

OSS

MYTHS

AND

FACTS

OSSの神話と真実

2023/12/01

Contents

Chapter 1	Executive Summary	2
Chapter 2	Introduction	4
2.1	ソフトウェア開発におけるEmployee Experience (EX)	4
2.2	先人としてのOSS	4
2.3	OSSコミュニティはユートピアか	5
2.4	方法	5
Chapter 3	Myths	6
3.1	OSSコミュニティのコミュニケーションは緩やかである	7
3.2	OSSコミュニティは眠らない	9
3.3	OSSコミュニティは、終わるのも早い	11
3.4	OSSコミュニティは、クラッカーに負けない	14
3.5	OSSコミュニティは要求に素早くこたえる	16
3.6	OSSコミュニティの参加者はトップ開発者だ	18
Chapter 4	提言	20
4.1	OSSはコミュニティには「だれもが」参加できているか	20
4.2	OSSはコミュニティには「いつでも/どこでも」参加できているか	20
4.3	OSSにおけるプロダクトマネジメント	21

Chapter 1

Executive Summary

私たちはOSSコミュニティに関する6つの神話をピックアップし、真であるかを否かを検証しました。

企業や組織におけるソフトウェア開発でのEmployee Experience (EX) を実現するために、OSSコミュニティの開発スタイルから学ぶことは何か、あるいは、取り組むべき課題が何かを明らかにするためです。OSSコミュニティはハッカーと呼ばれる優れた開発者たちにリードされ発達してきました。私たちは、OSSコミュニティのエンジニアと活動を尊敬し、多くのことを学びたいと考えています。

一方で、私たちは、何かに対する強い期待が、誤解を生じさせる可能性があることを知っています。特に当事者以外が、実践者らのプラクティスを伝聞で理解しようとする時、過度な期待や懸念が誤解＝神話を形成することがあります。

私たちは、文献調査とインタビューをもとに検証する神話をピックアップしました。ここでの神話を抱く人々は、OSSコミュニティの直接的参画者ではない、ソフトウェア開発のマネージャー層や顧客たちです。そして、1) 私たち自身によるリポジトリデータの分析、2) 先人たちによるデータ分析の文献サーベイ、あるいはその組み合わせにより、各神話についての質問に答えを得ました。

Myth: OSSコミュニティのコミュニケーションは緩やかである

Question: OSSコミュニティでは、コミュニケーションの応答に時間がかかるか？

Fact: 議論の種類によらず、約半分のコミュニケーションは4時間以内でコミュニケーションのやりとりをしている。

Myth: OSSコミュニティは眠らない

Question: OSSコミュニティでは、開発者の活動時間帯は分散しているか？

Fact: 開発者の活動時間帯は、北米のオフィスアワーに偏っている。

Myth: OSSコミュニティは、終わるのも早い

Question: OSSコミュニティはどの程度継続するか？

Fact: 採用から4年後もOSSコミュニティ活動の半数は、継続している。

Myth: OSSコミュニティは、クラッカーに負けない

Question: OSSコミュニティでは、脆弱性（セキュリティホール）の解決にかかる時間が短いか？

Fact: 脆弱性の解決時間3ヶ月とは必ずしも短くはない.

Myth: OSSコミュニティは要求に素早くこたえる

Question: OSSコミュニティでは、バグ報告や追加要求の解決にかかる時間が短いか？

Fact: バグ報告や機能追加要求は解決時間は多くは2週間以内と短い，が，少なくとも $1/4$ は3ヶ月以上かかるなど解決期間のばらつきが大きい.

Myth: OSSコミュニティの参加者はトップ開発者だ

Question: OSSコミュニティにおける開発者たちの役割にはどのようなものがあるか？

Fact: バリエティに富んだ役割がOSSには存在する.

Chapter 2

Introduction

2.1 ソフトウェア開発におけるEmployee Experience (EX)

私たちは、ソフトウェア開発に携わる人々のWell-beingを実現、Employee Experience (EX) の向上を目指し研究開発をしています。実現アプローチのひとつとして、だれでも、いつでも、どこからでも参加できるソフトウェア開発により、多様な人材の活躍の機会を拡大したいと考えています。COVID-19によって推進が後押しされたリモートワークは、働く場所の拡大による働く条件の拡大としても期待されています。多くの企業ではリモートワークの導入目的として、「勤務者のゆとりと健康的な生活の実現」、「通勤弱者への対応」、「優秀な人材の雇用確保」と、従業員の働きやすい職場の実現に関する項目を挙げています。就業者は、リモートワークのメリットとして、通勤時間・移動の削減や自由に使える時間の増加、育児・子育てと仕事の両立を回答しています。

従来、ソフトウェア開発、特に企業におけるソフトウェア開発では、働く場所が限定されていました。ソフトウェア開発の依頼者および開発者同士のコミュニケーションがとりやすいことが良いソフトウェア開発につながるからという面と、企業にとって、働く場所を固定した方が情報および人員の管理をしやすいという面があります。

2.2 先人としてのOSS

「インターネットのかぼそい糸だけで結ばれた、地球全体に散らばった数千人の開発者たちが片手間にハッキングするだけで、超一流の OS が魔法みたいに編み出されてしまう」-伽藍とバザールより-

OSSは、利用者の目的を問わずソースコードを使用、再利用、再配布が可能なソフトウェアの総称です。同時に、頻繁なリリースなどの開発スタイルや、インターネットによって緩やかに結びついたコントリビュータたちによる協働など、開発への関わり方の先進性においても注目を集めてきました。多くの価値のあるソフトウェアが、世界各地の開発者の協力により生み出されているのです。これらの開発から、Big techも学ぶべきものがあると我々は注目しています。OSSコミュニティが成長するにつれて、ソフトウェア開発の依頼者やマネジメント層も、OSSコミュニティのあり方やプロダクトマネジメントに関する関心を持つようになってきました。

2.3 OSSコミュニティはユートピアか

私たちは、OSSのコミュニティのあり方、あるいは開発者のプロジェクトへの関わり方についてのさまざまな意見を耳にします。ソフトウェア開発の依頼者やマネジメント層がもつ意見は、OSSコミュニティの成り立ちやなどからくる印象論によるものが多いことに注意する必要があるでしょう。実践者たちは、自身の経験則から学びを得ますが、それ以外の人々は経験則を伝聞し、時に先入観をもって解釈します。結果、過度な期待を持つことで失望したり、強すぎる懸念を持つことの要因になります。例えば、「アジャイル」という言葉は、その言葉を自身の活動の中から汲み出した開発者と手段として注目するマネジメント層では捉え方が異なります。

