

# コードクローンの特性に対する 開発者の熟練度の影響に関する調査に向けて

堀 田 圭 佑<sup>†1</sup>

本稿では、開発者の熟練度と、その開発者が生成したコードクローンの特性との間に関連があるかについて、その調査の必要性、有用性を議論する。また、関連分野の研究動向に触れると共に、調査を進める上で検討すべき事柄について議論する。

## Toward Investigating Relationship between Abilities of Developers and Quality of Clones

KEISUKE HOTTA<sup>†1</sup>

This paper discusses the need and the availability of revealing relations between abilities of developers and clones created by them. It mentions related work followed a discussion of how to perform the analysis.

### 1. ま え が き

コードクローンとは、ソースコード中に存在する同一の、あるいは類似するコード片のことをいい、コピーアンドペーストによる既存コードの再利用がその主たる生成要因であるといわれている<sup>1)</sup>。一般的にコードクローンの存在はソフトウェア保守を阻害すると考えられており、これまでにコードクローンの管理技法に関する研究が盛んに行われている<sup>2)</sup>。

また近年、多くの研究者によってリポジトリマイニングを用いたコードクローンに関する定量的分析が行われている。これらの研究では、コードクローンと不具合の関連<sup>3)</sup>、コードクローンの修正されやすさ<sup>4)</sup>、コードクローンの生存期間<sup>5)</sup>など、コードクローンに関する様々な特性が調査されている。

しかし、これらのコードクローンの特性と、生成者の熟練度との関連性は明らかになっていない。例えば、経験の少ない開発者が生成したコードクローンは、経験の豊かな開発者が生成したものと比較して、不具合の混入率や同時修正を必要とする確率が高いのではないかと考えられる。これは、経験の少ない開発者は再利用対象のコードに対する理解が不十分のまま再利用を行う傾向にあること<sup>6)</sup>に起因する。

開発者の熟練度とコードクローンの特性との関連が明らかになり、熟練者の生成するコードクローンの方が経験の少ない開発者が生成するものと比較してがソフトウェア保守に与える悪影響が小さいことが立証されれば、どのような状況でどのようなコード片を再利用すればよいのかという指標を熟練者から学ぶことができる。これにより、経験の少ない開発者でも安全にコード片の再利用を行うことが可能になる。

本稿では、関連分野の研究の動向について触れた後、調査を進める上で検討すべき事柄について議論する。

### 2. 関 連 研 究

これまでに、開発者とソースコードの品質との関連を調査する研究がいくつか行われている<sup>7),8)</sup>。それらの研究の報告では、多数の開発者が関わっているコード片より、特定の開発者が集中的に関わっているコード片の方が不具合が少ないことが報告されている。換言すれば、そのコード片にあまり関わっていない、そのコード片に対する経験の少ない開発者が関わることにより、コード片に不具合が混入する確率が高くなる。

また、コードクローンとそのコードクローンに関わった開発者の数との間の関連を調査した研究が Harder によって行われた<sup>9)</sup>。その報告によると、複数の開発者が関わっているコードクローンよりも、単一の開発者のみが関わっているコードクローンの方が多数存在していた。また、関わっている開発者数による不具合

<sup>†1</sup> 大阪大学大学院情報科学研究科

Graduate School of Information Science and Technology, Osaka University

数の差異を計測した結果、開発者数の違いによる差異はないことが報告されている。

さらに、コードクローンがなぜ生成されるのかについて調査した研究もいくつか存在する<sup>10),11)</sup>。これらの研究では、コピーアンドペーストによるコードの再利用は頻繁に行われていると同時に、様々な要因から避けたいものであることが報告されている。この結果から、コピーアンドペーストによる再利用を禁止しコードクローンの生成を完全に抑止することは困難であるといえる。従って、“どのような場合にコピーアンドペーストによる再利用を許容するのか”を検討することは極めて重要であり、本研究が成就することにより一つの指標を提示できるものと考えている。

