

UML と VDM を用いたテストケース生成支援に関する研究

— 図書管理システムを事例に —

2019SE076 徳永圭吾

指導教員：張漢明

1 はじめに

ソフトウェア開発におけるテストは妥当性確認と正当性検証で重要になる。しかし、テストケース生成には課題が存在する。それは手作業でテストケースを作成することとテストケース生成に必要な情報が資料が分散されていたり、陽に記述されていないことにより獲得が困難であることである。

そのために、本研究の目的は、UML 記述を仕様として、UML 記述と VDM 記述からテストケース生成について考察することである。UML 記述を仕様とみなし、VDM 記述にテストケース生成のための情報が集約されていると想定した。これに基づいて本研究の課題は、テストケースに必要な情報の整理とテストケース生成支援技術の検討とする。

上記の課題解決のため、テストケース生成に必要な情報とテストケース自動生成のための課題を整理する。本研究では、UML 記述にはテストケース生成の情報が不十分であり、VDM 記述にはテストケース生成のための情報が存在するため、UML 記述と VDM 記述からテストケース生成について考察することで整理する。

2 関連技術/先行研究

2.1 UML

UML[3] は OMG(Object Management Group) によって、採用された標準のモデリング言語であり、1 つのダイアグラムで、対象のシステムのある側面を記述し、複数種類のダイアグラムを用いて、システム全体を記述する。

2.2 VDM

VDM[2] は仕様を数学の記法を用いて厳密に記述するための形式手法の 1 つである。VDM-SL はモデル規範型の形式仕様言語で、対象の状態をモデル化して状態の変化を操作で記述する。VDM は無償の開発環境が整備されており実際のソフトウェア開発でも利用されている言語である。

2.3 テスト技術

ソフトウェアテストに関して数多くの定義が存在するが、一番の本質的な部分は、「実際どうなっているのか」と「本来はどうあるべきか」を比較するプロセスだということである [1]。テストを行う特定の条件を具体化したものがテストケースである。有効性と効率を追及するなら、テストケースをいい加減にでっち上げるのではなく、「設計」することが必要である。

2.4 状態マシンテスト

状態マシン図、状態遷移表を基にテストケースを作成する。状態マシン図はテストケース作成に利用可能であり、テストケースは状態遷移表から読み取り可能である。状態マシン図は、システムが受け取って処理するイベントを、システムの応答と一緒に文書化するものである。以前に何が起きたのかとか、処理の有効・無効な順番が存在したかなどを、システムが記憶しておかなければならない場合、状態マシン図は優れたツールとなる。

2.5 BWDM

BWDM[4] は、形式手法を用いたソフトウェア開発における、テスト工程の作業効率化を目的として作られた VDM++ 仕様を対象としたテストケース自動生成ツールである。VDM 記述からテストケースを生成する研究として、BWDM の試作及び有用性の上昇について研究を行っている。例えば、ドメイン分析テストのためのテストケース生成手法を提案し、既存の BWDM に適用することで、BWDM を拡張した。これにより既存の BWDM の問題点が解決し、VDM++ 仕様からドメイン分析テストのためのテストケースを自動生成することが可能になったため、BWDM の有用性とテストケース生成効率が向上した。

3 事例：図書管理システム

本研究では、事例として研究室の図書システムの記述を対象とした。図書管理システムとは、書籍の管理や書籍に対して管理者や利用者が行う操作などを管理するシステムである。以下では、図書管理システムの登録機能の記述を示す。

3.1 状態マシン図

図書管理システムの登録機能の画面遷移に関する状態マシン図を図 2 に示す。状態マシン図では書籍登録の状態遷移と操作の関係を示している

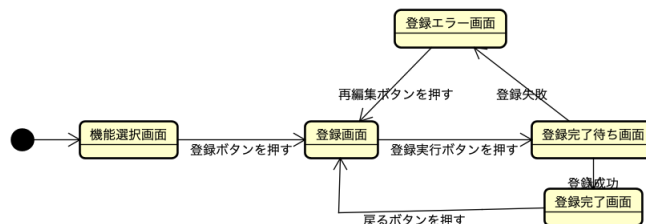


図 1 図書管理システム 状態マシン図

3.2 VDM 記述

上記の状態マシン図の VDM 記述を以下に示す。型定義ブロックでは、書庫を構成する「書籍リスト」と画面の状態を示す「画面型」を定義している。

```
types
  書籍型 = token;
  書籍リスト型 = seq of 書籍型;
  画面型 = <初期画面>| <登録画面>| <登録完了待ち画面>| <登録エラー画面>| <登録完了画面>;
```