2.4 方法

1. Topicsの選定

1. ソフトウェア開発の依頼者やマネジメント層のコメント収集

- 文献調査
- インタビュー

2. 関心ごとや背景を整理

3. 3つの分類とtopics

2. 各Topicsの検証

1. 1次調査文献調査

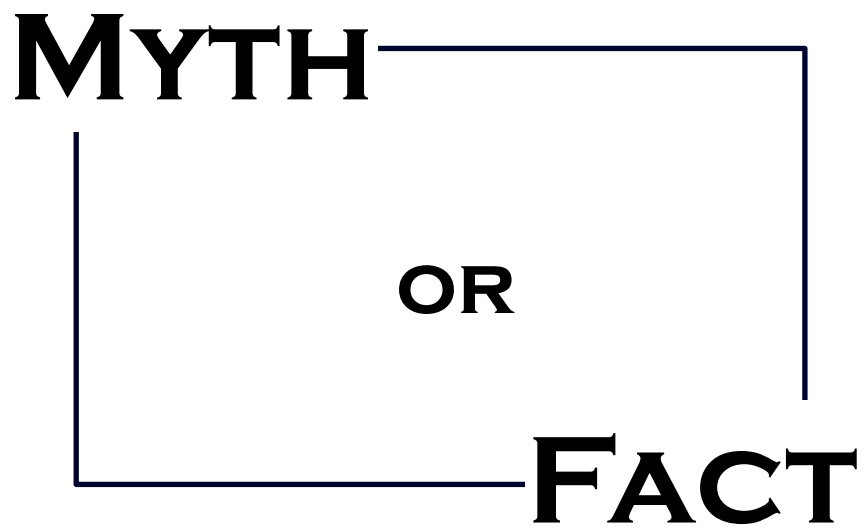
1. 文献調査
2. 判定
 - すでに論文などで議論済みなTopicsであれば、そのサマ리를記載する
 - 未議論あるいは追加調査可能なTopicsは、データ分析を行う

2. 2次調査データ分析

- 主な対象データセット
 - libraries.io open data
- 分析対象の抽出手順
 1. データセットに含まれるプロダクトをリストアップする
 2. GitHubリポジトリから取得できるものに限定する
 3. フォークリポジトリを除外する
 4. 各分析に合わせてフィルタリングする
 5. ランダムサンプリングする

Chapter 3

Myths



3.1 OSSコミュニティのコミュニケーションは緩やかである

Question: OSSコミュニティでは、コミュニケーションの応答に時間がかかるか？

ソフトウェア開発はコミュニケーションの協調ゲームであり、開発者間のやりとりの速度がプロジェクトの進捗に影響すると言う。一般的なソフトウェア開発では、開発者同士の距離を近づけることが良いプラクティスと言われる[1]。OSSコミュニティでは、開発者が同じ場所（オフィス）、かつ、同じ時間（オフィスアワー）に集まることはない。このような条件下では、相手のタイミングを見てコミュニケーションを開始することは難しく、メッセージに対する回答に時間がかかる緩やかなコミュニケーションが行われているのではないだろうか。神話を検証するために、OSSコミュニティでのコミュニケーション間隔を分析する。

Fact: 議論の種類によらず、約半分のコミュニケーションは4時間以内でコミュニケーションのやりとりをしている。

GitHubにおいて長く継続しているOSSでの、コミュニケーションの間隔を、Figure1は、課題チケット(issue)に関するやり取り、あるコメントから、その返答がなされるまでの時間間隔の分布を示している。Figure2は、プルリクエスト(pull request)のやりとりを同じく示している。これらFigureによれば、issueのやりとりは中央値で4時間程度、Pull Requestのやりとりでは中央値で40分を切り、ごく短い時間間隔でやり取りが行われていることがわかる。OSSコミュニティにおいて、素早いコミュニケーションが行われている[2]。

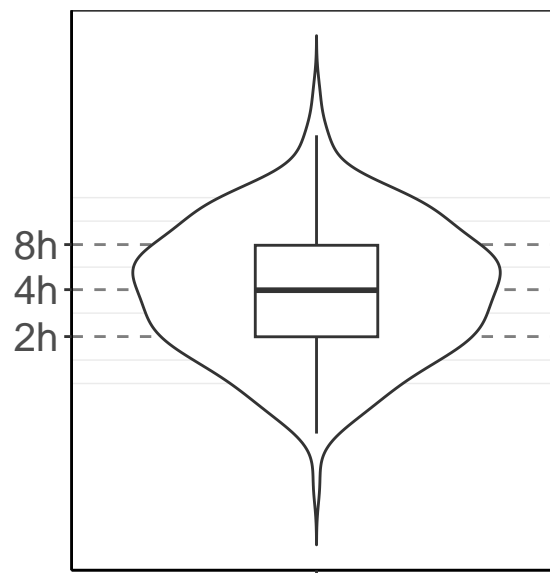


Figure1 The distribution of the time interval between a comment on an issue ticket

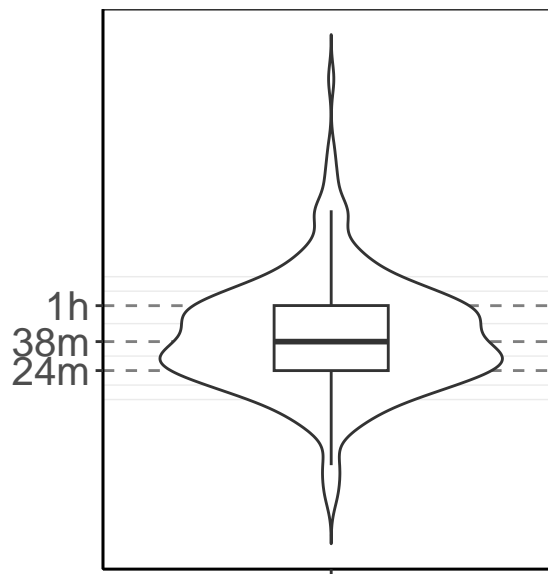


Figure2 The time interval distribution for a pull request

時間間隔の短く返答がなされるのは、非同期コミュニケーションであったとしても、議論自体は同期的に行いたいという開発者自身の希望による可能性がある。例えば、マイクロソフトの調査[2]において、開発者は短時間で多数のやり取りを行うコミュニケーションを好むという結果が報告されている。一方で、返答を素早く返すこと、同期的なコミュニケーションに参加し続けることが、開発者へのプレッシャーになっている可能性がある。この場合、開発者は、コミュニケーションを行っていない間でも、いつでも返答が返せるようにずっと拘束されていると感じ、well-beingに悪い影響を与える可能性がある。

