3L-9

プロジェクト横断的な オープンソースソフトウェア開発記録の分析手法

An Analysis Method for Open Source Software Development Records across Projects

中村 高士　　早瀬 康裕　　北川 博之

Takashi Nakamura　Yasuhiro Hayase　Hiroyuki Kitagawa

筑波大学 / University of Tsukuba

1. 序論

近年，数多くの開発者によって，オープンソースソフトウェア（OSS）の開発が盛んに行われている．例えば，SourceForge.net [1] と呼ばれる OSS 開発支援サイトでは，32 万件以上の OSS 開発プロジェクトを運営されており，340 万人以上の開発者が開発に参加している．OSS 開発者は様々な目的で開発に参加しているが，開発者を管理する組織などは存在しないため，その全貌は不明である．

本研究の目的は，様々な OSS プロジェクトにおいて，OSS 開発者達がどのように活動しているかの実態を明らかにすることである．具体的には，分散型バージョン管理システムの特性に着目し，複数のプロジェクトにまたがって開発記録の分析を行うことで，開発者がプロジェクトにどのように貢献しているかを定量的に調査する．

2. 研究の背景

OSS 開発者とプロジェクトの関係を分類する基準の 1 つに，プロジェクトの配布物を直接変更できる権限を持つコア開発者と，その権限を持たない周辺開発者の 2 種類に分ける方法がある．[2] 一般に，1 つのプロジェクトにおいて，コア開発者の数は，周辺開発者に比べて非常に少ない．また，開発者は複数のプロジェクトに参加することができるため，ある開発者がプロジェクト A においてコア開発者であり，プロジェクト B では周辺開発者であるといった事が起こりうる．

一般に，OSS 開発プロジェクトでは，バージョン管理システム（VCS）と呼ばれる，ファイルのスナップショットを保存するシステムを用いて，成果物の管理を行う．新しいスナップショットを VCS のデータベース（リポジトリ）に保存する作業はコミットと呼ばれる．また，コミットには，コミットを行った開発者のユーザ名，日時，変更内容についてのメモといったメタデータが付随する．

VCS には様々な実装が存在するが，それらは集中型 VCS と分散型 VCS の 2 種類に分類される．集中型 VCS は古くから存在するタイプの VCS であり，1 箇所に置かれたリポジトリを複数の開発者で共有して使用するのが特徴である．一方，分散型 VCS

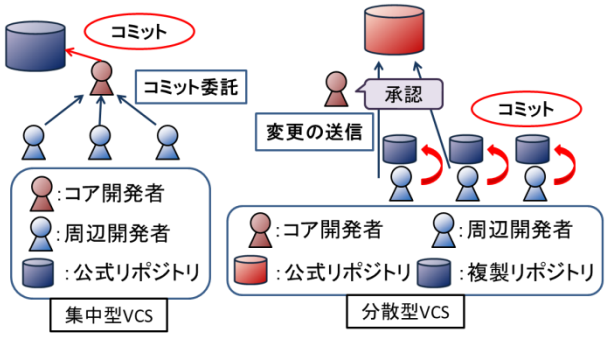

図 1 集中型・分散型バージョン管理システム

は相対的に新しい VCS であり，それぞれの開発者が 1 つ以上のリポジトリを所有し，リポジトリ間で変更内容をやりとりするという特徴がある．

OSS 開発で集中型 VCS を用いる場合，周辺開発者の名前や変更の日時といった情報が，リポジトリに正しく記録されないという問題がある．(図 1) この問題は，集中型 VCS の性質と，成果物をリポジトリに保存できるのはコア開発者だけであるという制約に起因する．周辺開発者が行った変更をプロジェクトに取り込んでもらうためには，コア開発者にリポジトリへの保存を依頼しなければならない．この時，リポジトリに記録されるメタデータはコア開発者の名前と保存日時となってしまう．

一方，分散型 VCS を用いて OSS 開発を行う場合には，周辺開発者の名前や日時を，リポジトリに正しく記録することができる．分散型 VCS であっても，プロジェクトの公式な成果物を管理するリポジトリ（公式リポジトリ）を変更できるのはコア開発者だけである．しかし，周辺開発者の変更についてのメタデータは各開発者が所有するリポジトリに記録されており，公式リポジトリに変更を取り込まれる際にはメタデータも同時に取り込まれるため，公式リポジトリに周辺開発者による

