

A Quantitative Study on Characteristics and Effect of Batch Refactoring on Code Smells

紹介者 森岡勇介

背景

- リファクタリングは多くの場合バッチで行われる
- 現在のリファクタリング手法を改善するためにはバッチリファクタリングの知識が必要
- 経験に基づいたバッチリファクタリングの知識を獲得する研究が少ない

目的

二つの観点からバッチリファクタリングを理解すること。

- ・ バッチを構成する特性
- ・ コードスメルに対するバッチの影響

リサーチクエスチョン

目的を達成するために二つのリサーチクエスチョンを設定

RQ1: バッチ特性ごとの出現の割合は？

RQ2: バッチはどのようにコードスメルに影響している？

コード変換の性質

TRANSFORMATION TYPES INVESTIGATED BY THIS WORK

Transformation Type	Definition
Move Attribute	Move an attribute across classes
Pull Up Attribute	Move attribute from child to parent class
Push Down Attribute	Move attribute from parent to child class
Extract Method	Extract a new method from an existing one
Inline Method	Move a method body to an existing method
Move Method	Move a method across classes
Pull Up Method	Move method from child to parent class
Push Down Method	Move method from parent to child class
Rename Method	Update name of an existing method
Extract Interface	Extract an interface from an existing class
Extract Superclass	Extract a superclass from an existing class
Move Class	Move a class across packages
Rename Class	Update name of an existing class

13あるコード変換をタイプごとにグループした表

バッチ特性の定量的分析

RQ1:バッチ特性ごとの出現の割合は？

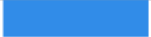
TYPICAL MANIFESTATIONS BY BATCH CHARACTERISTIC

Charact.	Manifestation	Description
<i>Cardinality</i>	2	Batch has two transformations
	≥ 3	Batch has three or more transformations
<i>Commit</i>	<i>One commit</i>	Batch completed in only one commit
	<i>Many commits</i>	Batch completed in two or more commits
<i>Scope</i>	<i>Method</i>	Batch transformed only one method
	<i>Class</i>	Batch transformed multiple methods or class
<i>Smelliness</i>	<i>Smelly</i>	Batch affected smelly code element
	<i>Not smelly</i>	Batch did not affect smell code element
<i>Variety</i>	<i>One type</i>	Batch has only one transformation type
	<i>Many types</i>	Batch has two or more transformation types

バッチ特性の定量的分析 結果

RQ1:バッチ特性ごとの出現の割合は？

TABLE V
FREQUENCY OF MANIFESTATIONS BY BATCH CHARACTERISTIC

Characteristic	Manifestation	# Batches	Frequency	
<i>Cardinality</i>	2	2,605	57%	
	≥ 3	2,002	43%	
<i>Commit</i>	<i>One commit</i>	4,265	93%	
	<i>Many commits</i>	342	7%	
<i>Scope</i>	<i>Method</i>	428	9%	
	<i>Class</i>	4,179	91%	
<i>Smelliness</i>	<i>Smelly</i>	2,911	63%	
	<i>Not smelly</i>	1,696	37%	
<i>Variety</i>	<i>One type</i>	3,330	72%	
	<i>Many types</i>	1,277	28%	

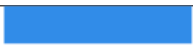
バッチ特性の定量的分析 結果

RQ1: バッチ特性ごとの出現の割合は？

Characteristic	Manifestation	# Batches	Frequency	
Cardinality	2	2,605	57%	
	≥ 3	2,002	43%	

半分以上のバッチが2つのコード変換で構成されている

単純なコードスメルに対しては有効であるかもしれないが私たちの研究で検討されているほとんどのコードスメルに対しては不十分であると考えられる

Commit	One commit	4,265	93%	
	Many commits	342	7%	

ほとんどのバッチが一つのコミットである

バッチ特性の定量的分析 結果

RQ1:バッチ特性ごとの出現の割合は？

<i>Smelliness</i>	<i>Smelly</i>	2,911	63%	
	<i>Not smelly</i>	1,696	37%	

機能の拡張やバグの修正のためのバッチが**Not Smelly**にあたるので妥当である

<i>Scope</i>	<i>Method</i>	428	9%	
	<i>Class</i>	4,179	91%	
<i>Variety</i>	<i>One type</i>	3,330	72%	
	<i>Many types</i>	1,277	28%	

コードスメルは複数のメソッドに影響を与えていることが多いので、**Scope**は期待通りの結果となっている

Varietyは複数のタイプを組み合わせた方がリファクタリングの結果がよくなるという先行研究があるので少し予想から外れている

バッチ特性がコードスメルに与える影響の定量的分析

RQ2:バッチはどのようにコードスメルに影響している？

BATCH EFFECT ON CODE SMELLS (PLUS FISHER'S TEST) BY BATCH CHARACTERISTIC

Characteristic	Manifestation	Positive		Negative		Neutral		Odds Ratio (OR)	p-value
		Absolute	%	Absolute	%	Absolute	%		
Cardinality	2	189	6%	847	29%	570	20%	0.90	> 0.05
	≥ 3	160	5%	647	22%	498	18%		
Commit	One commit	328	11%	1,414	49%	994	34%	0.92	> 0.05
	Many commits	21	1%	80	3%	74	4%		
Scope	Method	30	1%	52	2%	17	0%	2.60	< 0.05
	Class	319	11%	1,442	50%	1051	38%		
Variety	One type	240	8%	1,051	36%	749	26%	0.92	> 0.05
	Many types	109	4%	443	15%	319	12%		

