

ソースコードバージョン管理を目的としない GitHub リポジトリの調査・分析

茅田 武広¹ 戸田 航史¹

概要：GitHub はバージョン管理システム Git のホスティングサービスとして広く使われている。ただし、GitHub はソースコードのバージョン管理以外にも、コードレビューや Issue 管理、Wiki 等のシステムも提供している。近年の GitHub のリポジトリ数、ユーザ数の増加に伴い、Git の利用を主目的としない、もしくは README.md の更新以外には Git を利用しないプロジェクトが現れ始めている。これらのプロジェクトは GitHub を Issue 管理システムの利用やドキュメントの管理として利用することを主目的としている。本研究ではこのようなプロジェクト（Hub プロジェクト）に着目し、分析を行った。分析では Hub プロジェクトを 3 つのパターン（Issue Tracker Type, Document Type, Link Collection Type）に分類し、リポジトリ数の増加の推移や最古リポジトリ等について調査した。分析には GitHub に存在する約 1.2 億のオープンリポジトリのデータである GHTorrent を用いた。分析では、あらかじめ手動で収集した 10 の Hub プロジェクトから導出した共通点を利用し、不要なリポジトリをを除去した。分析の結果、タイプ別のリポジトリ増加推移や最古リポジトリ、そして増加のきっかけとなったリポジトリを発見できた。

キーワード：Git, GitHub, オープンソースソフトウェア, バージョン管理システム

1. はじめに

近年のオープンソースソフトウェア（以降 OSS と呼ぶ）には商用ソフトウェアと同等以上の機能・品質を備えたものが数多く存在し、エンドユーザのみならず、企業におけるソフトウェア開発においても広く利用されるようになってきている。このような OSS の隆盛の一つの要因として、商用プロジェクトとは異なり、誰でも参加可能ということが挙げられる。誰でも参加できる可能という障壁の低さから、異なる特性を持つ開発者が集まり、その異なる特性で互いを補い合い協働することで、より高機能、高品質なソフトウェアが開発されていると考えられる。

このような OSS におけるソフトウェア開発を支えるのがソフトウェア開発管理システムである。誰でも参加できるということは、その特性（スキル、志向等）も異なるため、協働作業を支援するシステムがなければ各個人が好き勝手に開発を進め、結果的にソフトウェアの品質の低下や同一機能の多重実装を招くことになる。

現在の OSS ではその開発過程において Modern Code Review (MCR) が採用されている [1]。MCR はパッチに含まれるのはパッチに含まれる欠陥の検出やコードの品質向

上を目的として行われる。MCR の実施においては複数のソフトウェア開発管理システムの連携が必要である。例えば、バージョン管理システムである Git、レビュー管理システムである rietveld や Gerrit、Issue 管理システムである Bugzilla や Google Issue Tracker 等を相互に連携させることで MCR の実施環境を成立させている。しかしながら、これらのシステムを OSS プロジェクトがおのおので管理することは高コストであり、それに対するアプローチの 1 つとして、GitHub [2]、Bitbucket^{*1}、GitLab^{*2} といった、これらのサービスを一括で提供するホスティングサービスが現れ始めた。

特に GitHub はホスティングサービスの開始以降広く利用されており、2013 年には 300 万ユーザ、500 万リポジトリを達成している^{*3}。また、GitHub はソフトウェア開発管理としての機能のみならず、協働作業を支援することを目的として、開発者への following やリポジトリへの starring 等の SNS 的な機能も備えている。

このような各種システムの統合と、SNS 的な機能の導入により、GitHub において Social Coding という概念が生

^{*1} <https://bitbucket.org/>

^{*2} <https://about.gitlab.com/>

^{*3} <https://thenextweb.com/insider/2013/04/11/code-sharing-site-github-turns-five-and-hits-3-5-million-users-6-million-repositories/>

¹ 福岡工業大学
Fukuoka Institute of Technology

まれ、数多くの研究の対象となっている [3][4][5].