Insight:

OSSコミュニティでも時間間隔の短いコミュニケーションが行われていることがわかった。リモートワークなど多様な働き方を認める時代において、企業のソフトウェア開発で、同期的なコミュニケーションが求められた場合、開発者のWell-beingを低下させる可能性がある。ゆるやかなコミュニケーションでソフトウェア開発に参画できるのであれば、開発者がそれぞれのライフスタイルに合わせて働く時間が選択できるなど様々な利点が得られる可能性がある。

References

1. Cockburn, Alistair, “Agile software development: the cooperative game.”, Pearson Education, 2006.

3.2 OSSコミュニティは眠らない

Question: OSSコミュニティでは、開発者の活動時間帯は分散しているか？

一般的な分散開発では、開発者間の時差や文化的違いにより、一部の開発者が“忘れられ”あるいは“無視され”た結果、混乱と意欲喪失につながる可能性があると言う。OSSコミュニティには世界中から開発者が参加している[1]。OSSコミュニティでは、開発者自身の住む地域のタイムゾーンで、時差を超えて活動できているのだろうか。だとすれば、いろいろなタイムゾーンに合わせて開発者の活動時間帯は分散し、OSSコミュニティ全体としては活動が絶えず行われているはずである。神話を検証するために、OSSコミュニティの開発者たちの活動時間帯の分布を分析する。

Fact: 開発者の活動時間帯は、北米のオフィスアワーに偏っている。

OSSコミュニティの開発者たちの活動時間帯の分布を、Figure1は、課題チケット(issue)へのコメント量を、コメントがつけられた時刻を軸としてプロットしたものである。時刻は、各開発者のタイムゾーンではなく、最も遅いタイムゾーンである「Anywhere on Earth: AoE」時間である。図から、AoE時間でのオフィスアワー(9時から17時)に活動が集中していることがわかる。Figure2は、プルリクエスト(pull request)へのコメント量を同じようにプロットしたもので、こちらもAoE時間のオフィスアワーの時間帯に活動が集中している。

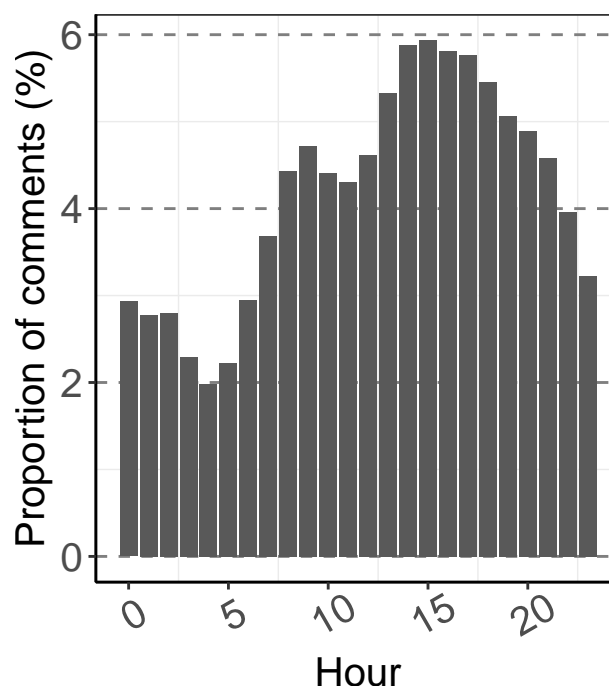


Figure1 The count of comments on an issue ticket

活動時間帯の偏りは、開発者の数、または活動量が多い開発者の数が、活動地域により偏っている可能性を示す。Githubによれば、2015年当時は3分の1が北米からの開発者であったが、近年インド、中国その他の

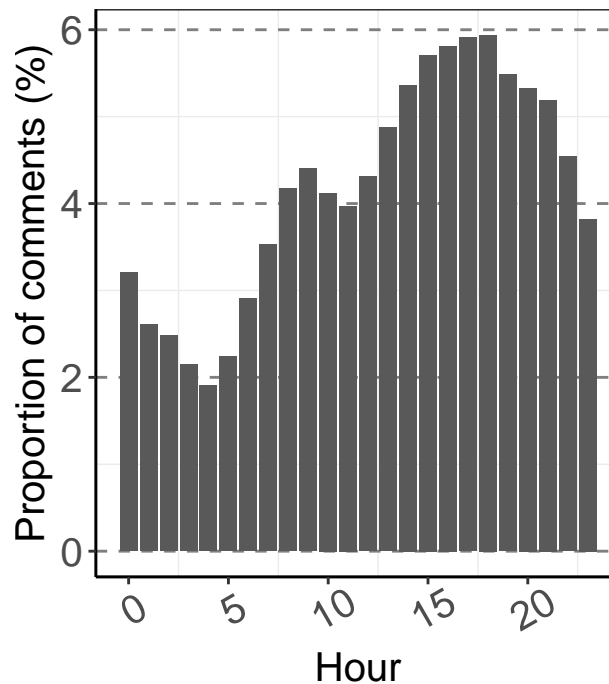


Figure2 The count of comments on pull requests

国・地域からの参加が著しく増加しているという[1]。あるいは、開発者がどこに住んでいるかにかかわらず、AoE時間に合わせて活動している可能性がある。例えば、インドの開発者がAoE時間帯のオフィスアワーにあわせると、深夜から早朝にかけて活動することになる。時差の影響を受けないように、他のタイムゾーンに合わせて活動していることを示唆している可能性がある。

Insight:

