

Title	クラス間の通信形態に注目した設計レベルの再構成手法の研究
Author(s)	大尾, 健介
Citation	
Issue Date	2007-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/3585
Rights	
Description	Supervisor:鈴木 正人, 情報科学研究科, 修士

クラス間の通信形態に注目した 設計レベルの再構成手法の研究

大尾 健介 (510058)

北陸先端科学技術大学院大学 情報科学研究科

2007年2月8日

キーワード: ソフトウェア再構成, クラス間の相互作用, ルール駆動手法,
アーキテクチャ, オブジェクト指向ソフトウェア, UML.

1 はじめに

アプリケーションに機能の拡張や追加を行う場合、ソースコードの変更方法が問題となっている。機能拡張/追加を実現するために、リファクタリングを用いた手法などが提案されている。しかし、機能拡張/追加に伴い新しいサービスとの連携を行うことがあり、従来の小規模かつ局所的な修正では対処できなくなる可能性がある。しかしながら全体的かつ統一性をもった設計変更に対しては有効な手法や支援環境は確立されておらず、設計を1から作り直す場合もあるのが現状である。

本研究では、システムの機能拡張/追加の際に必要な全体的なアーキテクチャの変更(設計レベルの再構成)を支援する手法を提唱する。設計レベルでの再構成手法の目標として以下の2つを掲げる。

1. システムの既存部分に対する変更を最小にする
2. 新しく作成するクラスの数をも最小にする

2 再構成支援手法の提案

本研究で提唱する再構成支援手法は、再構成ルールに基づいている。再構成ルールは以下の手順で作成される。

- (1) システム中のクラス間の相互作用 (communication) に着目する。一般に機能拡張/追加により既存クラス間でやり取りされるメッセージが変化するので、その変化を計測するためにクラス間の相互作用を特徴づける値として以下の3種類を採用する。

- Style: 通信の形態 (1 対 1 / 1 対多, 同期/非同期)
 - Amount: メッセージに含まれるデータ量
 - Frequency: メッセージの頻度
- (2) クラス間の相互作用を基にシステムのサービスを分析する。分析結果から相互作用の 3 種類に効果のある改善策を作成し、再構成ルールとする。
 - (3) 作成した再構成ルールの効果を検証するため、同じドメインに属する別のシステムに適用し、再構成を行う。そして、再構成を行った結果と比較対象との比較を行う。
 - (4) 比較結果から作成した再構成ルールの修正を行い、洗練する。より具体的には
 - (4-1) 機能拡張/追加を行う際に登録した再構成ルールを利用する。機能拡張/追加部分の内容からシステムの変更部分を抽出する。
 - (4-2) 変更部分から機能追加/拡張を行うにあたり効果のある再構成ルールを選び出す。再構成を行うことで、機能拡張/追加要求に対応しやすくする。

3 作成した再構成ルール

例題として、小規模なホテルの客室予約管理システムを対象に機能追加(キャンセル待ち)を行った場合の再構成ルールの抽出・作成を行い、以下の 3 つの再構成ルールを作成する。()内は関連する特性である。

- R1: クラスの役割の委譲 (Frequency)
- R2: クラスの従属化 (Style)
- R3: オブジェクトとデータの不整合の解消 (Amount)

R1 はクラスが持つ役割を新しいクラスに委譲する。あるオブジェクト間で頻繁なメッセージ通信が行われており、あるメッセージを送信する前処理だと仮定する。R1 を適用することで頻繁に発生する前処理メッセージを新しいクラスが受け持ちさらに、内部通信に変更することで大幅な外部メッセージ総数が減少する効果を得ることができる。

R2 はあるクラス間に従属関係を持たせる。別のクラスから同じタイミングでメッセージが送られており、送られるクラスが持つデータの性質が類似していると仮定する。R2 を適用することで個別に送信されたメッセージをマスタークラスが受け取り、スレーブクラスへまとめて送信する。1 対多の通信形態が 1 対 1 に変更され、外部メッセージ総数が減少する効果を得ることができる。