このように、GitHub はその名の通り特に Git を主軸としつつも、Social Coding の機能を備えた効率的なソフトウェア開発を可能とするサービスとみなされている。しかしながら、Gousios らによる広範な GitHub 上に存在するリポジトリの分析に際して Documentation のリポジトリが除外されたように [6], ソフトウェア開発や Git の利用を主たる目的としていないプロジェクトも GitHub 上には存在している。例えば `microsoft/TypeScript-wiki`[7] は、開発を担う Microsoft 自身によるプログラミング言語 TypeScript に関するドキュメントの公開。共有を目的としたリポジトリである。また `microsoft/WSL`[8] は Windows Subsystem for Linux の Issue 管理のみを目的としており、Git で管理されているファイルは 10 程度であるにも関わらず、これまでに報告された Issue は 4000 件を優に超えている。

このようなプロジェクトの存在は、もともと Git のホスティングサービスを主体として提供が始まった GitHub の使われ方が変化しつつあることを示している可能性がある。以降、本研究ではこのような GitHub 上に存在しながら Git の利用を主体としないプロジェクトのことを Hub プロジェクトと呼ぶ。

本研究ではこの Hub プロジェクトに着目し、Hub プロジェクトそのものの調査や、その発生（最古のプロジェクト）、増減の推移を分析し、その詳細を明らかにすることを目的とする。そのために、以下の 3 つの Research Question(RQ)を立て、GitHub 上に存在するプロジェクトの分析を行った。

RQ1: Hub プロジェクトにはどのようなものがあるのか

RQ2: Hub プロジェクトの件数は経年変化しているのか

RQ3: RQ2 にて変化が見られた場合、その原因は何か

以降 2 節で Hub プロジェクトの定義と詳細について述べ、3, 4, 5 節でそれぞれ RQ1, RQ2, RQ3 について述べる。最後に 6 章にて研究のまとめと今後の課題を述べる。

2. Hub プロジェクト

GitHub では Git によるソースコード等のバージョン管理以外にも、コードレビューや Issue 管理, Wiki 等のシステムも提供されている。そして近年の GitHub のリポジトリ数、ユーザ数の増加に伴い、むしろ Git の利用を主目的としない、もしくは全く利用しないプロジェクトが現れ始めている。これらのプロジェクトを本研究では Hub プロジェクトと呼び、分析対象とする。

代表的な Hub リポジトリを 3 例挙げて説明する。

1. `microsoft/WSL`[8]

`microsoft/WSL` は、Microsoft 社が開発中の Windows Subsystem for Linux についてユーザーからのフィードバックを GitHub の Issues 機能を用いて受けることを目

的としたリポジトリである。ユーザーは WSL の不具合や改善点等があれば Issue を建てることができ、GitHub の Issue 管理システムがデベロッパーとユーザとの議論の窓口の役割を果たしている。

2. `microsoft/TypeScript-wiki`[7]

`microsoft/TypeScript-wiki` は Microsoft 社が開発している代替 JavaScript 言語 (通称 altJS) のドキュメント公開を目的としたリポジトリである。ドキュメントにはマークダウン形式を採用しており、GitHub のマークダウンレビュー機能を用いてドキュメントを参照することが出来る。

3. `rehooks/awesome-react-hooks`[9]

`rehooks/awesome-react-hooks` は JavaScript ライブラリ React の機能である React Hooks で活用できる様々な Hook をリンク集として Markdown 形式のドキュメント (readme.md) に記述したものを Git で管理している。ドキュメントへのリンク追加フローは contributor がリポジトリを fork し、ドキュメントに追加した後に Pull Request を出すという、GitHub における Pull Request による開発に則った手順で行われる。