非同期活動の最前線であるOSSコミュニティでさえ、ソフトウェア開発では、同期的活動が求められている。一方、パンデミックの流行等様々な問題が発生する現代社会においては、企業の開発を止めないためにも非同期でソフトウェア開発を行う方法を考えていく必要がある。一つの可能性としては、生成系AIをベースとしたエージェントを使ったHuman-in-the-loopの仕組みを確立することは、開発者の負担軽減につながる可能性がある。

References

1. GitHub, “A global community of developers”, <https://octoverse.github.com/2022/global-tech-talent>

3.3 OSSコミュニティは、終わるのも早い

Question: OSSコミュニティはどの程度継続するか？

著名なOSSのWebサーバやDBサーバが、OSSコミュニティにより長年にわたって継続的に開発されていることは事実である。伽藍とバザールでは、「ソフト開発で重要なのは長期にわたり開発努力が継続することだがOSSは継続しない」と考えるソフトウェア開発マネージャーたちの議論を取り上げている[1]。GitHubでは、2021年に新しいリポジトリが8,570万以上作成され、年20%増えている[2]ように、OSSコミュニティは次々と生まれている。年単位で観察したときに、活動が終了するOSSコミュニティはどの程度なのだろうか神話を検証するために、OSSコミュニティの活動の継続可能性を分析する。

Fact: 採用から4年後もOSSコミュニティ活動の半数は、継続している。

Figure1は、活動中OSSの継続活動期間の確率を示す。Figure1より、最初の3ヶ月の確率の減少量が他の期間より大きく、全体の傾向は、線形に減少している。活動中OSSの半数が活動を停止するまでの期間は48ヶ月である。Figure2は、ある特徴量が、OSSの活動の継続/停止に影響を及ぼす程度を示す。縦軸は特徴量の影響の度合いを示し、負の値であれば活動の継続に、正の値であれば活動の停止に影響を及ぼすとみなせる。横軸は特徴量の重要度を示し、重要でないものは早くに0に近づく。Figure2より、OSSの活動の継続に強く影響を及ぼすものとして、開発者の住んでいるタイムゾーンの数、OSSの最上位ディレクトリ数が挙げられる。

最初の三ヶ月の確率の減少が他の区間と比較して激しいのは、現在活動しているOSSという母集団において、直近に作成されたOSSがある程度の割合を占め、それらの多くはすぐに活動を停止してしまうためである。開発者の住んでいるタイムゾーンの多様性がOSSの継続性に最も影響を及ぼすという結果は、OSSコミュニティの多様性がOSSの継続性に寄与することを示唆する。また、最上位ディレクトリ数については、これからも活動を長く継続するOSSは規模が大きい場合が多いと考えられ、その結果が反映された可能性がある。

Insight:

長く続いているOSSは多くの多様な人々に愛されていると考えられる。データからも、長く継続するOSSでは、開発者の居住地の多様性が高いことが確認できた。参加者の多様性と、長続きするプロジェクトの間の相関を示している可能性がある。企業における開発でも、開発関係者の多様性を高くすることで、開発者間での学びや、異なる視点を取り入れることによるプロジェクトの質向上につながり、最終的に多くの人に愛されるプロダクトにつながる可能性がある。

References

1. GitHub, “The 2021 State of the Octoverse”, <https://octoverse.github.com/>
2. METI Ministry of Economy, A collection of case studies on management methods for utilizing OSS and ensuring its security, <https://www.meti.go.jp/policy/netsecurity/wg1/CollectionOfUseCaseExamplesForUtilizi>

