microsoft 20 commits – 0.0.2% SLOC	google.com	738619
	ibm.com	249983
	intel.com	111923
	arm.com	90208
	develer.com	75532
godotengine/godot No Microsoft Google 9 commits – 0.00% SLOC	suse.cz	2924
	comcast.net	2679
	rastergrid.com	1222
	laughlin.com	1210
	gamblify.com	1076
kubernetes/Kubernetes Microsoft 305 commits – 0.44% SLOC	google.com	8370118
	redhat.com	2188159
	vmware.com	191770
	huawei.com	173676
	microsoft.com	155720
torvalds/Linux Microsoft 504 commits – 0.06% SLOC	amd.com	1416971
	intel.com	787366
	redhat.com	585041

Google 3032 commits – 0.64% SLOC	mellanox.com	223532
	ibm.com	193388
moby/moby Google 35 commits – 0.16% SLOC	docker.com	1298986.25
	microsoft.com	113752
	socketplane.io	79076
	hco.ntt.co.jp	58533
	tetrationanalytics.com	46331
mono/mono No Google commits	microsoft.com	766648
	xamarin.com	521337
	ibm.com	5725
	sinenomine.net	4853
	quamotion.mobi	4638
NixOS/nixpkgs No Microsoft commits Google 11 commits – 0.00% SLOC	merritt.tech	114706
	ironicdesign.com	83872
	logicblox.com	33141
	Obsidian.Systems	15683
	volth.com	14270
Nodejs/node Microsoft 61 commits – 0.09% SLOC	npmjs.com	197885
	google.com	193560
	ibm.com	100386
	bridgewater.de	43633
	microsoft.com	32475
pytorch/pytorch Microsoft 42 commits – 0.01% Google 98 commits – 0.05%	fb.com	1915022
	nvidia.com	42818
	intel.com	16105
	amd.com	13545
	google.com	8666
rust/rust No Microsoft commits	google.com	22124
	redhat.com	10334
	fortanix.com	9570
	scalexm.fr	7428
	system76.com	7044
servo/servo No Microsoft commits Google 1 commit – 0.00% SLOC	georepublic.de	50289
	igalia.com	36446
	samsung.com	19600
	github.com	3808
	algomi.com	2194
swift/swift No Microsoft commits Google 54 commits – 0.02% SLOC	apple.com	3214474
	fb.com	12117
	uber.com	3539
	citrix.com	3016
	google.com	2936
tensorflow/tensorflow Microsoft 119 commits – 0.02% SLOC	google.com	6927173
	intel.com	154268
	nvidia.com	79259
	codeplay.com	9233
	huawei.com	7383
vscode/vscode Google 2 commits – 0.00% SLOC	microsoft.com	6,103,554
	squarespace.com	7242

	corp.google.com	3503
	samsung.com	1039
	passportinc.com	489

Table 5. Centrality measures of one-mode network of firms with top-20 degree

	Firm	Degree	Betweenness centrality
1.	Google	1405	264473
2.	Microsoft	1193	172821
3.	Red hat	915	98789
4.	Oracle	813	30848
5.	Intel	725	24614
6.	Tencent	706	16033
7.	HP	685	19270
8.	Huawei	663	12396
9.	Cisco	654	17742
10.	Mirantis	620	17648
11.	VMWare	620	17648
12.	IBM	617	9183
13.	Ericsson	611	12481
14.	HPE	611	12481
15.	Yandex	611	12481
16.	Facebook	595	33078
17.	Canonical	583	11250
18.	ARM	569	10893
19.	Tuxera	556	14280
20.	amazon.com	544	4628

Table 6. Top-20 dyads by weight of tie in one-mode network of firms

	Firm 1	Firm	Weight of tie
1.	South works	Microsoft	287093325
2.	Red Hat	Intel	93443661
3.	Moravia	Microsoft	75225925
4.	Red Hat	Google	74495611
5.	Intel	Google	56810244
6.	Intel	AMD	56586496
7.	Red hat	AMD	47169080
8.	Mellanox	Intel	36849892
9.	Red Hat	Mellanox	30778648
10.	Oracle	Intel	26835188
11.	Intel	Huawei	26648797
12.	Intel	ARM	25421122
13.	Red Hat	Huawei	24937983
14.	Huawei	Google	23249531
15.	SUSE	Intel	22802346
16.	Red Hat	Oracle	22465148
17.	Microsoft	Knowledge factory	21284225
18.	Red Hat	ARM	21184136
19.	Xamarin	Microsoft	21165388
20.	Nvidia	Google	19148103