3. RQ1: Hub プロジェクトにはどのようなものがあるのか

本研究では分析対象とする Hub プロジェクトについて、2 節で 3 例挙げた。しかしながらそれ以外にも Hub プロジェクト、すなわち Git を積極的に利用していないプロジェクトは存在する可能性がある。そこで、以降で示す手順に従って GHTorrent[10] から Hub プロジェクトを抽出し、そこからさらにランダムに 10 件抽出したプロジェクトについて著者らが確認・分類を行った。

3.1 GHTorrent

GHTorrent[10] は GitHub API を用いて取得可能な GitHub 上の全ての活動データを保存したデータセットである。本研究では、GHTorrent を用いて Hub プロジェクトを抽出し分析に用いた。最新の収集タイミングは 2019 年 6 月 1 日時点であり、2007 年 10 月から 2019 年 6 月までの約 1.2 億のオープンリポジトリが格納されている。なお、2019 年のデータは 6 月までしか収集されていないことから本研究の分析対象は 2018 年末までとした。

3.2 Hub プロジェクト抽出

GHTorrent に記録されている約 1.2 億のオープンリポジトリデータから Hub プロジェクトを抽出するために表 1 に示す条件を設定した。

抽出条件の設定理由について順に述べる。

(1) Fork 元が存在しない

別のリポジトリから fork された派生プロジェクトを除外

表 1 Hub プロジェクト抽出条件

- (1) Fork 元が存在しない
- (2) 削除されていない
- (3) Description (説明書き) が設定されている
- (4) 主要言語指定が無い
- (5) 作成日と最新更新日が同一ではない

するため、この条件を設定した。

(2) 削除されていない

GHTorrent のレコードは論理削除を行っているため、GitHub から削除されたプロジェクトであっても GHTorrent 上ではデータとして残っている。分析では現存するリポジトリのみを対象としたいため、この条件を設定した。

(3) Description (説明書き) が設定されている

Description が無いプロジェクトは作成された状態で放置されているケースが多いため、この条件を設定した。

(4) 主要言語指定がない

Hub リポジトリであればソースコードが含まれていないことから主要言語指定は無いと断定できるため追加した。なお、GitHub の主要言語指定には `linguist`^{*4} と呼ばれるライブラリが用いられている。

(5) 作成日と最新更新日が同一ではない

GitHub のテスト等の作成後一度も更新されていないプロジェクトを除去するために追加した。

分析のために、表 1 を元に GHTorrent データセットから Hub プロジェクトを抽出する SQL スクリプトを作成した。スクリプトにより約 1.2 億件のリポジトリデータを約 37 万件までに絞り込むことができた。なお、前処理および分析に用いたスクリプトは GitHub にて公開している^{*5}。

抽出した約 37 万件のプロジェクトからランダムに 10 件抽出し、その特徴を調査した。調査対象のプロジェクトサンプルを表 2 に示す。

3.3 調査結果

調査の結果、Hub プロジェクトは Issue Tracking Type (Type 1), Document Type (Type 2), Link Collection Type (Type 3) の 3 種類に分類できることが明らかとなった。その概要を以下に示す。以降、この分類に従って分析を行う。

Issue Tracking Type (Type 1)

Issue 機能の利用を主な目的としたリポジトリ

Document Type (Type 2)

ドキュメント (Wiki を含む) 公開を目的としたリポジトリ

Link Collection Type (Type 3)