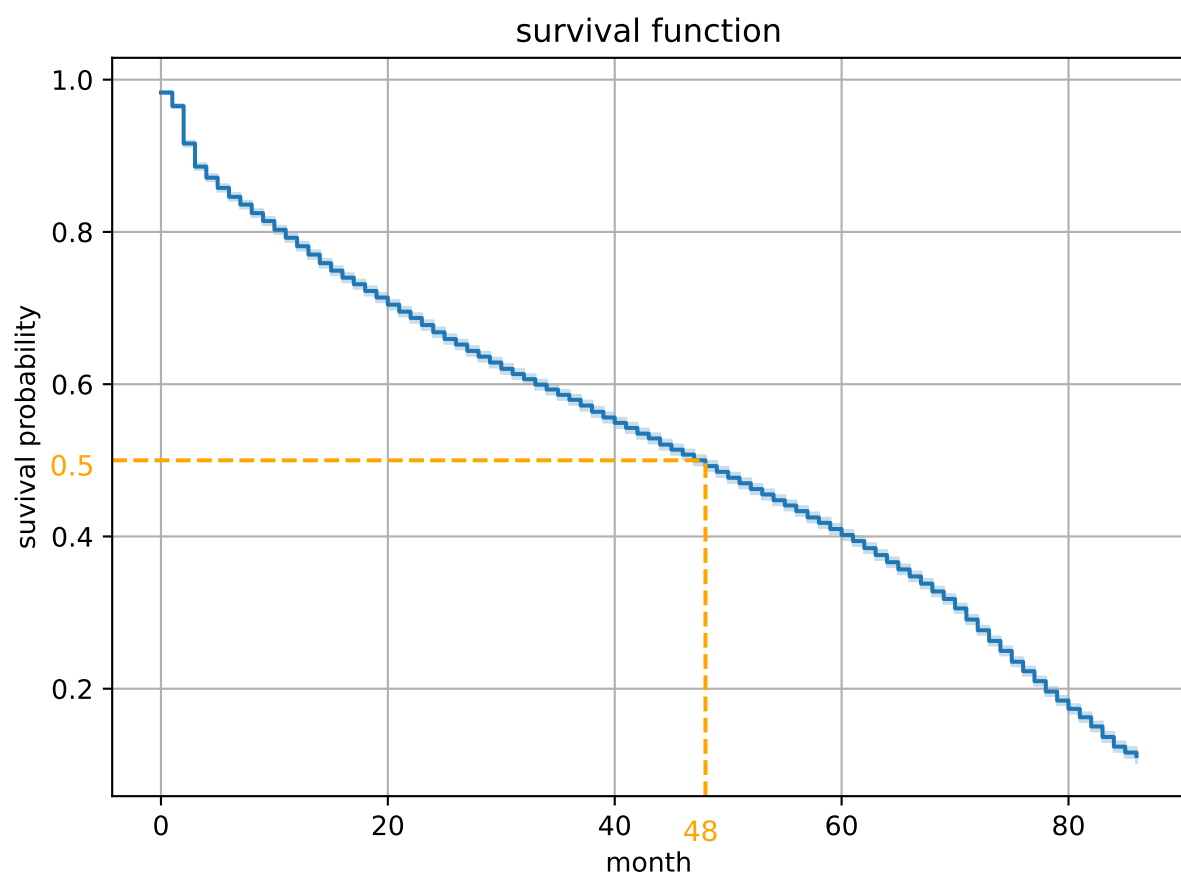


Figure1 The probabilities for the duration of OSS survival time

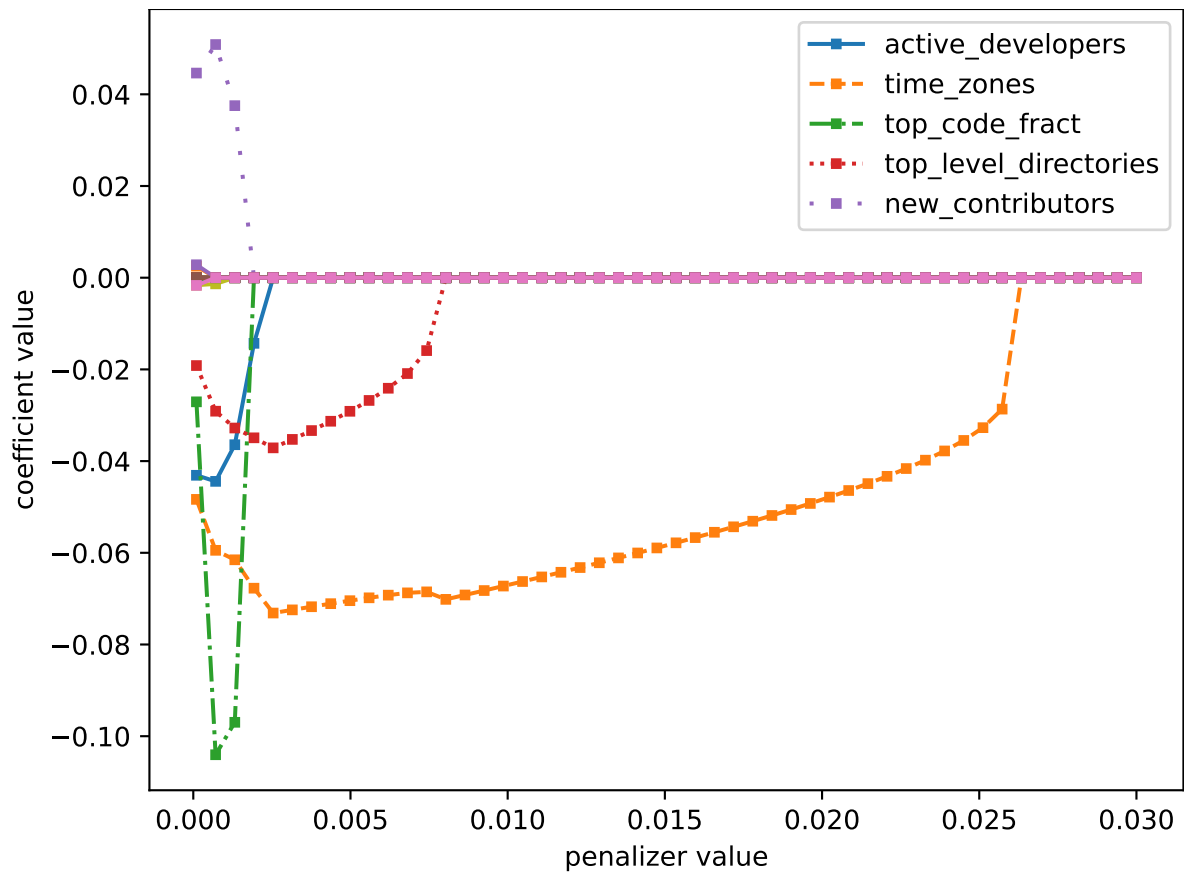


Figure2 The impact of different features on the survival or cessation

3.4 OSSコミュニティは、クラッカーに負けない

Question: OSSコミュニティでは、脆弱性（セキュリティホール）の解決にかかる時間が短いのか？

伽藍とバザールは、リーナスの法則として「十分な数の目があれば、全てのバグは発見される」、つまり、十分な数の開発者がいれば、全てのバグはすぐに発見され修正されるため、ソフトウェアの品質を高く保つことができると述べている。[1] OSSはソースコードが公開されており、OSSコミュニティには世界中の開発者が参加している。OSSに含まれる全てのバグはすぐに発見され、すぐに修正されると期待してよいのであろうか。神話を検証するために、いくつかのOSS、また、一般的なバグと比較して時間がかかると考えられる脆弱性は改修に焦点を当て、改修にかかる期間を分析する。

Fact: 脆弱性の解決時間3ヶ月とは必ずしも短くはない。

OSSにおいて開発者が多いApacheの2つのプロジェクトHadoopとhttpdにおける、脆弱性への対応速度を分析する。Figure1で、x軸は修正までにかかる日数、y軸は修正された脆弱性の数を示す。Figure1より、脆弱性は概ね100日以内には修正されていることがわかる。中央値はどちらのプロジェクトでも87日程度（3ヶ月弱）である。図のサイズの関係でx軸の上限を500日としているが、実際には修正までに上限を超えた長い期間がかかっている脆弱性も存在した。例えば、httpdで最も長い修正期間を要した脆弱性は、解決までの期間が1,842日（5年強）である。