Table 7. Network of contributors and firms on GitHub, 01.01.2015 to 30.04.2019
Directed-weighted network

Size: 26,594

Projects: 135

Contributors: 26,459

No. of ties: 36,980

Density: 0.00005

	Mean	**min**	**1st Qua.**	**2nd Qua.**	**3rd Qua.**	**Max**
Indegree of projects	275	1	49	131	271	3,378
Outdegree of firms	1.4	1	1	1	1	133
Tie weights	76	1	1	2	4	329,552

Table 8. Top-100 contributors to selected 135 repositories on GitHub

	Category	**Email domains**	**GitHub accounts**	**Number of commits**
Shared email domains	Firms	41	11,538	1,025,681
	FOSS foundations	19	1,069	149,293
	No affiliation*	20	68,913	1,060,168
	Research institutions	1	565	42,451
Individual domains	Individual developers	19	19	68,133
Total top-100	N/A	100	82,104	2,345,726
All domains	N/A	26,459	113,614	2,824,690

* Comprises email domains such as users.noreply.github.com, outlook.com, qq.com (common among Chinese developers), and mail.ru (a popular Russian email provider).

Table 9. Contributions of developers committing with both firm and personal email addresses

	Total No. developers	Total No. commits	Mean No. individual commits	Median No. individual commits	Total SLOC	Mean individual SLOC	Median individual SLOC
All developers	113,614 (100%)	2,824,690	24.9	2.0	305,211,890	3,524	8
Developers committing with both firm and non-firm email addresses	3,279 (2.9%)	614,746 (21.8%)	183.9	26.0	74,536,051 (24.4%)	24,535	487

Table 10. Committing time patterns of developers using multiple email (N: 3,279)

Category	Description	Number of developers	Total commits with firm email addresses	Total commits with non-firm email addresses
Occasional	Less than 10% overlap between firm / non-firm email account use	1010	236,234	143,160
Parallel	Developers who frequently switched between firm and non-firm addresses and never used one type for more than 180 days	750	107,133	79,313
Sequential	Developers who only changed email address type once, or who used the same type of email address for more than 180 days	862	25,074	21,600
Left out	Developers positioned in the bottom quintile by total number of commits	675	1088	1144