他のリポジトリや外部サイトへのリンク集を公開目的

^{*4} <https://github.com/github/linguist>

^{*5} <https://github.com/TakehiroTada/hub-projects-collection-script>

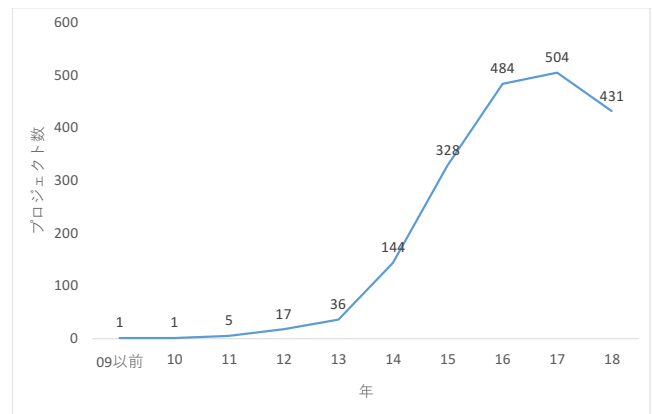


図 1 Hub プロジェクト数の経年変化

としたリポジトリ

4. RQ2: Hub プロジェクトの件数は経年変化しているのか

3.2 節で抽出した Hub プロジェクトの経年変化を図 1 に示す。図中の横軸は各年を、縦軸はその年に新規作成された (fork 元が存在しない) リポジトリ数を表している。図より、年を追うごとに Hub プロジェクトの数は増加傾向にあるが、2017 年を境に停滞、減少傾向にあることが分かる。

4.1 Type ごとの経年変化

4.1.1 Hub プロジェクトのタイプ別分類方法

Type ごとの経年変化について調査するためには、3.2 節で抽出したプロジェクトをさらにタイプ別に抽出する作業が必要になる。具体的な抽出方法を以下に示す。

Type 1 の抽出方法

表 2 の Hub プロジェクトサンプル中の Type 1 の description に共通する特徴を調査した結果として description に「issues」というワードが含まれていることが分かった。この特徴を利用して 3 の抽出結果から description に「issues」が部分一致する条件で抽出を行った。また抽出結果中の数件について、この抽出条件が正しいことを目視で確認した。

Type 2, 3 の抽出方法

表 2 の Hub プロジェクトサンプル中の Type 2, Type 3 に共通する特徴を調査した結果として、リポジトリ名に「最高」や「素晴らしい」を意味する「awesome」という単語が含まれていることが分かった。そこで、まず 3.2 節の抽出結果から Type 1 を取り除き、次に Repository Name に「Awesome」が部分一致する条件で抽出を行った (以降、Type 2, 3 の抽出結果を **Awesome** プロジェクトと呼ぶ)。

次に、Type 2, 3 を区別するため、Awesome プロジェクト中の全ての readme.md をクローリングし、その空白行を除いた行数からリンクが含まれている行数の割合を算出

表 2 Hub プロジェクトサンプル

Repository Name	Type	Description
microsoft/WSL	Type 1	Issues found on WSL
rehooks/awesome-react-hooks	Type 3	Awesome React Hooks Resources
awesomedata/awesome-public-datasets	Type 3	The list of a topic-centric public data sources in high quality.
awesome-selfhosted/awesome-selfhosted	Type 3	This is a list of Free Software network services and web applications which can be hosted locally
Microsoft/TypeScript-wiki	Type 2	This repo is the source of truth for the GitHub wiki
gaerae/awesome-algorithms-education	Type 3	An algorithm is a list of rules to follow in order to solve a problem.
Azure/service-fabric-issues	Type 1	This repo is intended for reporting and tracking general issues and small feature requests for Azure Service Fabric.
MarkMoHR/Awesome-Image-Colorization	Type 3	A collection of Deep Learning based Image Colorization papers and demos.
RainwayApp/bug-tracker	Type 1	Your feedback helps us improve Rainway.
philipwalton/flexbugs	Type 1	This repository is a community-curated list of flexbox issues and cross-browser workarounds for them.

