

## さまざまな視点に合わせた仕様書の作成・維持の支援手法

## Support method for preparation and maintenance of specifications tailored to various perspectives

2016年度 SQiP 研究会 演習コース II 「形式手法と仕様記述」

Exercise Course II "Formal Method and Specification Description" of SQiP Study Group in 2016

○酒井 雄太<sup>1</sup>石川 冬樹<sup>2</sup>栗田 太郎<sup>3</sup>

○Yuuta Sakai

Fuyuki Ishikawa

Taro Kurita

**Abstract** In the development of business systems, it is necessary to represent the contents of the specification in a variety of perspectives and levels of abstraction to promote discussion and validation by stakeholders with different concerns. A specification document, a description aggregating the specification contents, is usually created by engineers. Thus, it leads to issues such as difficulty for non-technical customers and inconsistency between specification descriptions prepared for various viewpoints of readers. In response to these issues, this work proposes a method for supporting construction and maintenance of specification documents. In the method, a specification description is constructed by using VDM (Vienna Development Method), one of formal methods, and then descriptions for various viewpoints of readers are automatically generated. This method leads to specification documents that prevent inconsistency between multiple descriptions and are suitable for various readers such as non-technical person.

## 1. はじめに

## 1.1 業務システムの開発

著者は、業務システムの開発に携わった経験がある。業務システムの開発においては、関心事の異なるステークホルダーに対して、業務システムの機能や画面といった、仕様に関する情報をさまざまな視点や抽象度で表現し、議論や妥当性確認を促す必要がある。例えば、現場担当者においては、現場で業務システムをどのように使うかということについて興味があるので、業務システムの基本的な使い方の視点で表現した仕様を示すことが、議論や妥当性確認を促すことにつながる。

## 1.2 業務システムの開発における仕様書の課題

関心事の異なるステークホルダーに対して議論や妥当性確認を促す必要がある場面において、例えば、技術者と技術者ではない読み手とのコミュニケーションのために、仕様書が用いられる。仕様書は、通常、技術者が作成・維持し、仕様に関する情報を仕様記述（仕様をある形式で表現したもの）として集約したものである。

このような仕様書の課題のひとつは、技術者ではない読み手には不適切な形式で仕様が表現されるということである。例えば、読み手が現場担当者である場合、以下に挙げるような問題が生じうる。

- (1) 読み手は、どの記述を読むべきかわからないため、仕様書の全ての記述に向き合う必要があり、限られた時間内に理解することができなかつたり、レビューを完了させたりすることができない
- (2) 読み手は、具体的な動作を理解したいとき、複数の記述を横断的に読む必要がある

---

<sup>1</sup> Freelance engineer, yuuta.sakai@wellsd.jp

<sup>2</sup> 国立情報学研究所, National Institute of Informatics

<sup>3</sup> ソニー株式会社, Sony Corporation

- (3) 読み手は、内部のロジックや外部の業務管理システムとの連携についての記載については、関心がない
- (4) 読み手は、普段行なっている業務がどのように変化するのかを想像することができず、業務が楽になるのか分からない

上記の問題を扱うために、読み手に合わせた仕様記述を用意することが多い。しかし、読み手のさまざまな視点に合わせた複数の仕様記述間に不整合が生じるという課題がある。不整合が生じる原因として、ある仕様記述から、視点や抽象度を変えた仕様記述を作成する際に、欠陥が混入することや、ある仕様記述を更新したときに、その仕様記述から作成されている仕様記述が更新されないことなどが挙げられる。

## 2. 本研究の提案

### 2.1 さまざまな視点に合わせた仕様書の作成・維持の支援手法

そこで本研究では、仕様に関する一通りの情報が集約されたものとして、形式手法のひとつである VDM (Vienna Development Method) [1] によって形式仕様記述[2]を行い、その仕様記述から機械処理によって、読み手のさまざまな視点に合わせた仕様記述を生成することによる、仕様書の作成・維持の支援手法を提案する。VDM においては、仕様記述言語によりシステムのモデルを厳密に記述する。本研究では、VDM による厳密な仕様記述が機械処理可能であることを活かし、その記述から状態遷移図やシーケンス図といった表現形式の仕様記述を生成する手法を提案する。