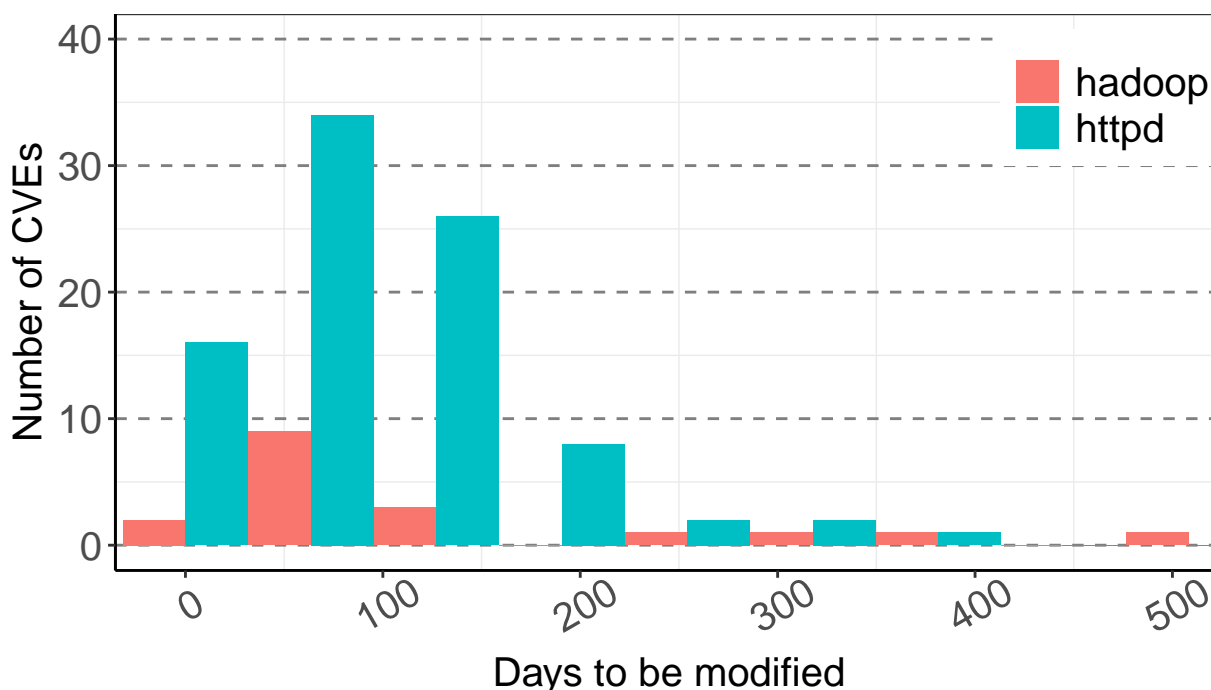


Figure1 The response time to vulnerabilities in Hadoop and httpd

開発者数が多いプロジェクトにおいても、品質に大きな影響を与える脆弱性への対応に3ヶ月弱の時間がかかる事実は、OSSにおけるバグへの対応に時間がかかる可能性を示す。また、例外的なケースではあるが、5年以上放置された脆弱性がある点にも注意が必要である。脆弱性のような解決が難しいバグの場合、多くの人の目だけでは、解決時間に影響しない可能性がある。ここまでの分析ではバグが発見され報告されてからの時間を調査しているため、バグ発見までの時間が含まれていない。先行研究では、バグが混入してから修正までにかかる時間の中央値が200日であるという報告があり[1]、我々の結果を合わせると、発見までに100日程度かかる可能性がある。

Insight:

企業では、リリース後のコードの改修に備えコードの知見を持った開発者を引き止めるケースがある。OSSコミュニティの、脆弱性修正までの時間が極端に早いわけではない。しかし、OSSコミュニティでは開発者の入れ替わりが激しく、脆弱性を混入した開発者と修正する開発者が異なる可能性がある中でも脆弱性を修正している。企業は、OSSコミュニティから、属人性を排した形で脆弱性対応を行うプラクティスを学べる可能性がある。

References

1. Raymond, Eric. “The cathedral and the bazaar.” Knowledge, Technology & Policy 12.3 (1999): 23-49.
2. Kim et al., How long did it take to fix bugs?, Proc. of MSR2006

3.5 OSSコミュニティは要求に素早くこたえる

Question: OSSコミュニティでは、バグ報告や追加要求の解決にかかる時間が短いのか？

伽藍とバザールは、「ソフトウェアを使うユーザを共同開発者として扱うことで、コードの改良を高速化し、デバッグ効率化できる」と述べている。OSSコミュニティでは、誰もがバグの報告やコードの修正に貢献できる。また、OSSの特徴として「変更」に重点が置かれていることがあげられる[1]。これらから、OSSは、どのような要望でもすぐに反映されるのだという期待を持つ人々がいる。OSSコミュニティに対するバグや機能追加の要求は、素早く対応されるのだろうか。神話を検証するために、OSSコミュニティによる開発で、バグ報告と機能追加の2種類のユーザの要望が解決されるまでの期間を分析する。

Fact: バグ報告や機能追加要求は解決時間は多くは2週間以内と短い、が、少なくとも1/4は3ヶ月以上かかるなど解決期間のばらつきが大きい。

Figure1は、ユーザからのバグ報告に対する対応時間を、箱ひげFigureおよびヴァイオリンプロットで表している。バグ報告への対応時間の中央値が10日、75パーセンタイルが117日である。Figure2は、同じく機能追加要求に対する対応時間の分布をプロットしている。機能追加要求に対する対応時間の中央値は14日、75パーセンタイルは146日である。バグ報告、機能追加要求のそれぞれ半数は、2週間以内に解決されるが、バグ報告の25%は、解決までに3ヶ月以上、機能追加要求の25%は4ヶ月以上の時間がかかっている。

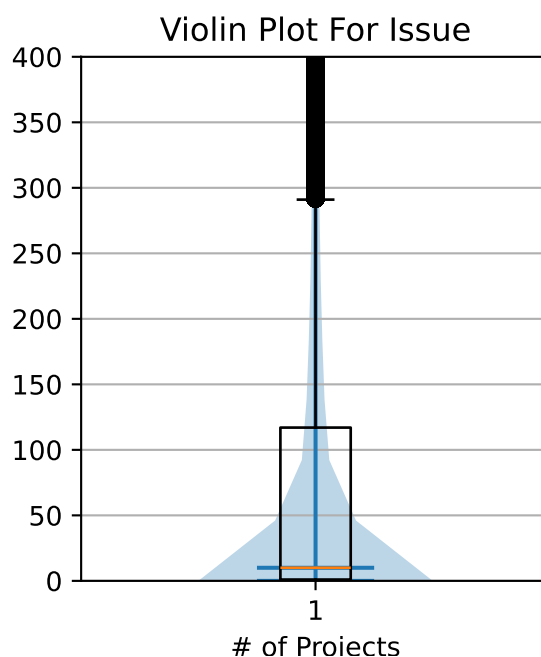


Figure1 The response times in bug reports

バグ報告も機能追加要求も半数程度が2週間程度で解決されるにも関わらず、3/4の解決に4ヶ月以上というより長い時間がかかる。この結果は、バグ報告や機能追加要求を行うユーザ全てが開発者として開発に貢献す

