VDM を用いた状態マシン図の妥当性確認支援に関する考察 ―図書管理システムを事例として―

2019SE037 盛巧樹
指導教員：張漢明

1　はじめに

ソフトウェア開発において，仕様はプログラムの正当性を検証する基準であり，妥当性確認の判断においても重要な役割を担っている．仕様の妥当性確認には仕様が実行できることが有効である．しかし，UML 図では実行に必要な記述が足りず，一般的には実行することができない．形式仕様である VDM には，実行可能な形式仕様言語の VDM-SL がある．本研究の目的は，VDM を用いた状態マシン図の妥当性確認支援である．状態マシン図とは，状態遷移の視点からソフトウェア開発においてよく用いられる動的構造の表現の一つである．状態マシン図の振る舞いを確認するために VDM-SL にて，仕様の実行をおこなう．本研究の目的を達成するために，以下の問題が挙げられる．

- 状態マシン図を実行するために必要な情報は何か
- VDM-SL 記述を支援するために必要な技術は何か

以上を踏まえ，方法として，状態マシン図と VDM-SL 記述の対応関係を確認し，状態マシン図より形式的に変換できる部分とできない部分を確認する．また，変換できない部分を記述するための支援技術を考察する．

2　関連技術・先行研究

2.1　状態マシン図

状態マシン図は，システムの振る舞いを記述するための技術としてよく知られてる [1]．システムの動的な振舞いをグラフィカルに記述したもので，ある状態と別のある状態への遷移を表現する．画面遷移の記述で利用される場合が多い．

2.2　VDM-SL の概要

VDM(Vienna Development Method) とは，1970 年代にウィーン研究所において開発された形式手法である [2]．VDM はモデル規範型と呼ばれる形式手法の 1 つである．モデル規範型では，システムの内部構造の状態として定義して，状態の変化を操作を用いて記述する．

3　事例

3.1　図書管理システムの概要

図書管理システムとは，研究室が保有する書籍の管理や，書籍に対して管理者と利用者が行う操作などを管理するシステムのことである．図書管理システムは，5 つの機能があるとものとし，それぞれの機能の状態マシン図と VDM-SL を記述した．本要旨では，5 つある機能の 1 つ，書籍の「登録」についての状態マシン図と VDM-SL 記述を踏まえて説明する．

3.2　状態マシン図

図 1 の状態マシン図について説明する．書籍の「登録」とは，研究室が既に保有している書籍，または新しく保有することになった書籍を管理者が「図書管理システム」に登録することができる機能である．登録の機能では，主系列で 4 つ，代替系列で 1 つの状態を「画面」として作成した．一部抜粋した画面の内容を説明する．

- 登録完了待ち画面
  登録画面で実行ボタンを押し，遷移した先の画面となっており，登録の実行が開始され，実際に登録が完了されるまでの間に表示される画面である．登録が成功した場合は登録を完了し，登録完了画面へと遷移することができる．または，登録が失敗した場合は登録が失敗し，登録エラー画面へと遷移することができる．
- 登録完了画面
  登録完了待ち画面で成功ボタンを押し，遷移した先の画面となっており，登録が完了されたことを伝える画面である．戻るボタンを押すことで登録画面へ遷移することができる．

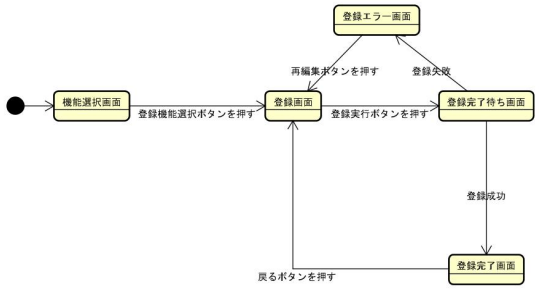

図 1　図書管理システム-登録の状態マシン図

3.3　VDM-SL 記述

記述した VDM-SL を説明する．書籍を書籍型という名前にし，トークン型として定義し，登録した書籍を格納する書籍の集合体を書籍型という名前にし，集合型として定義した．図 1 の「機能選択画面」などの 5 つの状態を画面型として定義した．

```
書籍型 = token;
書庫型 = set 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>;
```