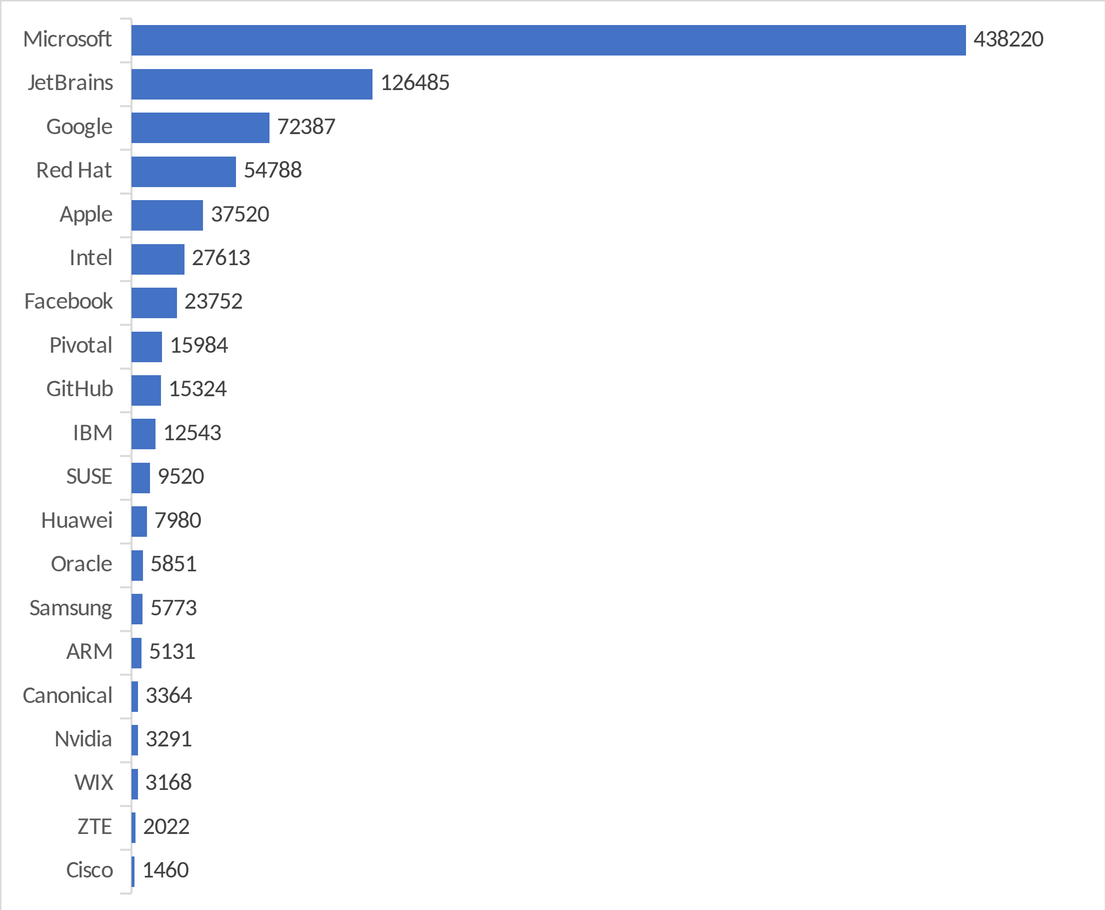

Figure1. Top-20 firm contributors (by number of commits) in 135 GitHub repositories, 01.01.2015 to 30.04.2019