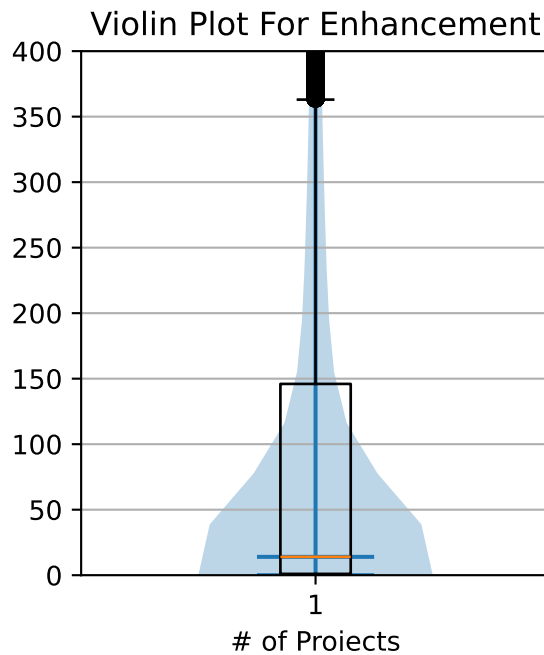


Figure2 The response times in feature requests

るわけではなく、バグ報告や機能追加要求量に対して、開発者数が限られていることの可能性を示す。また、多量のバグ報告・機能追加要求は、その解決にかかる労力がそれぞれに異なる。全く議論されずにクローズされるものから、多数の議論を経て、解決されるものもある。限られたリソース（開発者）で解決できる要求には限りがあり、要求のトリアージが重要となる。

Insight:

OSSという世界中の開発者を集められる開発体制だとしても、報告されるすべての要求を直ちに満たすことは難しい。今回の結果はその実情を反映していると考えられる。限られた対応能力で最善を尽くすためには、要求の適切なトリアージが重要となってくる。適切に要求がトリアージされれば、開発者の負荷も適切な範囲でコントロールできる。先行するOSS開発を対象としたトリアージ方法研究から、技術を導入することは企業における開発を最適化することにつながる。

References

1. Raymond, Eric. "The cathedral and the bazaar." Knowledge, Technology & Policy 12.3 (1999): 23-49.

3.6 OSSコミュニティの参加者はトップ開発者だ

Question: OSSコミュニティにおける開発者たちの役割にはどのようなものがあるか？

OSSコミュニティには著名なハッカーたちが大勢いる。近年では、開発者からも、企業の採用担当者からも、OSS開発者としての才能が採用や高い報酬につながると考えられている[1]。これらは、OSSコミュニティに参加するエンジニアはスキルが高い、トップ開発者であるとのイメージをもたらしている。GitHubを利用するエンジニアは2023年1月時点で、1億人を超え[2]、OSSコミュニティは拡大している。OSSコミュニティは、トップ開発者ばかりで構成されるのだろうか。神話を検証するために、OSSコミュニティに参加する開発者の役割について分析する。

Fact: バラエティに富んだ役割がOSSには存在する。

OSSコミュニティにおける開発者の役割およびコミュニティの形態に関する先行研究の調査を行った。先行研究では、コミュニティの形態として、コアチームの存在が仮定・観察されている場合が多い[2]。開発者の役割の分類は研究により様々であるが、コアとなって働く開発者（コア開発者）、一部の機能追加等限定的に参加する開発者（エピソード開発者）、新規で入る開発者（新規開発者）等が仮定・観察されている[3,4]。また、コメントに対するスタンプなどリアクションのみを行うリアクター[5]が存在するなど、企業での開発と異なる、バラエティに富んだ役割がOSSには存在する（Figure1）。

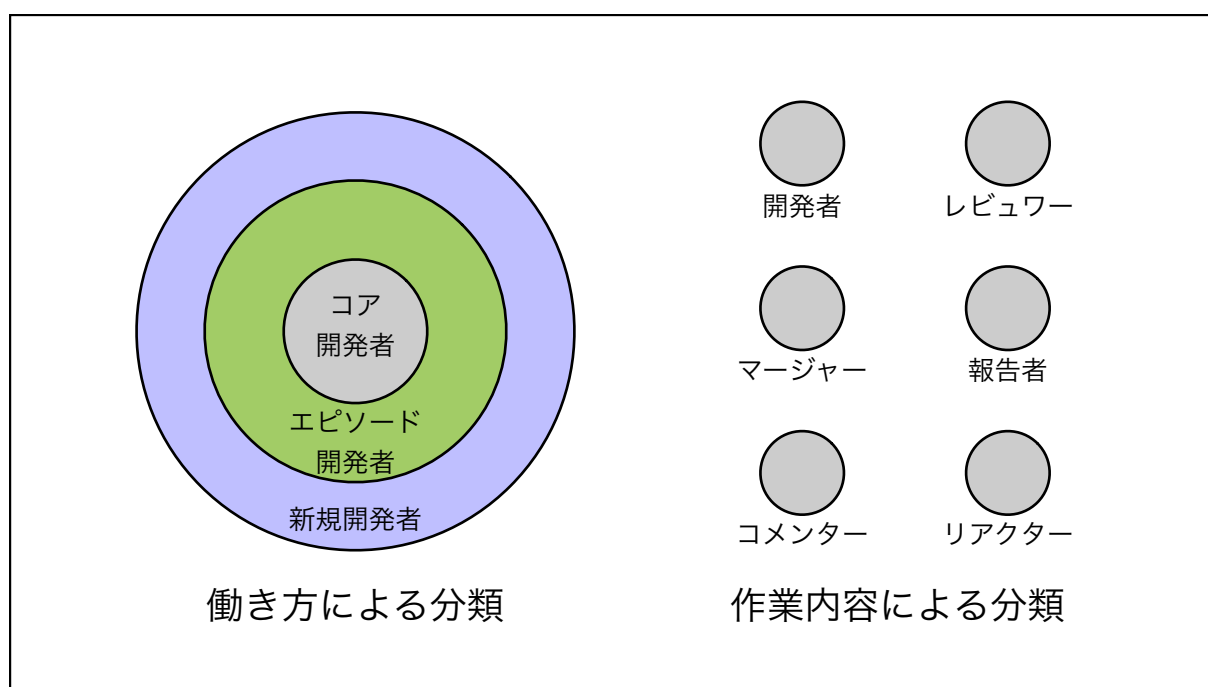


Figure1 The roles in OSS / corporate development

一部のOSSコミュニティでは、ドキュメントの整備や軽微なバグ修正などにGood First Issueのラベルを付け見つけやすくすることで、開発経験は少ないがコミュニティへの参加意欲がある開発者が、コミュニティに