表 3 Awesome プロジェクトの一部

file	links	lines	pers
awesome-objc-frameworks	540	544	99.26%
awesome-personal-blogs	188	190	98.95%
Awesome-Hacking-Tools	1,069	1,081	98.89%
awesome-lists	1,010	1,022	98.83%
awesome-vue	465	480	96.87%
...	...	...	...
awesome-adb	20	1,780	1.12%
JS-Regular-expression-awesome	1	90	1.11%
awesome-json2json	2	422	0.47%
awesome-manifesto	0	230	0%
awesome-product-thought	0	83	0%

した。Type 3 はその性質上、リンクが大量に含まれていると考えられる。従って「リンクを含む行数 ÷ readme.md 中の非空白行数」の値が大きければ Type 3 と判定可能であると考えたためである。手順としては、まず readme.md が存在しない、もしくはその行数が 30 未満のリポジトリを除外し、その上で各行のリンクの有無判定には「http」文字列の部分一致検索を行いリンクありの行数を算出した。リンク行数割合が上位および下位のプロジェクトの一部を表 3 に示す。この過程で得られた 10% ごとのヒストグラムを図 2 に示した。これらを元に今回は 25% 未満を Type 2, 26% 以上を Type 3 とした。

これらを踏まえたプロジェクトごとのタイプを表 2 に記載した。

4.1.2 Hub プロジェクトのタイプ別経年変化

4.1.1 節での抽出・分類の結果、2009 年時点における Hub プロジェクトは 1 件 (Type 1: 1 件, Type 2: 0 件, Type 3: 0 件) であり、2018 年末あたりにおける Hub プロジェクトは 431 件 (Type 1: 99 件, Type 2: 11 件, Type 3: 321 件) であった。

Hub リポジトリのタイプ別経年変化を図 3~図 5 に示す。図 3 は Type 1, 図 4 は Type 2, 図 5 は Type 3 の年

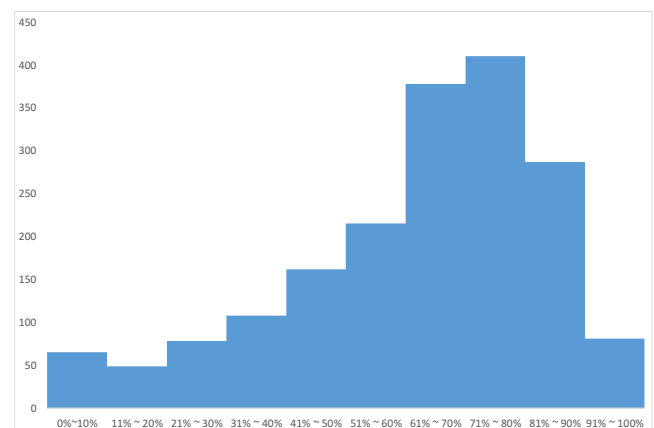


図 2 Awesome プロジェクトのリンクの割合のヒストグラム

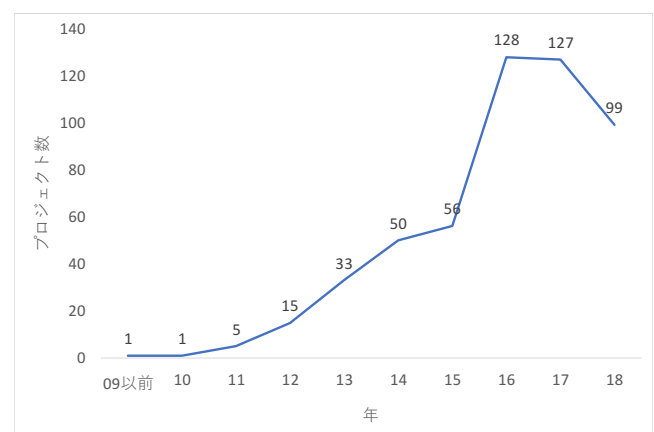


図 3 Type 1 プロジェクト数の推移

ごとの推移である。また比較のために、GitHub 全体における fork 元の存在しない新規リポジトリ数の経年変化を図 6 に示す。

全ての図において縦軸は新規作成された (fork 元が存在しない) プロジェクトの数、横軸は年を表す。Type 1 は 2010 年頃から徐々に増加し、2015 年から 2016 年にかけての急激な増加があったことが、Type 3 は 2013 から 2015