操作定義ブロックでは、各遷移ごとに操作名をつけ、状態間の遷移の関係を事前条件、事後条件にすることで、VDM-SL 記述に対応させる

```
operations
  登録実行 : 書籍型 ==> ()
  登録実行 (書籍) ==
    (登録書籍 := 書籍;
     画面 := <登録完了待ち画面>)
  pre 画面 = <登録画面>
```

4 テスト技術の適用

テストケースは操作の系列と期待値で構成されている。ここでの課題は、操作の系列の作成である。

この課題に対して、テスト技術を適用させることで解決させる。操作の系列を状態マシンテストで生成、期待値は VDM の操作の系列の結果から生成する。作成した状態遷移テストのテストケースをカバレッジのレベル別に紹介する。

- すべての状態を少なくとも 1 回訪れるような 1 組のテストケース
- すべての遷移を少なくとも 1 回実行する 1 組のテストケース
- すべてのパスが少なくとも 1 回実行する 1 組のテストケース
- 1 回のイベントで遷移する経路をすべて網羅する 0 スイッチカバレッジ
- 前状態から後状態までに状態を 1 つ経由させる 1 スイッチカバレッジ

0 スイッチカバレッジのテストケースを表 1 に示す。

表 1 状態マシンテスト 0 スイッチカバレッジ

現在の状態	操作	次の状態
機能選択画面	登録機能選択	登録画面
登録画面	登録実行	登録完了待ち画面
登録完了待ち画面	登録成功	登録完了画面
登録完了待ち画面	登録失敗	登録エラー画面
登録完了画面	戻る	登録画面
登録エラー画面	再編集	登録画面

5 考察

5.1 テストケース生成に必要な情報

操作系列は状態マシン図の状態テストのカバレッジの基準を利用することで、入力 VDM 記述の情報とインスタンス数で求められる。これらの情報を用いることで、テストケースを生成する見込みを得ることができる。

5.2 操作列の表現

VDM 記述における登録実行 (書籍 A) のインスタンスをパラメータ化すると登録実行 (書籍) となり、機能は登録実行 (書籍) の後で成功するか失敗するかを判断するものとなり、操作列で表現することで、過去に登録実行 (操作) があったかに関して、ある場合は失敗し、ない場合は成功するため、過去の情報が必要となることが分かる。さらに過去の情報をパラメータ化すると、登録実行 (書籍集合、操作) となり、これの後で成功か失敗を判断し、書籍集合は過去の登録された書籍の集合であり、登録実行の機能で表現できる。

5.3 正当性検証のための支援

正当性検証は、正しい設計かを判定することである。判定の基準は仕様であり、設計が仕様のテストケースを満たしているか調べる。仕様は操作のパスと VDM 記述があれば自動生成することができる。操作のパスの表現方法が課題となった。

6 おわりに

本研究では、図書管理システムの事例を用いて、状態マシン図と VDM 記述からテストケース生成について考察した。その結果、テストケースを自動生成するための方針を仕様=操作系列+VDM 記述というように定めることができることが分かった。今後の課題として、UML と形式仕様言語による操作列の表現の形式化が挙げられる。課題に対するアイデアの候補として、操作の系列は操作の実行順序を表示することができる「プロセス代数」を利用することが考えられる。

参考文献

- [1] リー・コーブランド (宗 雅彦 訳) : 『はじめて学ぶソフトウェアのテスト技法』日経 BP 社, 東京, 2005
- [2] 荒木啓二郎・張漢明 : 『プログラム仕様記述論』情報処理学会, 東京, 2002
- [3] マーチン・ファウラー (羽入田栄一 訳) : 『UML モデリングのエッセンス第 3 版』株式会社翔泳社, 東京, 2005
- [4] 平木場 風太 片山 哲郎 : 『VDM++ 仕様を対象としたテストケース自動生成ツール BWDM へのドメイン分析テストの適用』ソフトウェアエンジニアリングシンポジウム 2019, pp128-134,