次に状態定義について説明する．「初期画面」という状態が実行されている段階での画面や書庫，登録書籍の集合が値を持っていない状態にするため初期化状態定義をおこない初期化する．

```
state St of
    画面 : 画面型
    書庫 : 書庫型
    登録書籍 : [書籍型]
init s ==
    s = mk_St(<初期画面>, 空書庫, nil)
end
```

以下の記述は，登録が成功し，書庫に書籍が登録される操作である．事前条件と代入によって，登録完了待ち画面から登録完了画面への遷移を表現した．

```
    登録成功 : 書籍型 ==> ()
    登録成功(書籍) ==
        (書庫 := {書籍} union 書庫;
        画面 := <登録完了画面>)
    pre  画面 = <登録完了待ち画面>;
```

以下の記述は，「書籍を登録する」テストケースである．期待値としては「書籍 A を書庫に登録する」こと，画面が初期画面→登録画面→登録完了待ち画面→登録完了画面→登録画面へと遷移することである．

```
    テストケース1 : () ==> ()
    テストケース1() ==
        (初期化();
        登録機能選択();
        登録実行(書籍A);
        登録成功(書籍A);
        戻る_登録画面());
```

このテストケースを実行することにより状態マシン図の妥当性を確認することができる，

4 考察

4.1 構文レベルの関係

クラス図

クラスと引数，関連，操作は，クラス図より形式的に VDM-SL へ変換できた．対応関係の例は以下に示す．

- クラス ⇒ 型名
- 操作の引数 ⇒ 操作の引数
- 関連（1 対多）⇒ 集合型

状態マシン図

図 2 に示した一般的な状態遷移に対する VDM 記述を示す．遷移の「操作 X」は VDM-SL 記述の操作に対応し，次画面への遷移は，画面変数の代入に対応している．前画面は事前条件に対応している．

```
画面型 = <初期画面>| <前画面> | <次画面> | ...
state St of
    画面 : 画面型
...
init s == s = mk_St(<初期画面>, ...)
end
```

```
操作X(書籍) ==
    (...
    画面 := <次画面>)
pre  画面 = <前画面>;
```

図 2 状態マシン図

4.2 意味レベルの関係

4.2.1 UML では記述されないもの

UML は構文レベルの規定であるとみなすと状態マシン図では状態と操作の関係を記述している．状態マシン図は状態と操作の関係を記述しており，第 4.1 節の VDM-SL 記述と図 2 の対応関係は第 3.3 節の VDM-SL 記述と比較すると，「機能」の部分が記述されていないのがわかる．抽象度の高い段階ではまだ決定していない機能がある．機能はクラス図に記述されているが，その意味が厳密に記述されていない場合が多い．したがって，機能の意味を定義する必要がある．

4.2.2 記述支援

開発の早期で抽象度の高い段階では，状態マシン図の記述には，クラス（型）や操作の意味は記述されていない．状態マシン図の妥当性を確認するためには，それらの型と操作の意味を簡潔に記述する必要がある．数学に基づいた概念を用いて記述することが困難である．したがって，典型的に自然言語で表現される表現と VDM 記述とを対応付けた用語集の提示が有効であると考えられる．

5 おわりに

本研究では，図書管理システムという事例をもとに，UML では記述されないが，実行するために記述する必要がある部分の確認した．UML では記述されない部分を VDM-SL へ変換する際の支援方法として，用語集が有効であると考えられる．今後の課題として，典型的な自然言語と対応付けられた VDM 記述の収集が挙げられる．

参考文献

[1] マーチン・ファウラー (羽入田栄一 監訳)，“UML モデリングのエッセンス第 3 版門”，翔泳社，2005.

[2] 石川冬樹，“VDM++ による形式仕様記述”，近代科学社，2011.

[3] 荒木啓二郎，張漢明，“プログラム仕様記述論”，オーム社，2002.