Positive : バッチがコードスメルを減らしている

Negative : バッチがコードスメルを増やしている

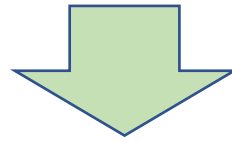
Neutral : バッチの前と後で変わらない

コードスメルに関するバッチのうち約半数はコードスメルを増やしてしまっていると判明

バッチ特性がコードスメルに与える影響の定量的分析

RQ2: バッチはどのようにコードスメルに影響している？

単一のリファクタリングでは33%がコードスメルを増やしてしまっているという先行研究があるが、バッチでは51%もある。



バッチリファクタリングを行う際のガイドラインを作成する必要があるのではないか。

コード変換の組み合わせに着目したバッチの影響

NATURE OF TRANSFORMATION TYPES

Nature	Definition	Transformation Types
<i>Extraction</i>	Extracting (parts of) an existing code element aimed to create a new element	Extract Interface Extract Method Extract Superclass
<i>Inline</i>	Copying a code element's body for pasting into an existing element	Inline Method
<i>Motion</i>	Moving a code element within the software project	Move Attribute Move Class Move Method
<i>Pull Up</i>	Moving a code element from subclass to superclass	Pull Up Attribute Pull Up Method
<i>Push Down</i>	Moving a code element from superclass to subclass	Push Down Attribute Push Down Method
<i>Rename</i>	Renaming a code element	Rename Class Rename Method

コード変換を性質ごとにグループした表

コード変換の組み合わせに着目したバッチの影響

BATCH EFFECT ON CODE SMELLS ACCORDING TO THE NATURE OF CODE TRANSFORMATIONS

Nature of Code Transformations	Total		Code Smell Type									
	Abs.	%	Class Data s/b Private	Complex Class	God Class	Intensive Coupling	Spaghetti Code	Speculative Generality	Feature Envy	Long Method	Long Param. List	Message Chain
Extraction	1,449	31	100	78.7	92.3	87.1	83.2	100	83.8	61.4	95.1	82.2
Motion	1,142	25	98.5	85.5	89.3	*100	88.5	86.8	82.4	*100	*100	41.7
Rename	580	13	100	100	91.7	100	80	100	98.7	94.1	100	97.8
Pull Up	352	8	87.5	88.9	77.8	*100	57.1	95	47.8	*100	*33.3	28.6
Extraction/Inline	233	5	100	87.5	92.3	87.5	81	*100	84.8	88.2	81.3	80
Inline	221	5	100	73.9	*66.6	88.9	75.0	*100	45.1	60	72.2	76.5
Extraction/Motion	153	3	100	78.4	88.9	*100	94.4	*100	98.1	73.7	100	37.5
Extraction/Rename	105	2	100	92.6	100	*100	100	*100	87.5	87.5	100	100
Motion/Pull Up	84	2	*50	85.7	*50	*0	*50	100	*0	*0	*0	*0
Motion/Rename	59	1	100	100	*100	*50	*50	*0	77.3	*66.6	*0	37.2
Push Down	58	1	*100	75.0	*0	*0	*0	100	*0	*0	*0	*0
Inline/Motion	52	1	*100	75.0	*50	*100	*33.3	*0	35	66.7	40	*50
Extraction/Inline/Motion/Rename	28	1	100	100	*100	*100	*100	*0	100	100	*100	*100
Inline/Rename	25	1	*100	100	100	*100	*100	*100	*100	*100	*100	*0
Extraction/Pull Up	18	0	*0	*0	*100	*0	*0	*0	50.0	*0	*100	*0
Extraction/Push Down	11	0	*100	*66.6	*0	*0	*100	*100	80	*0	*0	*100
Extraction/Inline/Motion/Pull Up/Push Down/Rename	11	0	*0	*0	*0	*0	*0	*66.6	*66.6	*0	*100	*0
Inline/Pull Up	8	0	*0	*0	*0	*0	*0	*0	*0	*0	*0	*0
Pull Up/Rename	7	0	*0	*100	*0	*0	*0	*0	*0	*0	*0	*0
Pull Up/Push Down	6	0	*100	*0	*0	*0	*0	*100	*100	*0	*0	*0
Inline/Push Down	3	0	*0	*0	*0	*0	*0	*0	*20	*0	*0	*100
Push Down/Rename	2	0	*0	*0	*0	*0	*0	*0	*0	*0	*0	*0
All	4,607	100	98.3	84.2	89.4	88.6	83.0	94.4	81.9	69.9	91.9	74.6

Negative Positive Neutral *は数が少なすぎるのであてにならないもの

やはり全体的に良い影響が出ていない

まとめ

現在のリファクタリングツールは一つのコード変換をサポートするもの

(1)コード構造の改善に向けて2つ以上のコード変換が相互にどのように関連するかについて議論が必要

(2)同じリファクタリングを目的とした複数のコード変換の組み合わせをサポートするツールが必要

まとめ

- ソフトウェアアーキテクチャに着目したバッチの影響を調査することがこの研究の展望に挙げられる