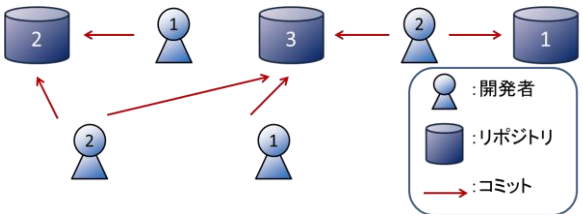

図 2 開発者・リポジトリ間の参加関係の抽出

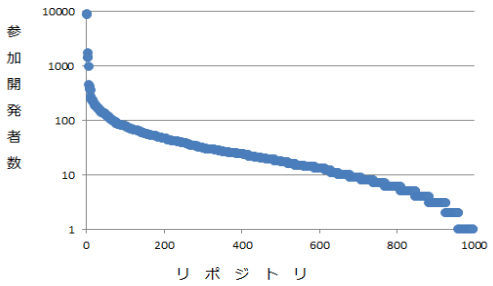

図3 リポジトリの変更に参加した開発者数

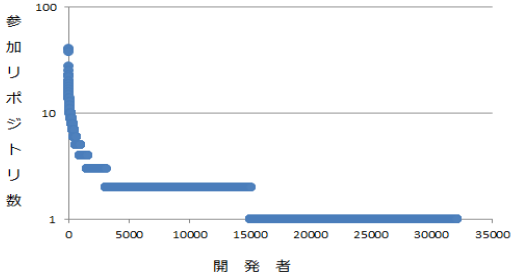

図4 開発者が変更に参加したリポジトリ数

開発の記録が正しく保存される.

3. 研究の目的と提案手法

本研究では，OSS 開発者の活動の実態を明らかにすることを目的とする．そのためには，コア開発者だけでなく周辺開発者を含めた，開発者全体の活動を調べる必要がある.

そこで我々は，分散型 VCS である Git のリポジトリから取得できる開発者のコミット記録から，開発者の名前を抽出し，その開発者がどのリポジトリに対してコミットを行なっているかを分析する．今回は，GitHub にホストされたリポジトリから，リポジトリに注目している人数の多い順に 1000 件のリポジトリのコミット記録を取得する．その後，リポジトリへのコミットに参加している開発者数・開発者がコミットに参加しているリポジトリ数をカウントすることで，分布の調査を行う．(図 2)

また，分析を行うにあたって，対象とするリポジトリの判別をする．GitHub 上のプロジェクトは，開発の中心的なリポジトリの集合と，それらに変更を送信するために複製された一時的なリポジトリによって構成される．公式リポジトリに取り込まれたコミットの多重カウント避けるため，GitHub の API を用いてリポジトリが複製されたものであるかを確認し，複製元のリポジトリのみを対象に分析を行うことで，一時的なリポジトリを除去する.

4. 分析結果

本節では，分析によって得られた開発者・リポジトリ間の参加関係について述べる.

図 3 は，1000 件のリポジトリそれぞれに対して，リポジトリの変更に参加した開発者数を調べ，その人数を多い順に並べたものである．2 人以上の開発者が参加しているリポジトリの割合は 96%，10 人以上の開発者が参加しているリポジトリの割合は 68%であった．このグラフから，大部分のリポジトリでは複数の開発者がリポジトリの変更に参加していること，一部のリポジトリは参加開発者数が多いことが見て取れる.

図 4 は，1000 件のリポジトリそれぞれに対して，いずれかの変更に参加した開発者が，変更に参加したリポジトリ数を調べ，その件数を多い順に並べたものである．1000 件のリポジトリから確認できた全体の開発者数は 32081 人，2 つ以上のリポジトリに参加している開発者の割合は 47%，3 つ以上のリポジトリに参加している開発者の割合は 10%，10 以上のリポジトリに参加している開発者の割合は 0.5%であった．このグラフから，およそ半数程度の開発者が複数のリポジトリの変更に参加していること，少数の開発者は非常に多くのリポジトリの変更に参加していることが見て取れる.

5. まとめと今後の予定

本稿では，分散型バージョン管理システムを対象とした OSS 開発記録の分析手法を提案し，実際の開発記録から開発者のプロジェクトへの参加関係を抽出した．その結果，一部のプロジェクトに非常に多くの開発者が参加している事や，開発者のうち半数以上は 2 つ以上のプロジェクトに参加している事，一部の開発者が 10 個以上のプロジェクトに参加している事などが判明した.

一方で，今回の分析では，GitHub 上の著名なリポジトリのみを分析の対象としているため，分析結果にも偏りが存在している可能性がある．また，分析の対象としたリポジトリは，プロジェクトの中心となるリポジトリのみであるため，個人用のリポジトリで行われた変更を調べることはできていない.

今後の課題としては，より多くのリポジトリに対して分析を行うことで，OSS 開発プロジェクトの規模・構成の実態をより正確に調査する必要がある．また，プロジェクトへの参加の関係だけではなく，ソースコードの変更内容などの詳細な情報を調査することや，調査によって得られた知見を OSS 開発支援に活用することなどが挙げられる.

6. 参考文献

[1] http://sourceforge.net/

[2] Yunwen Ye, Kouichi Kishida. Toward an Understanding of the Motivation of Open Source Software Developers. ICSE’03.