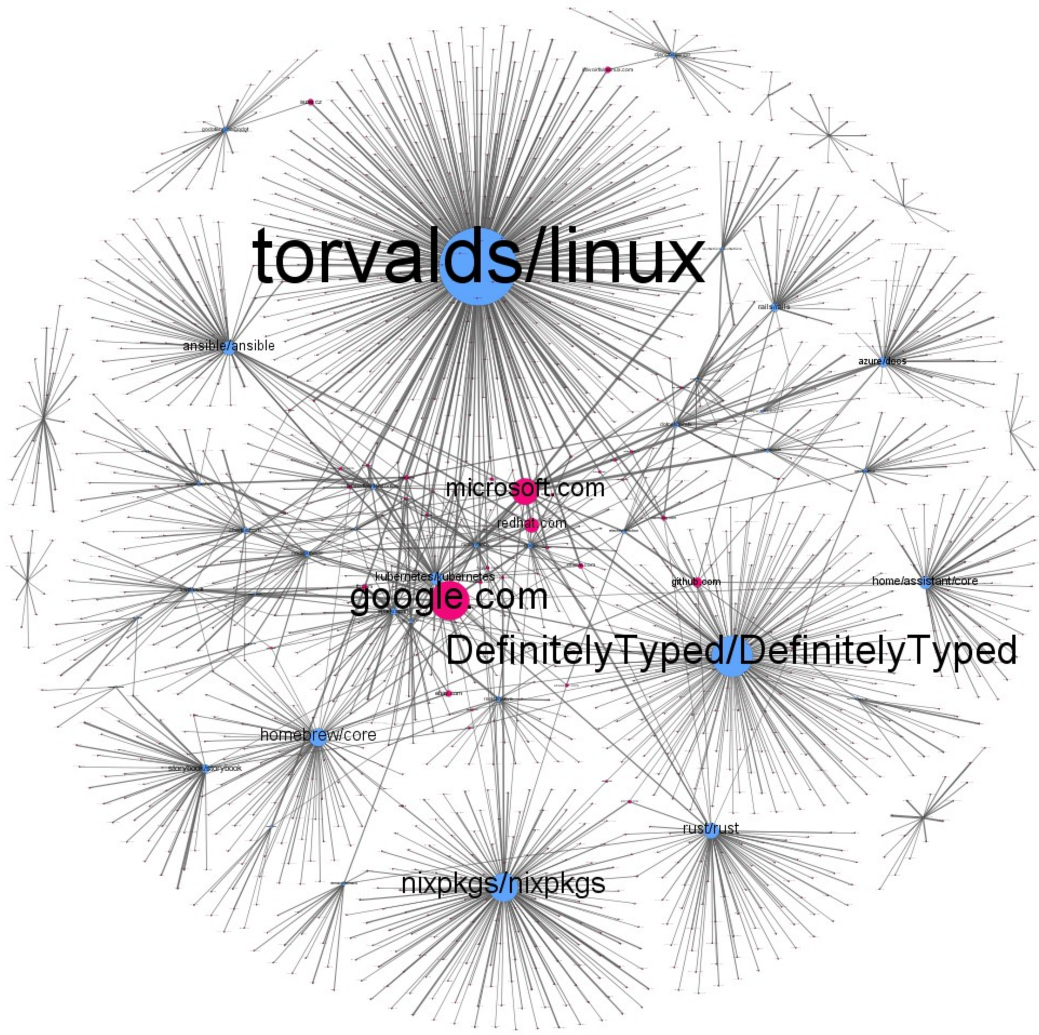

Figure 2. Two-mode network of firms and repositories. Node size reflects betweenness centrality.

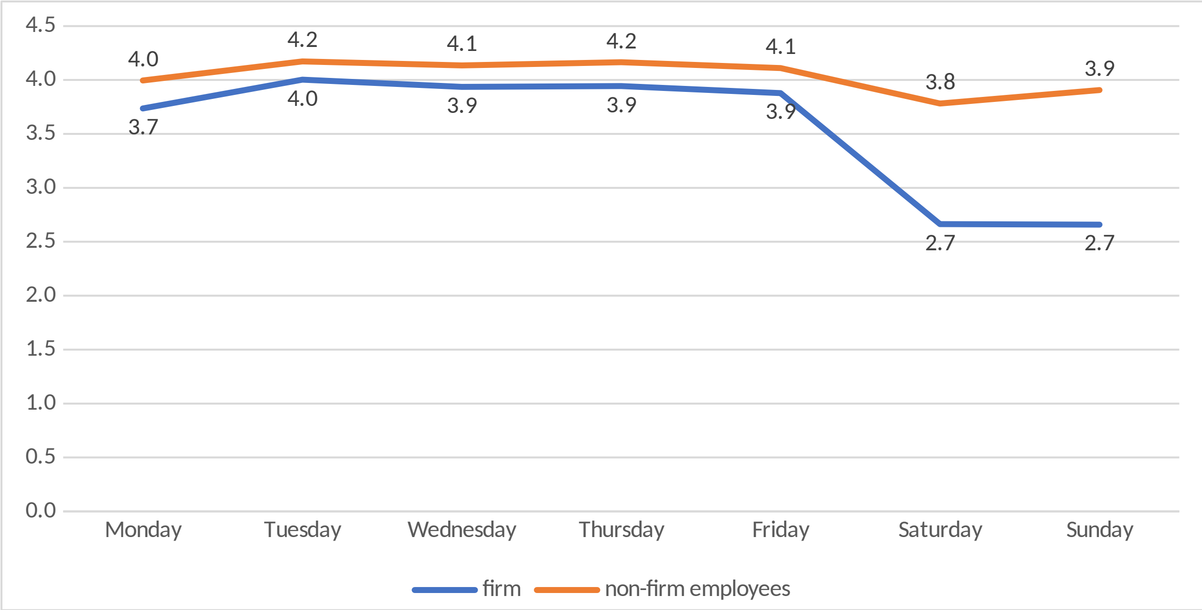

Figure 3. Average number of commits by a firm-employee and a non-firm-employee per day, 01.01.2015 to 30.04.2019