参加しやすくしている。また、リアクターは、対象が機能の設計等ではなく、ある機能が欲しいかであれば、自身の技術力によらずリアクションが可能である。OSSコミュニティの開発者の役割は多様にあり、OSS開発に参加した事実が、開発者の技術力を保証するわけではない。OSSコミュニティに参加するすべての開発者がトップ開発者ではない。

Insight:

企業においてはグリーンベレー人材といった優秀な開発者のみでチームを構成することが着目された時期がある。一方、OSS開発に参加する開発者はスキルも貢献の仕方も多種多様である。コーディングやドキュメンティングのみならず、議論に対してリアクションを行う役割といった、企業では重要視されていない役割が存在することがわかった。OSSにおける役割の多様性から、企業は新しい価値観を学ぶことができる可能性がある。

References

1. edX and Linux Foundation Research, “10th Annual Open Source Jobs Report”, <https://training.linuxfoundation.org/open-source-jobs-report/>
2. GitHub, “100 million developers and counting”, <https://github.blog/2023-01-25-100-million-developers-and-counting/>
3. Robles et al. “Evolution of the core team of developers in libre software projects”, MSR2009
4. Barcomb et al. “Uncovering the Periphery: A Qualitative Survey of Episodic Volunteering in Free/Libre and Open Source Software Communities”, TSE, 2020.
5. Yue et al. “Off to a Good Start: Dynamic Contribution Patterns and Technical Success in an OSS Newcomer’s Early Career”, TSE, 2022.
6. Cánovas et al. “On the analysis of non-coding roles in open source development”. *Empir Software Eng* 27, 18 (2022). <https://doi.org/10.1007/s10664-021-10061-x>

Chapter 4

提言

3章では、6つの神話それぞれについて、真偽を判定し、またそこから得られるinsightは何かを議論しました。本章では、これらの神話をいくつかの視点にまとめ、企業や組織におけるソフトウェア開発でのEmployee Experience (EX) を実現するために、OSSコミュニティの開発スタイルから学ぶことは何か、あるいは、取り組むべき課題が何かを議論します

私たちの、大きな関心は、だれでも、いつでも、どこからでも参加できるソフトウェア開発による多様な人材の活躍の機会の拡大です。世界中の開発者がOSSコミュニティに参加していますが、依然として幾つかのハードルがあります。

4.1 OSSはコミュニティには「だれもが」参加できているか

OSSでの実態：「OSSコミュニティの参加者はトップ開発者だ」の検証から、貢献の役割はコアチームとその他に分かれていると考えられます。全てのOSSコミュニティではありませんが、参画コストを低減するための工夫がなされているようです。「だれもが」ではありませんが、それに向けた工夫が行われています。

課題：ソフトウェア開発に必要な知識・スキルは多様化しています。いずれかの分野に特化したスキルよりも、フルスタックエンジニアと呼ばれるような、幅広い知識・スキルを備えたエンジニアが期待されています。開発者としてプロジェクトに参画して活躍できるまでの道のりが長くなっています。

提言：OSSの先例も参考にしながら、各人が保有する知識・スキルを活かしてプロジェクトに参画でき、また、参画により実践的なスキルアップを実現するためのプラクティスを見出すことが必要でしょう。コーディングやテスト以外の議論へのリアクションなどがソフトウェア、開発プロジェクトにどのような影響を与えているかの分析は新しい価値観の発見につながる可能性があります。

4.2 OSSはコミュニティには「いつでも/どこでも」参加できているか

OSSでの実態：「OSSコミュニティは、終わるのも早い」から、プロジェクトの継続に参加者のタイムゾーンが多いことが寄与する可能性が示されました。OSSコミュニティには世界中の開発者たちが参加しています。「OSSコミュニティは眠らない」「OSSコミュニティのコミュニケーションは緩やかである」から、活動時間は、特定の時間帯に集中しており、コミュニケーション間隔の短い同期的なコミュニケーションがおこなわれていることが観察されました。ネットワークを介することで「どこでも」は比較的实现されているようですが、「いつでも」とは限らないようです。

課題：ソフトウェア開発プロジェクトでは、コミュニケーションを取りやすくするため開発者同志の距離を近づけることがプラクティスとしてよく取りあげられます。企業では、ソースコードや設計に含まれる機密情報の管理のため開発者の作業場所を固定していることが多くあります。ネットワークやツールの発達はある物理的な場所の制約を減らしましたが、いまだに制約は存在します。また、ある時間帯に行われるコミュニケーションに参加できない場合、ソフトウェア開発プロジェクトへに参画できない場合があります。

提言：コミュニケーションのタイムラグを超越する、情報管理を維持しつつ地理的な自由度を上げる開発スタイル、ツールを新たに確立することが必要でしょう。

4.3 OSSにおけるプロダクトマネジメント

OSSでの実態：「OSSコミュニティは要求に素早くこたえる」「OSSコミュニティは、クラッカーに負けない」から、OSSにおいて全ての要求やバグが速やかに解決されるわけではないことがわかりました。

課題：企業においてもバグや要求が古典的なパスのみからではなく、SNSやレビューなど多様なパスから大量に届くようになっている。それらの全てを解決するには開発者が不足しています。

提言：OSSの先例も参考にしながら、大量の要求やバグ報告の中から優先的・緊急で対応すべきバグや要求の選別方法やプラクティスを学び、ツール化を検討する必要があるでしょう。企業でのソフトウェア開発においても、クリティカルではないと判断した要求やバグについては、インナーソーシングなどこれまでとは異なる貢献を促すようなアプローチの検討が有効かもしれません。

Myths of OSS

2023/12/1 First edition issued