R3 はデータ境界とオブジェクト境界の不整合を解消する。メッセージが送るまとまったデータの一部分をオブジェクトが使用すると仮定する。R3 を適用することでオブジェ

クトが使用する分のデータだけをメッセージが送るので使用しないデータを削除する効果を得ることができる。

作成した再構成ルールをシステムに適用し、機能追加にどのような影響を及ぼすのか実験を実施した。

4 再構成ルールの適用実験

この例で追加されたキャンセル待ち機能は、以下の2つのDB操作からなる。

- (1) 予約記録をキャンセル待ち記録に変更し、キャンセル待ちデータベースへ登録するまでのDB操作(以下キャンセル待ち記録登録機能:SA1と呼ぶ)。
- (2) 予約記録のキャンセルが発生した場合に、予約記録に変換できるキャンセル待ち記録を検索し、予約データベースへ登録するまでのDB操作(以下キャンセル時の利用可能記録検索機能:SA2と呼ぶ)。

適用実験は、以下の順番で実施する。

- (1) 機能追加前の客室予約管理システム(SB)に再構成ルールを行わずキャンセル待ち機能を追加する(SA1/SA2)。SBからSA1/SA2に変更した部分の分析を行う。
- (2) R1, R2, R3をそれぞれSBに適用する(SB+R1, SB+R2, SB+R3)。
- (3) SB+R1, SB+R2, SB+R3にキャンセル待ち機能を追加する(SA1+R1/SA2+R1, SA1+R2/SA2+R2, SA1+R3/SA2+R3)。
- (4) SA1/SA2の結果とSA1+R1/SA2+R1, SA1+R2/SA2+R2, SA1+R3/SA2+R3の結果の比較を行いR1, R2, R3の効果を評価する。

5 評価結果

比較評価を行った結果を各再構成ルールのごとに利点・欠点・改善案を順番に示す。

- 利点

R1:キャンセル待ち記録登録機能におけるSA1とSA1+R1を比較した結果、R1を適用することでキャンセル待ち記録登録機能のメッセージ総数が1つ減少した。

キャンセル時の利用可能記録検索機能におけるSA2とSA2+R1を比較した結果、R1を適用することでキャンセル時の利用可能記録検索機能のメッセージ総数が3つ減少した。既存クラスに追記するメソッドの数が4つ減少した。

R2:キャンセル時の利用可能記録検索機能におけるSA2とSA2+R2を比較した結果、R2を適用することでキャンセル時の利用可能記録検索機能のメッセージ総数が3つ減少した。

R3:キャンセル時の利用可能記録検索機能における SA2 と SA2+R3 を比較した結果、R3 を適用することでキャンセル時の利用可能記録検索機能で発生するメッセージのデータとオブジェクトとの不整合が解消した。

- 欠点

R1:キャンセル待ち記録登録機能における SA1 と SA1+R1 を比較した結果、R1 を適用することで新しく作成したクラス/メソッドが 1 つ増加した。

キャンセル時の利用可能記録検索機能における SA2 と SA2+R1 を比較した結果、R1 を適用することで新しく作成したクラス/メソッドが 1 つ増加した。

R2:キャンセル時の利用可能記録検索機能における SA2 と SA2+R2 を比較した結果、R2 は既存クラスを修正する部分の減少に効果がなかった。

R3:キャンセル時の利用可能記録検索機能における SA2 と SA2+R3 を比較した結果、R3 は既存クラスを修正する部分の減少に効果がなかった。

- 改善案

R1:キャンセル待ち記録登録機能では、前処理となるメッセージが 1 つしか存在しないので R1 の効果が得られない。最低 2 つ以上のメッセージが発生する場合に R1 を適用すべきである。(パラメータの設定) キャンセル時の利用可能記録検索機能でも同様な改善案となった。

6 結論

本研究では再構成ルールを用いた設計の再構成支援を提唱し、3 つの再構成ルールを作成した。3 つの再構成ルールの適用実験を行った結果、R1 を適用した時に既存部分の変更箇所を減少することができた。これにより R1 を適用することで機能追加がしやすくなった。

今後は、さらに適用実験を行い再構成ルールの洗練を行っていく。