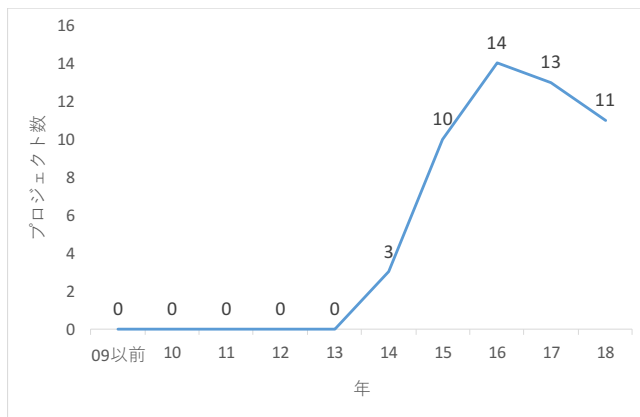


図 4 Type 2 プロジェクト数の推移

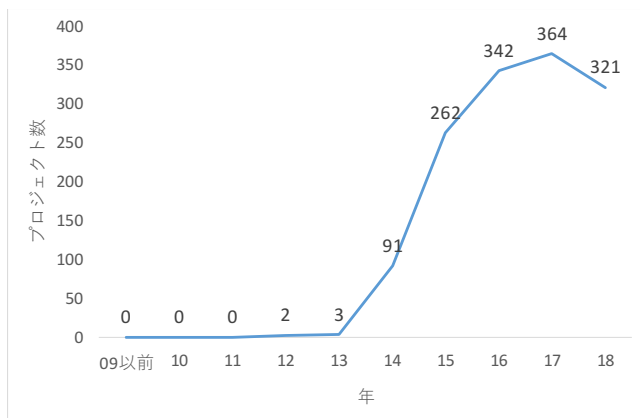


図 5 Type 3 プロジェクト数の推移

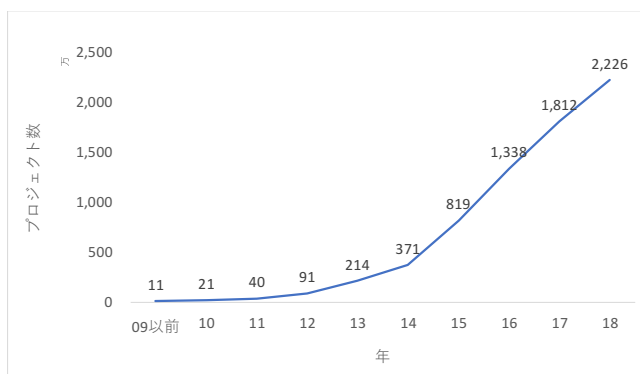


図 6 GitHub 全体の新規リポジトリ数の推移

年にかけて急激な増加があったことが分かる。ただし 2016 年以降、新規に作成されるリポジトリの数は停滞、減少傾向にある。

また 2014 年以降の新規プロジェクト数を表 4、前年と比較した場合のプロジェクト数数の増加率を表 5 に示す。2014 年以前は特に Type 2, Type 3 のプロジェクトがほとんど存在しないため、省略した。また比較対象として、GitHub 全体の fork 元の存在しない新規リポジトリ数を追加した。

表から、2016 年までは新規に作成された Hub プロジェクトは前年度を上回り、特に 2014 年、2015 年の Hub プロ

ジェクト全体、2016 年の Type 1 のプロジェクトの新規作成プロジェクト数は GitHub 全体のリポジトリ増加率を上回っている。また前述の Type 1 の 2015 年から 2016 年にかけての急激な増加、Type 3 は 2013 年から 2015 年にかけての急激な増加もこの増加率の結果から裏付けられる。

5. RQ3: RQ2 にて変化が見られた場合、その原因は何か

RQ2 の結果から、Hub プロジェクトは近年までは増加傾向にあり、特に Type 1, Type 3 では特定の期間に大幅に増加していることが分かった。これを踏まえて RQ3 として、Type 1, Type 3 の大幅な増加の原因について調査する。ここで、Type 2 についてはそもそもの件数の少なさから除外した。

