

適切なクラス抽出を行うためのユースケース記述推敲支援

Writing Tool to Developing Use Case Documents for Proper Class Extraction

山梨 敦志[†]松浦 佐江子[‡]

Atsushi Yamanashi

Saeko Matsuura

1 はじめに

近年、オブジェクト指向分析設計の手法としてユースケース (use case) が使用されるようになった。ユースケースとは、システムが果たすべき機能をユーザレベルで記述したものとして、システムの外部仕様を表現するのに使用される。

ユースケースの内容を具体的に記述したものがユースケース記述となる。ユースケース記述ではアクターとシステムのやり取りを1ステップずつ記述する。チームによるユースケース設計における最大の障害は、整合性の欠如である。これは、アイデアや手法が、プロジェクトが進行するに従って進化しているからである。また、人々は、単に異なるアイデアを持っているだけでなく、その表現方法も異なる。[1]

オブジェクト指向によるモデリングの難しさは適切なクラスの抽出にある。ユースケース記述からシーケンス図による分析を踏まえてクラスを定義する場合、ユースケース記述における単語の統一、能動態の使用、実装に依存しない単語の使用、文における主語の明確化の整合性が取れていないと、複数人で分析作業を行うことにより、クラスの責務がシーケンス図毎に異なったり、メッセージの粒度や意味が不揃いになり、クラスの抽出がより一層難しくなる。

[3]では個々のユースケース記述に着目して整合性を持たせるための推敲を提案した。本研究では、ユースケース記述の整合性およびクラス抽出に着目したユースケース記述の推敲を提案する。

2 ユースケース記述の整合性を欠く原因

この章では、ユースケース記述の整合性を欠く原因と整合性を欠いた場合に、クラス抽出にどのような問題が発生するか記述する。これらの問題が発生すると、複数人でシステム設計を行う場合だけでなく、個人でシステム設計を行う場合にも適切なクラス抽出の妨げになる。

2.1 書式の不統一

ユースケース記述は短く明確に、読みやすいように記述しなければならないが、ユースケース記述には決まったスタイルが無いため、設計者のスキルによって表現方法が変わる。[2]

この場合、ユースケース記述を読み取る際の誤解の原因となる。また、複数人でシステム設計を行う際の意思統一の妨げになり責務、名称の不統一につながる。

2.2 単語の不統一

ユースケース記述は複数人で作成するため、各々が使

[†]芝浦工業大学大学院工学研究科電気電子情報工学専攻
Graduate School of Engineering,

Shibaura institute of technology

Department of electronic engineering and computer science

[‡]芝浦工業大学システム工学部電子情報システム学科
Shibaura institute of technology

Department of electronic information system

いやすい単語を使用する。そのために、同じ単語が異なる概念を表現し、異なる単語で違う概念を表現する。

この場合には、シーケンス図のオブジェクト名、メッセージ名が統一されないため、クラスの名称やメソッドの名称を統一することが出来ない。

2.3 実装に依存した単語の使用

ユーザインターフェース設計に対するわずかな変更でも、「要求」自体を変更しなければならなくなるため、要求が不安定になる。[2]

また、要求分析の段階でシーケンス図に用いるオブジェクト名を使用すると、ユースケース記述自体の柔軟性がなくなり、クラスの責務の分担が困難になる。

3 不適切なクラス

3.1 責務の不統一

クラスの責務とは、クラスが持つ役割のことであり、クラスはそれぞれ責務が決まっているべきである。

しかし、ユースケース記述の整合性が取れていないと同じ責務を持つオブジェクトが抽出されたり、同様の動作を行うメソッドが抽出され、責務の統一が困難になる。

3.2 名称の不統一

クラスやメソッドの名称はそれぞれの責務や機能を表す名称が付けられるべきである。

しかし、ユースケース記述の整合性が取れていないと同じ責務を持つ異なる名称のクラスや異なる責務を持つ同じ名称のクラスが抽出されることがある。同様に、同じ機能を持つ異なる名称のメソッドや異なる機能を持つ同じ名称のメソッドが抽出されることがある。

4 不整合に対する解決方法の提案と考察

この章では、[3]で提案したユースケース記述の不整合の解決方法を示し、考察を行う。また、考察を踏まえて[3]の提案に新たな提案を追加する。4.1,4.4に以前の提案を記述し、4.2,4.5にそれぞれの考察を記述する。また4.3,4.6にそれぞれの新たな提案を記述し、4.7に推敲の流れを記述する。

4.1 形態素解析を利用した書式と単語の統一

ユースケース記述は一般的に自然言語で書かれているため、分析には自然言語処理の技術である形態素解析を用いる。それによって、ユースケース記述の単語(名詞、動詞)の抽出を行い、どのユースケースで使用されているか確認できる単語一覧表を作成する。単語一覧表では、単語、使用しているユースケース数、使用しているユースケース名を表す。この単語表を用いて、単語一つ一つをユースケース記述と照らし合わせ、その単語の使用用途を定めて単語の統一を行う。書式を能動態に統一し、「～し、～する」の様に文を区切っている場合には、ステップを2つに分割して、各ステップの書式を統一する。