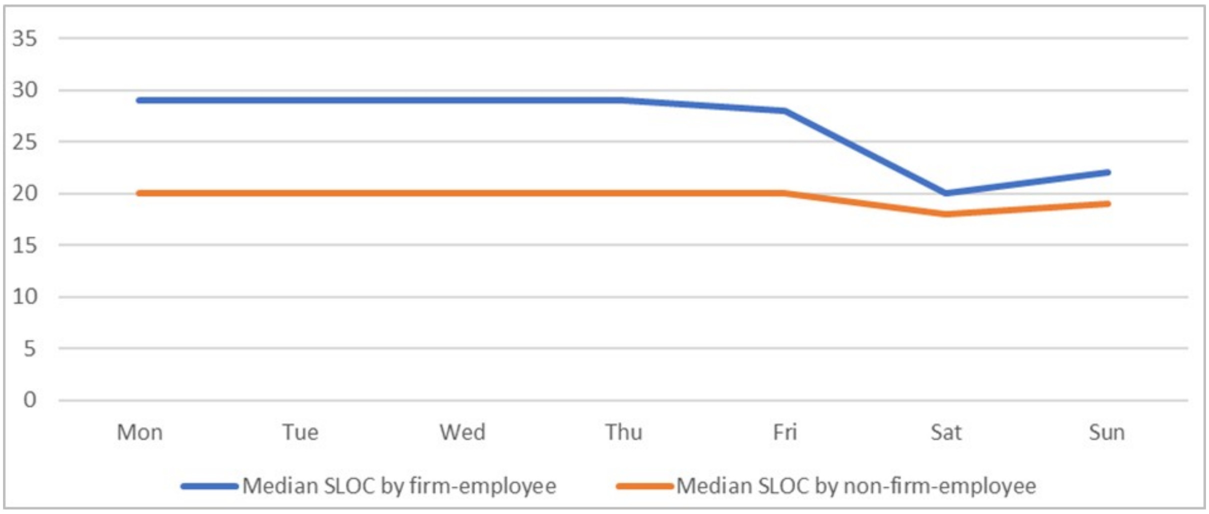

Figure 4. Median Source Lines of Code committed by individuals in a week

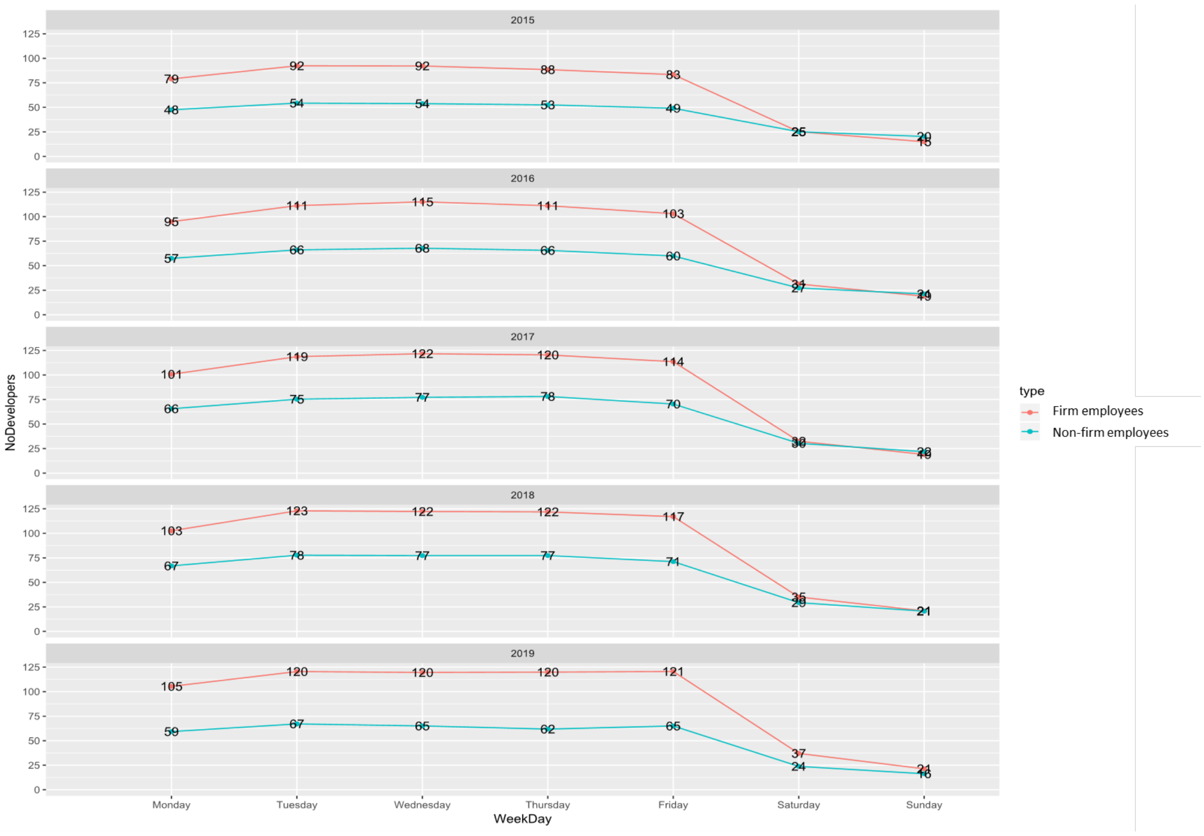

Figure 5. Average number of firm or personal email addresses of the 3,279 developers using both firm and personal email addresses to commit per week, 2015-2019