5.1 Type 1 の増加原因

Type 1 プロジェクトの増加の原因としては microsoft/WSL の影響が考えられる。当該プロジェクトは 2016 年 4 月に作成されており、そのプロジェクトオーナーが Microsoft であることもあり、Issue 管理システムとしてのみ GitHub を使うという、ある種のモデルケースとして扱われ、そこから 2016 年末にかけての Type 1 プロジェクトの増加に繋がったと考えられる。

5.2 Type 3 の増加原因

Type 3 プロジェクトの増加の原因としては sindresorhus/awesome の影響が考えられる。当該プロジェクトは 2014 年 7 月に作成されており、そのプロジェクトオーナーである sindresorhus は、H5BP^{*6}や editorconfig^{*7}をはじめとした 23 もの Organizations に所属しており、その影響力の大きさが Type 3 プロジェクトの増加に繋がったと考えられる。また Awesome Lists^{*8}と呼ばれる GitHub Topics が存在し、それにより 2019 年 2 月現在で 3000 以上ものプロジェクトが登録されているほどまでに増加したと考えられる。

本分析において Hub プロジェクト増加のきっかけについては調査したが、そもそも 3 つのタイプの Hub プロジェクトがいつ発生したのかは明らかではない。そこで追加調査として、3 つのタイプについて、その発生時期とプロジェクト詳細について調査した。

5.3 追加調査：最古の Hub リポジトリの調査

タイプ別に最古の Hub リポジトリを調査を行った。SQL スクリプトにて抽出したデータをリポジトリ作成日の昇順にソートした後、上から順に目視で

*6 <https://github.com/h5bp>

*7 <https://github.com/editorconfig>

*8 <https://github.com/topics/awesome>

表 4 新規リポジトリ作成数の年推移

	2013	2014	2015	2016	2017	2018
GitHub 全体	2,136,018	3,710,667	8,192,282	13,380,377	18,119,332	22,255,637
Hub プロジェクト全体	36	144	328	484	504	431
Hub プロジェクト Type 1	33	50	56	128	127	99
Hub プロジェクト Type 2	0	3	10	14	13	11
Hub プロジェクト Type 3	3	91	262	342	364	321

表 5 新規リポジトリ作成数の前年度比

	2014	2015	2016	2017	2018
GitHub 全体	173.7%	220.8%	163.3%	135.4%	122.8%
Hub プロジェクト全体	400.0%	227.8%	147.6%	104.1%	85.5%
Hub プロジェクト Type 1	151.5%	112.0%	228.6%	99.2%	78.0%
Hub プロジェクト Type 2	—	333.3%	140.0%	92.9%	84.6%
Hub プロジェクト Type 3	3033.3%	287.9%	130.5%	106.4%	88.2%

調査した。調査の結果、Type 1 は 2016 年 4 月に作成された Microsoft 社の microsoft/BashOnWindows であった。現在は機能名称変更に伴いリポジトリ名が microsoft/WSL に変更されている。Type 2 は 2014 年 7 月の kaiinui/android-awesome-libraries であった。Type 3 は 2012 年 5 月の caesar0301/awesome-pcaptools であった。このリポジトリは pcap ファイルを分析する際に用いられる Linux コマンドや各ツールをまとめたリポジトリである。pcap ファイルとは、コンピュータネットワーク管理の分野におけるパケットスニファに用いられるファイルのことである。

6. まとめと今後の課題

本研究では GitHub 上にありながら Git の利用を主体としないプロジェクトを Hub プロジェクトと定義し、3 つの RQ を設定した上で分析を行った。以降、3 つの RQ とそれに対する回答を述べる。

RQ1: Hub プロジェクトにはどのようなものがあるのか
Hub プロジェクトは大きく分けて以下の 3 タイプに分類できる。

Issue Tracking Type (Type 1)

Issue 機能の利用を主な目的としたリポジトリ

Document Type (Type 2)