4.2 書式と単語の統一に対する考察

4.1 の推敲により、書式の統一を行いユースケース記述の理解を助けることが出来た。また、単語の統一を行い総単語数が減少した。これにより単語の統一が行えたと言える。しかし、その一方で、単語の統一が正しいものかどうかの指標を決めることは出来なかった。なぜなら、単語の統一が正しいかどうかは、ユースケース記述を作成したチームが決めるからである。

4.3 単語の統一に対する追加提案

単語の統一が正しいものかを判断するためにユースケース記述を作成したチームで話し合いを行い意思統一をする必要がある。話し合いの際に必要なものは、何を基準に統一を行ったかである。よって、推敲作業のログを保存する支援を行う。

4.1 の推敲方法では、単語の使用用途を確認する単語一覧表を作成するため、比較的容易に推敲作業のログを残すことが出来る。ログを保存することにより、何を基準に単語を統一したかが明確になる。ログとして保存するのは「単語」、「単語を使用しているユースケース」、「使用用途」、「どの単語と意味が重複しているか」である。

例として、自販機シミュレータのユースケース記述を分析した一部を表1に示す。

表1 保存するログ

単語	使用ユースケース	使用用途	重複単語
渡す	商品を登録する	情報を送る	
送る	お釣を排出する	情報を送る	渡す
送信する	物があるか調べる	情報を送る	渡す

4.4 要求仕様書に基づく単語の選別

要求仕様書から単語を抽出し、ユースケース記述との整合性を検証する。ユースケース記述から抽出した単語一覧表と比較し、要求仕様書では使用されていないが、ユースケース記述では使用されている単語を抽出する。これを元に、実装に依存している単語を検出し、訂正を行う。

4.5 単語の選別に対する考察

この推敲により、実装に依存する単語を検出することは出来たが、明確な指標がなかったために、実装に依存する可能性がある名詞が多数抽出され、実装に依存する名詞を検出するための作業量が多かった。

4.6 単語の選別に対する追加提案

システムを実装する際に使用する言語のコンポーネントに属する単語を使用している場合には、実装に依存している可能性があるため、該当する名詞を抽出し、4.3の場合と同様に話し合いによって名詞を変更するかどうか決定する。

例えば、JavaでGUIを実装するために用意されているSwingコンポーネントがあり、Swingには様々なコンポーネントが用意されている。これらのコンポーネントのユースケース記述での名称の例を表2に示す。これらの名称がユースケース記述に記述されている場合には、実装に依存する可能性がある単語として抽出する。

表2 Swingコンポーネントの一部

コンポーネント名	ユースケース記述での名称
JButton	ボタン
JRadioButton	ラジオボタン
JComboBox	コンボボックス
JTextField	テキストフィールド
JMenuBar	メニューバー

4.7 推敲の手順

上記で提案した推敲を図1に示すような手順で行う。要求仕様書、依頼者の定義した要求文からそれぞれユースケース記述、要求文を抽出する。それらから抽出した単語一覧表を元に推敲を行う。

単語一覧表を元に全ての単語に対して使用用途を定めたら、チームによる話し合いによってユースケース記述を訂正する。訂正したユースケース記述から単語一覧表を作成し、この手順を繰り返す。訂正する単語がなくなるまでこの作業を繰り返す。

単語の統一が終了したら、単語の選別による実装に依存する可能性がある名詞一覧表を作成し、一覧表を元にチームで話し合いを行い実装に依存する単語を検出する。

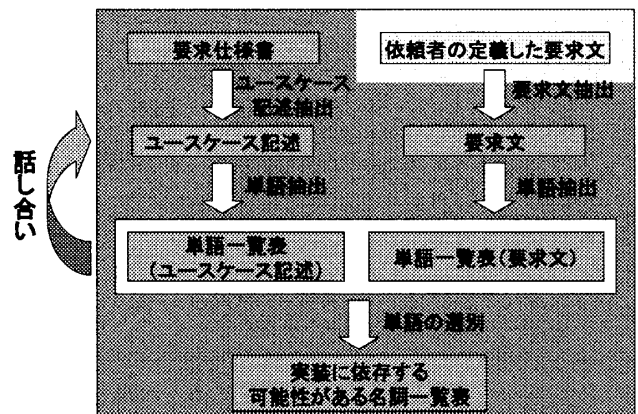


図1 推敲の手順

5 まとめ

4.3,4.6で追加提案した解決方法で実際に推敲を行い、有効性を検証する。また、ユースケース記述の整合性を取ることで適切なクラス抽出に繋がることを示す。

4.7で提示した推敲の手順では、単語の統一の推敲によって訂正したユースケース記述を元に、何度か推敲作業を繰り返す行うため、作業量が多くなる。よって、推敲作業を終了する指標を定める必要がある。

6 参考文献

- [1]Daryl Kulak and Eamonn Guiney, : ユースケース導入ガイド,ピアソン・エデュケーション,2002
- [2]Alistair Cockburn: ユースケース実践ガイド,ウルシステムズ,2001
- [3]山梨,松浦: 自然言語処理を利用したユースケース記述推敲支援,第68回情報処理学会全国大会,6A-3,2006