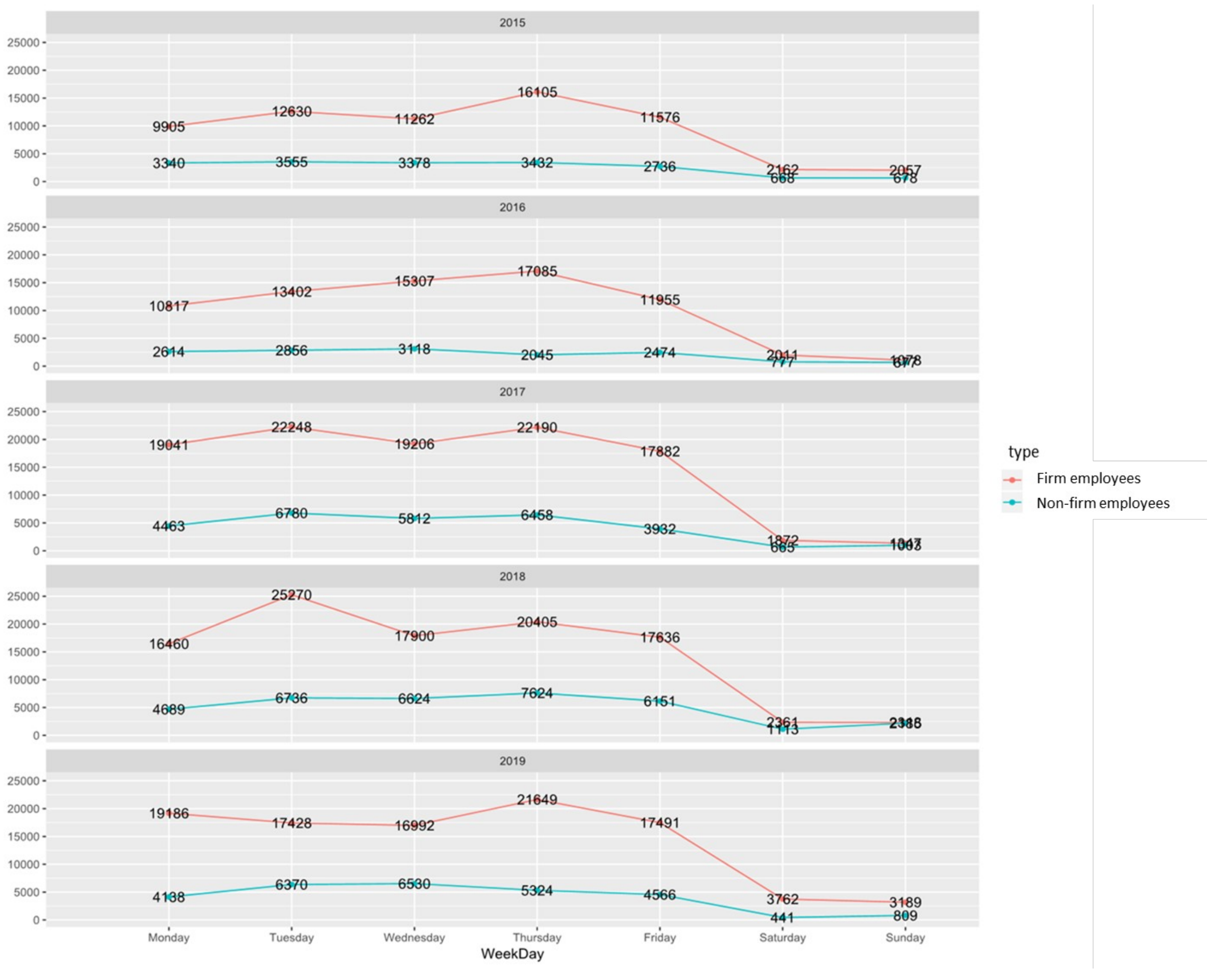

Figure 6. SLOC submitted by 3,279 developers using both firm and personal email addresses by type of email address in a week, 2015-2019

