

# アクティビティ図の記述支援機能を持つモデリングツール 「モデカツ！」の研究

橋浦研究室 1135230 齋藤 智之 1135360 平田雅弥

## 1.はじめに

近年ソフトウェアの複雑化・大規模化に伴い、開発するシステムの要求を分析し、明確化することの重要性が高まっている[1]. 自動車や家電に用いられる組み込み制御ソフトウェアの開発量は増大しており、開発効率の向上が重要な課題となっている[2]. プロジェクトの管理を行う際、どの役割の人がどのような業務処理をどのようなタイミングで行っているかを捉える必要がある.

このような業務の流れを捉える際に UML の振る舞い図に分類されているアクティビティ図が使われている.

## 2.初級者の問題

アクティビティ図は優れた表現力を持っている反面、初級者が記述した際に様々な問題が発生する.

初級者がアクティビティ図を記述した際にどのような問題が起こるのかを確認するため、予備実験を行った. 実験結果を以下に示す.

- ① 動機をアクションとして記述してしまう
- ② ガードに書くべき内容をアクションとして記述してしまう

これはアクティビティ図の性質上、内容が間違っただけの図としても記述ができてしまうからである.

①と②を初級者に起こりやすい問題として判断し、初級者がアクティビティ図を記述する際には支援が必要になるという結論を得た.

## 3.研究目的

本研究では初級者の問題で挙げられた①や②の問題の解決を目的とする. 支援の方法として、アクションの修正基準を表示させ、初級者でもアクションの内容の間違いを修正できるように支援を行う.

アクションの修正基準は以下の通りである.

- A) 「～する」のような動詞になっているか
- B) システムなどの動作を表しているか
- C) そのアクションは目的を持っているか

- D) 状態を表していないか
- E) 現在進行形になっていないか
- F) 条件分岐の内容をアクションとして書いていないか
- G) アクションが抽象的になっていないか
- H) 1つのアクションに詰め込み過ぎでないか
- I) アクターに合った位置にアクションが書かれているか
- J) アクションが冗長になっていないか
- K) 動機が書かれていないか

## 4.モデカツ！の実装

GWTUML をベースに、アクティビティ図の記述が可能な環境(以下、「モデカツ！」と記す)を構築する. 構築する環境が持つ機能は以下の通りである.

- アクティビティ図の記述が可能な環境
- アクションの修正基準を表示する機能

これらを用いることで、アクティビティ図の記述時や、見直し時に初級者であったとしても修正が必要なアクションかどうかを見分けることができる.

## 5.評価項目

今研究では実験の回答結果を評価項目に基づいて採点を行う. 評価項目は大きく分けると以下となる.

- 図に余分な要素が書かれていないか
- アクターが正しく書かれているか
- アクションが正しく書かれているか
- 遷移がどの図につながっているか

これらの項目を、実験で出題する図の要素に照らし合わせ、さらに細かく分ける. アクションについては修正基準と照らし合わせて採点を行う.

## 6.実験

モデカツ！と AmaterasUML を用いて比較実験を行い有効性の検証を行った. 導入のしやす

さや操作性の高さなどの理由から AmaterasUML を本実験の比較対象と選定した。

実験の概要を以下に示す。

- 被験者:日本工業大学情報工学科の学生 14人
- 問題数:2問
- 採点方法:評価項目を基に採点

実験の流れを以下に示す。

1. 対象者をランダムな2グループに分ける
2. 資料を配布し、解説を行う
3. 実験開始(50分)
4. 実験終了後アンケート回答させる

## 7.結果と考察

表1より、モデカツ!は AmaterasUML と比べ合計点が高くなっていることが確認できた。合計点が高くなっている原因として、アクションの合計の項目で点数に大きな差が出ている。よって、被験者はアクションについて理解しきれていない箇所が多く、アクションについて確認できるモデカツ!の点数が高くなったのだと考えられる。

AmaterasUML を使用したグループの点数が低かった原因として、ガードをアクションとして記述していたことや、アクションが「メッセージを送る相手を決め」のような口語になっていたため点数が低くなっている。

モデカツ!の場合はアクションの修正基準を表示する機能があるため、アクションを正しく記述することや、ガードの内容をアクションとして記述しても、間違いとして気付くことができたため、間違いの数が少なく点数が高くなっていると考えられる。

AmaterasUML を使用したグループにもアクションの修正基準を提示することで、ガードをアクションとして記述していた問題やアクションが口語になっていた問題を解決することができると考えられる。

## 8.結論と今後の課題

本研究では初学者がアクティビティ図を学習する際、動機をアクションとして記述してしまう問題やガードに書くべき内容をアクションとして記述してしまう問題を解決することを目的とし、記述環境を構築し、アクションの修正基準を表示させ、処理の抜けや無駄な処理に気付くことができる支援機能の開発を行った。

モデカツ!と AmaterasUML と比較実験を行った結果、初級者でも見直し時にアクションの間違いに気付くことができ、図を修正することができた。

今後は目的の達成に必要なアクションの欠落に関して考慮しなかったため、対応していく必要があると考えられる。

## 参考文献

- [1] 松井 駿介, 奥田 博隆, 式見 遼, 野呂 惇, 岡田 康治, 小形 真平, 松浦 佐江子, ” 初学者を対象としたユースケース記述の自動評価による学習支援, ” 信学技報. KBSE, 知能ソフトウェア工学 112(165), pp139-144, 2012-07-20.
- [2] 田中 亨祐, 黒木 裕太, 横山 孝典, 兪 明連, ” アクティビティ図生成機能を付加した Simulink・UML モデル変換ツール, ” 情報科学技術フォーラム講演論文集, 14(1), pp261-262, 2015-08-24.

表 1 実験結果

	モデカツ!		AmaterasUML	
	問題1	問題2	問題1	問題2
アクターの合計	20/20	20/21	20/20	21/21
アクションの合計	53/70	52/63	28/70	27/63
分岐の合計		7/7		4/7
ガードの合計		14/14		8/14
遷移の合計	63/91	72/98	68/91	64/98
その他の合計	68/70	69/77	70/70	67/77
全体の合計	235/259	266/280	208/259	220/280