ドキュメント (Wiki を含む) 公開を目的としたリポジトリ

Link Collection Type (Type 3)

他のリポジトリや外部サイトへのリンク集を公開目的としたリポジトリ

RQ2: Hub プロジェクトの件数は経年変化しているのか

2016 年まではおおむね増加傾向にあった。特に 2014 年～2016 年頃に急激に増加しているが、2016 年を境に停滞・減少しつつある。

RQ3: RQ2 にて変化が見られた場合、その変化の原因は何か

Type 1 及び 3 において急激な増加のきっかけとなるプロジェクトの存在が示唆された。具体的には、Type 1 では microsoft/WSL が、Type 3 では sindresorhus/awesome が急激な増加のきっかけとなるプロジェクトであることが示唆された。

これらの結果から、今日 GitHub がソースコードのホスティングだけではなく多様な目的で利用されていることが分かった。今後も GitHub をソースコードのホスティング以外の目的で利用するプロジェクトが現れる可能性もあり、引き続き調査の対象となりうると考えられる。本研究における問題点として、本研究の抽出条件では漏れた Hub プロジェクトが存在する。抽出条件の精査は今後の課題である。また 2016 年を境に Hub プロジェクトは停滞・減少傾向にあるが、この原因の 1 つとして、特に Type 3 において新たなプロジェクトの出現の余地がなくなった可能性がある。この仮説の検証には、既存の Type 3 プロジェクトの更新頻度の調査が必要と考えられる。

今後の課題として、他の競合サービスを分析し、比較していくことが考えられる。また分析上の問題として今回用いた手法ではリポジトリの抽出精度・速度に難があったと感じている、これには SQL スクリプトの改良が必要である。

謝辞 本研究の一部は JSPS 科研費 17H00731 による助成を受けた

参考文献

- [1] Bacchelli, A. and Bird, C.: Expectations, outcomes, and challenges of modern code review, *2013 35th International Conference on Software Engineering (ICSE)*, IEEE, pp. 712–721 (2013).
- [2] GitHub Inc.: GitHub: The world's leading software development platform, <https://github.com>.

- [3] Dabbish, L., Stuart, C., Tsay, J. and Herbsleb, J.: Social coding in GitHub: transparency and collaboration in an open software repository, *Proceedings of the ACM 2012 conference on computer supported cooperative work*, pp. 1277–1286 (2012).
- [4] Thung, F., Bissyande, T. F., Lo, D. and Jiang, L.: Network structure of social coding in github, *2013 17th European conference on software maintenance and reengineering*, IEEE, pp. 323–326 (2013).
- [5] Vasilescu, B., Van Schuylenburg, S., Wulms, J., Serebrenik, A. and van den Brand, M. G.: Continuous integration in a social-coding world: Empirical evidence from GitHub, *2014 IEEE international conference on software maintenance and evolution*, IEEE, pp. 401–405 (2014).
- [6] Gousios, G., Pinzger, M. and Deursen, A. v.: An Exploratory Study of the Pull-Based Software Development Model, *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, New York, NY, USA, Association for Computing Machinery, p. 345–355 (online), DOI: 10.1145/2568225.2568260 (2014).
- [7] Microsoft Corporation: A repository to make changes to the TypeScript Wiki on GitHub, <https://github.com/microsoft/TypeScript-wiki>.
- [8] Microsoft Corporation: WSL: Issues found on WSL, <https://github.com/microsoft/WSL>.
- [9] Sorhus, S.: Awesome lists about all kinds of interesting topics, <https://github.com/sindresorhus/awesome>.
- [10] Gousios, G.: The GHTorrent dataset and tool suite, *Proceedings of the 10th Working Conference on Mining Software Repositories*, MSR '13, Piscataway, NJ, USA, IEEE Press, pp. 233–236 (online), available from <http://dl.acm.org/citation.cfm?id=2487085.2487132> (2013).