APPENDIX

Firm email address domains

microsoft.com, google.com, fb.com, apple.com, redhat.com, intel.com, docker.com, amd.com, databricks.com, huawei.com, xamarin.com, mellanox.com, linux.vnet.ibm.com, coreos.com, github.com, oracle.com, arm.com, plataformatec.com.br, samsung.com, nvidia.com, canonical.com, elasticsearch.com, ti.com, suse.com, itseez.com, free-electrons.com, hashicorp.com, nxp.com, us.ibm.com, volth.com, renesas.com, zte.com.cn, broadcom.com, jetbrains.com, osg.samsung.com, imgtec.com, bitnami.com, socionext.com, baylibre.com, suumit.com, cloudera.com, bootlin.com, sophiebits.com, lowenna.com, mediatek.com, logicblox.com, call-em-all.com, netronome.com, atmel.com, linux.ibm.com, synopsys.com, cisco.com, tuxera.com, rock-chips.com, st.com, cavium.com, embeddedor.com, cn.fujitsu.com, microchip.com, de.ibm.com, freescale.com, visionengravers.com, vmware.com, alibaba-inc.com, sang-engineering.com, marvell.com, moravia.com, percolatestudio.com, unisys.com, collabora.com, dependabot.com, realtek.com, chelsio.com, cumulusnetworks.com, alipay.com, amadeus.com, newtonking.com, fathomdb.com, wdc.com, windriver.com, virtuozzo.com, stanton-nurse.com, basecamp.com, ca.ibm.com, cogentembedded.com, savoirfairelinux.com, amazon.com, elao.com, ingics.com, xilinx.com, zoho.com, palantir.com, ebay.com, xmission.com, bp.renesas.com, hovoldconsulting.com, ericsson.com, netapp.com, sandisk.com, imagia.com, fortinet.com, alienfast.com, poweredbysearch.com, telegraphics.com.au, thoughtbot.com, qualcommdatacenter.com, hichroma.com, qti.qualcomm.com, mirantis.com, isquaredsoftware.com, qca.qualcomm.com, i2se.com, ironicdesign.com, igalia.com, taehoonlee.com, wyeworks.com, toradex.com, hisilicon.com, mips.com, atlassian.com, hortonworks.com, ibm.com, axis.com, shopify.com, mentor.com, citrix.com, omnibond.com, solarflare.com, qlogic.com, 2ndQuadrant.com, tieto.com, jp.ibm.com, hammerspace.com, slack-corp.com, hpe.com, speartail.com, in.ibm.com, thatsquality.com, syncfusion.com, apm.com, lge.com, mani-desktop.redmond.corp.microsoft.com, oculus.com, mvista.com, cn.ibm.com, instagram.com, purestorage.com, beyondcent.com, codeplay.com, epam.com, tetrationanalytics.com, uk.ibm.com, caviumnetworks.com, dell.com, diasemi.com, emikra.com, hp.com, squarespace.com, flightdataservices.com, lenovo.com, microsemi.com, aquantia.com, amarulasolutions.com, sensiolabs.com, salesforce.com, opensource.cirrus.com, sap.com, janeasystems.com, uber.com, endlessm.com, ni.com, appunite.com, linux.alibaba.com, tencent.com, cmss.chinamobile.com, opensource.altera.com, mleia.com, partner.samsung.com, 