

Title	複数のプログラミング言語間で依存関係を持つコードクローンを対象としたリファクタリング学習支援システムの提案
Sub Title	Proposal of a refactoring support system for code clones with cross-language dependency
Author	平野, 尚行(Hirano, Naoyuki) 当麻, 哲哉(Tōma, Tetsuya)
Publisher	慶應義塾大学大学院システムデザイン・マネジメント研究科
Publication year	2018
Jtitle	
JaLC DOI	
Abstract	
Notes	修士学位論文. 2018年度システムエンジニアリング学 第284号
Genre	Thesis or Dissertation
URL	https://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40002001-00002018-0053

慶應義塾大学学術情報リポジトリ(KOARA)に掲載されているコンテンツの著作権は、それぞれの著作者、学会または出版社/発行者に帰属し、その権利は著作権法によって保護されています。引用にあたっては、著作権法を遵守してご利用ください。

The copyrights of content available on the KeiO Associated Repository of Academic resources (KOARA) belong to the respective authors, academic societies, or publishers/issuers, and these rights are protected by the Japanese Copyright Act. When quoting the content, please follow the Japanese copyright act.

複数のプログラミング言語間で
依存関係を持つコードクローンを
対象としたリファクタリング
学習支援システムの提案

平野 尚行

(学籍番号 : 81733575)

指導教員 教授 当麻 哲哉

2019 年 3 月

慶應義塾大学大学院システムデザイン・マネジメント研究科
システムデザイン・マネジメント専攻

論 文 要 旨

学籍番号	81733575	氏 名	平野 尚行
論文題目： 複数のプログラミング言語間で依存関係を持つコードクローンを 対象としたリファクタリング学習支援システムの提案			
(内容の要旨) 近年、ソフトウェアの大規模化、複雑化に伴い、ソフトウェア保守の重要性が高まっている。そして、ソフトウェア保守を困難にする要因の一つとしてコードクローンが注目されている。コードクローンとは、ソースコード中に存在する互いに一致もしくは類似した部分を持つコード片のことである。また、リファクタリングとは、ソフトウェアの外部から見た振る舞いを変えずに、ソースコードの理解や修正が容易になるように、内部構造を改善する作業である。一般的に、コードクローンの存在はソフトウェアの保守を困難にするといわれており、ソフトウェアの保守性を改善するためには、リファクタリングの実施が考えられる。 現在は、複数の言語を組み合わせてソフトウェアの開発がされることは珍しくなく、1つの機能を実装するために複数のプログラミング言語を用いることがある。その結果、検出されるコードクローン間にも、プログラミング言語間の依存関係が存在する。 これらにより検出されるコードクローンには、リファクタリングが必要となる部分も多く含まれる。しかし、コードクローンが生じる要因には様々なものがあり、全てのコードクローンがソフトウェア保守の障害になるとは限らないことが指摘されている。そのため、コードクローンの検出結果をソースコードの品質改善に活用するためには、コードクローンが保守の障害になるかどうかを開発者が効率的に判断し、保守性や信頼性を低下させているコードクローンを特定して改善するリファクタリング手法の習得が必要となる。 そこで本研究では、中上級レベルを目指すソフトウェア開発者を対象として、複数のプログラミング言語間の依存関係をもつコードクローンのリファクタリング学習支援システムを提案し、要求分析、アーキテクチャ設計、開発を経て、実際にシステムを構築して適用を行う実践的な検証を行った。 評価実験では、プレテストおよびポストテストの成績と、アンケート結果を用いて検証を行った。その結果、プレテストとポストテストの成績の検証では、ポストテストの成績はプレテストの成績より有意に高得点であった。これは、複数のプログラミング言語間の依存関係をもつコードクローンのリファクタリング学習支援システムの利用により、リファクタリングに関する要点の把握が進み、その結果として学習効果を高めることが出来たためと考えられた。また、アンケート結果を用いた検証でも、一定の評価を得られ、開発したリファクタリング学習システムの有効性が実証できた。			
キーワード (5 語) コードクローン、リファクタリング、複数プログラミング言語、ウェブアプリケーション、学習支援システム			

SUMMARY OF MASTER'S DISSERTATION

Student Identification Number	81733575	Name	Naoyuki Hirano
Title Proposal of a Refactoring Support System for Code Clones with Cross-Language Dependency			
Abstract In recent years, along with the increase in the scale and complexity of software, the importance of software maintenance is increasing. And code clones are attracting attention as one of the factors that make software maintenance difficult. A code clone is a code fragment having matching or similar parts existing in the source code. In addition, Refactoring is a task of improving the internal structure so as to make it easy to modify the source code without changing the function of the software. Generally, it is said that the existence of a code clone makes maintenance of software difficult, and refactoring can be considered to improve software maintainability. Currently, it is not uncommon for software to be developed by combining multiple languages, and there are cases where one function is implemented in multiple programming languages. As a result, dependency exists also between detected code clones. Code clones also contain many parts that require refactoring. However, there are various factors causing code clones, and it is pointed out that not all code clones are degrading the maintainability of software. For this reason, in order to utilize the detected result of the code clone to improve the quality of the software, it is necessary for the developer to efficiently judge whether or not the code clone becomes a factor that deteriorates the maintainability of the software. And developers need to acquire refactoring techniques to identify and improve code clones that reduce software maintainability and reliability. Therefore, in this research, we proposed a refactoring support system for code clones that have dependencies between multiple programming languages, for software developers who want to move on to intermediate or advanced level. Then, we implemented system requirements analysis and architectural designing, developed a system, and carried out practical verification. We conducted the verification using the results of the pretest and the posttest and the questionnaire result. As a result, in the verification of the results of the pretest and posttest, the results of the posttest were significantly higher than the results of the pretest. This is probably because we were able to grasp the point of refactoring by using refactoring support system. In addition, we also obtained a certain evaluation from the questionnaire result verification, and we demonstrated the effectiveness of the refactoring support system we developed.			
Key Word(5 words) Code Clone, Refactoring, Cross Language, Web Applications, Support System			