

Title	リファクタリングの履歴を用いたソフトウェア開発支援ツールの研究
Author(s)	高橋, 克史
Citation	
Issue Date	2006-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1974
Rights	
Description	Supervisor:鈴木 正人, 情報科学研究科, 修士

リファクタリングの履歴を用いたソフトウェア開発支援ツールの研究

高橋 克史 (410072)

北陸先端科学技術大学院大学 情報科学研究科

2006 年 2 月 9 日

キーワード: リファクタリング, 履歴, ソフトウェアメトリクス.

背景と目的

ソフトウェア開発では仕様の変更や機能の拡張が連続的に発生し、その結果ソフトウェアは徐々に複雑化し、可読性、保守性が低下する。これを改善する方法としてリファクタリング [Fowler] がある。リファクタリングとは、ソフトウェアの機能を変えずに内部構造を変更する作業を指す。しかしながら、実際の開発現場では、リファクタリングは必ずしも有効に適用されていない。その理由の一つに、開発における特定の目的をリファクタリングによって達成することが困難だということがあげられる。個々のリファクタリング操作の粒度が細かいため複雑なソフトウェアを改善するには、多くのリファクタリング操作の組み合わせが必要になる。さらに、リファクタリングした結果は、適用したリファクタリング操作の種類や順番によって異なってくる。そのため経験の浅い開発者が目的を達成することは難しい。通常は、目的を達成するためにいくつかの操作手順を、試行錯誤しながら少しずつ改善していく必要があるが、既存のツールでは手順を記録、再利用できるものがほとんどない。本研究では、リファクタリング操作の履歴を用いることでより効果的な適用を支援するツールを開発し、それを利用した方法論を提唱することを目的とする。

リファクタリングの履歴を用いた開発支援

本研究は、利用者が行ったリファクタリング操作の履歴 (**操作履歴**) を保存できることが特徴である。操作履歴は、試行錯誤の過程を記録するために木構造をしている。利用者が行ったリファクタリング操作は、操作履歴のノードとして保持される。リファクタリングでは、複数の基本操作をソースコードの異なる箇所に適用する必要がある。本研究では対象を Java 言語とし、基本操作としては、[Fowler] から使用頻度の高い 19 個の操作を抽出し自動化する。基本操作を計算機で実行可能なようにプログラムの構造間の写像として形

式化したものを**基本命令**と呼ぶ。基本命令は、基本操作を識別する識別子と適用箇所と適用実体からなるオペランドを持つ。適用箇所とは、基本操作を適用する場所を(パッケージ名、クラス名、メソッド名、メソッド内相対位置)の組で表したものである。適用実体とは、基本操作の実行に必要な情報であり操作ごとに異なる。例えば「メソッド抽出」では、抽出されたメソッドに付加する名前である。操作履歴の各ノードとしては、この基本命令が保持される。複数の操作を組み合わせたものを一つの**変更方針**として管理する。変更方針としては、(操作列、部分目標)の組として管理する。部分目標とは、変更方針の操作列のゴールを表したものである。これにより、後で操作履歴を参照したときに、これまでどのような操作が行われてきたのか一目で分かるようになる。

操作履歴を参照して、行き詰まりになった場合でも回復を容易にする。行き詰まりとは、それ以上リファクタリング操作を適用できない、または適用することで以前行った操作の効果を失う状態のことを言う。リファクタリングには試行錯誤が不可欠であるが、履歴を利用することでこれの発生を最小化し、また複数の変更方針を同時に適用し結果を比較することで目的を達成する変更方針の発見を支援する。具体的には、あるソースコード O に対し、ある変更方針に従って操作列を適用した結果を A とし経過を履歴 $H_O \rightarrow A$ に保存する。利用者は、 A が目的を達していない、またはこのまま継続しても目的を達成するのが困難と判別したら、ソースコード O に戻し、別の変更方針を選択する。それに従ってリファクタリングを適用し結果 B と履歴 $H_O \rightarrow B$ を得る。もし B を適用後、やはり A の方が目的に近いことが判明したら、直ちに $H_O \rightarrow A$ を適用し直すことで手戻りを最小化できる。このとき結果でなく履歴を保存することで必要ならば O から A または B に至る途中での分岐の作成を容易にし、比較対象の選択肢を増やすことが可能になる。

リファクタリング適用箇所の支援には複数のソフトウェアメトリクスを提供する。メトリクスは、目的の決定とリファクタリング操作の適用結果の評価に利用する。特に利用者にリファクタリングの経験が少ない場合は、複雑なソフトウェアから問題となる部分の発見が困難である。ソースコードの特性を数値化することでリファクタリングによって改善が可能な箇所の発見と解決の方針決定を支援する。また、操作結果を比較評価する場合は、操作前と後のメトリクス値を比較することで客観的な評価を行えるようにする。

まとめ

本研究では、リファクタリングの履歴を用いることでより効果的なリファクタリングする手法を提案し支援ツールを作成した。提案した手法では、利用者の行ったリファクタリング操作の過程と結果をその部分目標と共に操作履歴として保存する。操作の自動化のためリファクタリング操作を形式化し、複数の操作の組み合わせを一つの変更方針として管理する。履歴を参照することで、複数の変更方針を同時に実行し、その結果を比較できるようにすることで、目的を早く達成する。本研究により、開発現場にリファクタリングを導入し目的を達成することが容易になり、開発コストの低下が期待できる。