6wind.com, analog.com, sony.com, avinetworks.com, spreadtrum.com, deis.com, baidu.com, mojatatu.com, andestech.com, itslearning.com, ninja-squad.com, opengridcomputing.com, quantenna.com, netflix.com, networktocode.com, quatrodev.com, amlogic.com, ntdev.microsoft.com, develer.com, etacompute.com, fortanix.com, lyft.com, twitter.com, jp.fujitsu.com, leadformance.com, sifive.com, arangodb.com, emulex.com, mycase.com, nokia.com, obsidianresearch.com, wizery.com, asana.com, cloudflare.com, siemens.com, visteon.com, wix.com, corp.microsoft.com, sonymobile.com, averesys.com, nicira.com, appier.com, ninemoreminutes.com, trendmicro.com.cn, xamla.com, akamai.com, au1.ibm.com, deltatee.com, mce-sys.com, qiniu.com, ezchip.com, fromorbit.com, secunet.com, synaptics.com, contentmaster.com, f5.com, madriska.com, pmcs.com, vinsol.com, whirlscape.com, ah.jp.nec.com, android.com, orange.com, orobix.com, raizlabs.com, arista.com, plumgrid.com, spiritit.com, teledini.com, compuix.com, efficios.com, goldelico.com, openmesh.com, tensortable.com, capitalone.com, engineyard.com, esri.com, lemote.com, profitbricks.com, aliyun.com, boundarydevices.com, jfrog.com, tw.ibm.com, twosigma.com, xenapto.com, antfin.com, datadoghq.com, webenchanter.com, bendarnell.com, clockwisehq.com, cnexlabs.com, csr.com, enlambda.com, canva.com, electromag.com.au, airbnb.com, softwire.com, xebia.com, adobe.com, HansenPartnership.com, mirakl.com, parallels.com, unboxedconsulting.com, autodesk.com, liferay.com, wolksoftware.com, akvelon.com, logand.com, rackspace.com, thoughtworks.com, chibifire.com, meituan.com, okta.com, stelligent.com, areca.com.tw, cloudbasesolutions.com, redpinesignals.com, truelocal.com.au, visi.com, wx.jp.nec.com, zendesk.com, hitachi.com, konsulko.com, semihalf.com, calsoftinc.com, emc.com, stylight.com, blick-labs.com, clarifai.com, holidaycheck.com, widen.com, brocade.com, grin.com, hill-wood.com, journeyteam.com, linbit.com, metanate.com, za.ibm.com, appscode.com, bytedance.com, centricular.com, cray.com, idt.com, ikuai8.com, nventive.com, runbox.com, ultimatesoftware.com, apcera.com, claudetech.com, cypress.com, manas.com.ar, open-homeautomation.com, sigmadesigns.com, narrativescience.com, onepeloton.com, ambiqmicro.com, apprenda.com, itron.com, joyent.com, kyup.com, newdealmultimedia.com, sgi.com, whmsi.com, xiaomi.com, box.com, candelatech.com, javigon.com, Netapp.com, nuvoton.com, princeton.com, qualdan.com, ru.ibm.com, shutterstock.com, blue-zinc.com, linkedin.com, squareup.com, tngtech.com, xing.com, ddn.com, didichuxing.com, diephouse.com, eversports.com, lazada.com, publicisfrontfoot.com.au, realeyes.com, ronline.com, sendgrid.com, technergetics.com, terminajones.com, awakenetworks.com, creditkarma.com, digitalmentat.com, elsevier.com, genesys.com, navercorp.com, sleepybelly.com, annapurnalabs.com, deif.com, delmarsd.com, grabcad.com, m2mobi.com, rakuten.com, red-bean.com, toptal.com, upserve.com, animalcreek.com, apptus.com, ciferox.com, clear-code.com, climate.com, evilmartians.com, mendix.com, parrot.com, peak-system.com, telogis.com, thenetcircle.com, xored.com, armadeus.com, cadence.com, cockroachlabs.com, criteo.com, hds.com, ingarfield.com, jda.com, kitware.com, miniconfig.com, open-mesh.com, platonix.com, reaktor.com, simplearchitect.com, synology.com, theobroma-systems.com, thinkoomph.com