### 3. 研究の展望

開発者の熟練度とコードクローンの特性を調査する上で検討すべき事柄として、以下のものが挙げられる。

- コードクローンが生成された時期、箇所をどのように特定するか
- どのような研究課題を設定するか
- 開発者の熟練度をどのように計測するか

コードクローンが生成された時期や箇所の特定については、我々の研究グループが開発したコードクローン追跡技法<sup>12)</sup>を用いることを予定している。この技法をソースコードリポジトリに適用することで、コードクローンが生成された時期や箇所を自動的に特定することができる。

コードクローンに関する研究分野では、コードクローンに関する特性として、不具合の有無、一貫性のない修正の有無、修正されやすさなどが調査されている。本研究では、これらの特性を調査対象として研究課題を設定する予定である。また、ある特定のプロジェクトに対する熟練度の高い開発者は、そのプロジェクトに関する知識を多数持っているため、再利用可能なコード片がどこに存在するのかをよく知っていると考えられる。このように、本研究では、生成されるコードクローンの特性だけでなく、コピーアンドペーストによるコードの複製に対する開発者の熟練度の影響を幅広く調査したいと考えている。

また開発者とソースコードの品質の関係を調査した既存研究では、各開発者が行った対象プロジェクトに行ったコミット数を用いて開発者の熟練度を計測している。本研究ではこれに加え、Githubなどのリポジトリホスティングサービスを用いてそれぞれの開発者が他のプロジェクトにどの程度貢献しているのかを計測し、調査に用いたいと考えている。

### 4. あとがき

本稿では、コードクローンとその生成者の熟練度との間の関連を調査することの必要性を検討した。また、関連分野の研究動向について触れると共に、調査を進めていく上で検討すべき事柄について議論した。

謝辞 本稿は、日本学術振興会特別研究員奨励費(課題番号: 25・1382)の支援の下で作成された。

### 参考文献

- 1) 肥後芳樹, 楠本真二, 井上克郎: コードクローン検出とその関連技術, 電子情報通信学会論文誌, Vol.91-D, No.6, pp.1465-1481 (2008).
- 2) Zibran, M. F. and Roy, C. K.: The Road of Software Clone Management: A Survey, *Technical Report, University of Saskatchewan* (2012).
- 3) Juergens, E., Deissenboeck, F., Hummel, B. and Wagner, S.: Do Code Clones Matter?, *ICSE'09*, pp.485-495 (2009).
- 4) Harder, J. and Göde, N.: Cloned code: stable code, *Journal of Software : Evolution and Process* (2012).
- 5) Kim, M., Sazawal, V., Notokin, D. and Murphy, G. C.: An Empirical Study of Code Clone Genealogies, *ESEC/FSE'05*, pp.187-196 (2005).
- 6) Singer, J., Lethbridge, T., Vinson, N. and Anquetil, N.: An Examination of Software Engineering Work Practices, *CASCON'97*, p.21 (1997).
- 7) Rahman, F. and Devanbu, P.: Ownership, Experience and Defects: A Fine-grained Study of Authorship, *ICSE'11*, pp.491-500 (2011).
- 8) Bird, C., Nagappan, N., Murphy, B., Gall, H. and Devanbu, P.: Don't Touch My Code!: Examining the Effects of Ownership on Software Quality, *ESEC/FSE'11*, pp.4-14 (2011).
- 9) Harder, J.: How Multiple Developers Affect the Evolution of Code Clones, *ICSM'13*, pp.30-39 (2013).
- 10) Kim, M., Bergman, L., Lau, T. and Notokin, D.: An Ethnographic Study of Copy and Paste Programming Practices in OOP, *ISESE'04*, pp.83-92 (2004).
- 11) Zhang, G., Peng, X. and Zhao, Z. X. W.: Cloning Practices: Why Developers Clone and What can be Changed, *ICSM'12*, pp.285-294 (2012).
- 12) Higo, Y., Hotta, K. and Kusumoto, S.: Enhancement of CRD-based Clone Tracking, *IW-PSE'13*, pp.28-37 (2013).