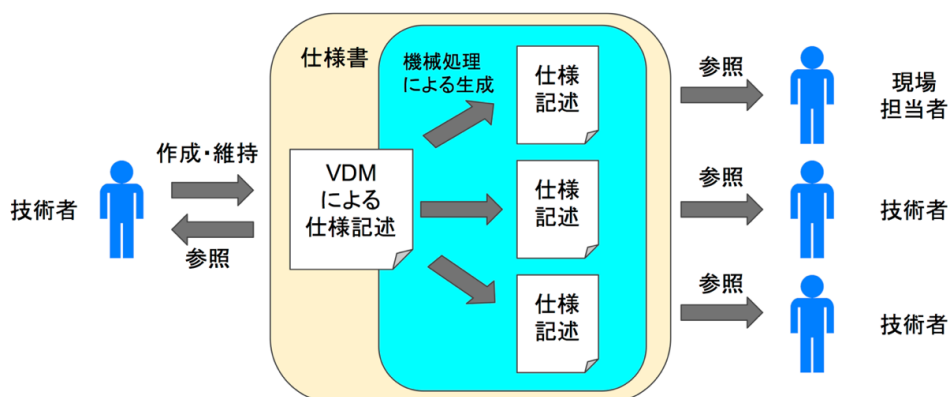


図1 提案手法の概念

図1にて、本提案手法の概念を説明する。VDMによる仕様記述は、技術者により作成・維持され、仕様に関する一通りの情報が集約されたものである。これは、さまざまな視点で読むことができるが、技術者ではない読み手には不適切な形式による表現である。そこで、VDMによる仕様記述から、現場担当者や技術者といった読み手の視点に合わせて、機械処理による仕様記述の生成を行う。

本提案手法により作成・維持される仕様書は、複数の仕様記述間の不整合を防ぎつつ、現場担当者や技術者といったさまざまな読み手に適した仕様記述を含むという特徴を持つ。なぜならば、ある仕様記述から機械処理によって生成された仕様記述は、欠陥が混入する可能性が低く、機械処理を自動化することによって、常に整合性を維持することができるからである。さらに、仕様書に、読み手や状況に応じた仕様記述が存在し、読むべき仕様記述を取捨選択することが容易だからである。

また、機械処理により生成された仕様記述の修正は、VDMによる仕様記述を修正することで実現する。このような手順とする背景として、通常、機械処理により生成された仕様記述は、VDMによる仕様記述に比べ情報量が少ないものとなり、情報量の少ない仕様記述の変更から、情報量の多い仕様記述の変更を一意に決めることが困難だということがある。この手順の具体的

な運用は、レビューにおいて、参加者は機械処理により生成された仕様記述に対して赤入れを行い、技術者はその結果と一致するように、VDMによる仕様記述を修正するといったことが考えられる。

## 2.2 本研究の実装

本研究では、本提案の第一歩として、一般的な仕様記述の形式であると考えられるシーケンス図と画面遷移図による、本提案手法に基づいた以下の2つの仕様記述生成機能を実装する。

- (1) 現場担当者向けの仕様記述（シーケンス図）の生成機能
- (2) 技術者向けの仕様記述（画面遷移図とシーケンス図）の生成機能

これらの仕様記述生成機能によって生成された仕様記述は、現場担当者や技術者が、業務システムについてある視点から理解することを促す。表1に、視点の具体例を挙げる。

表1 本研究で生成の仕組みを検討する仕様記述とその視点

仕様記述	図の種類	視点
現場担当者向けの仕様記述	シーケンス図	業務システムの基本的な使い方を理解する
技術者向けの仕様記述	画面遷移図	画面遷移に関する仕様を俯瞰し、全体像を理解する
	シーケンス図	業務システムの具体的な動作の一例を通して、仕様の理解を促す

## 3 例題を用いた提案手法の説明

### 3.1 VDMによる仕様記述

本提案手法を構築、試行するにあたり、機器のメンテナンス業務を管理するシステムを例題として設定し、例題についてのVDMによる仕様記述を行なった。

表2 本提案手法のVDMによる仕様記述に対する要求事項

要求	定めたルール
各画面に関する仕様が記述されており、画面毎にまとまった形で容易に抽出できること	各画面を、1つのVDM++クラスで記述する(ルール1)
各UI要素に関する仕様が記述されており、画面に紐づく形で容易に抽出できること	各UI要素を、画面について記述したVDM++クラスのインスタンス変数で表現する(ルール2)
画面遷移に関する仕様が記述されており、遷移元と遷移先の画面名が容易に抽出できること	画面遷移を、画面について記述したVDM++クラスの操作の戻り値にて、遷移先の画面のインスタンスを返すことで表現する(ルール3)
基本的な使い方が記述されていて、また、それを実行したときの内部的なことも含めた振る舞いについての情報を抽出できること	ユーザによる一連の操作の一例を記述する(ルール4)
	全ての操作に対して、以下の形式によるログ出力を記述する(ルール5-1, 5-2) - <ログ出力元クラス名>.<操作名> - <ログ出力元クラス名>.<操作結果>

表3 本研究におけるVDMによる仕様記述のクラスの種別構成

クラスの種別	記述される情報
アクター	ユースケースシナリオ ユーザによる画面の操作
画面	UI部品や画面遷移といった、画面に関わる仕様 図による画面のイメージ
システム	画面仕様以外の、内部的なことも含めた仕様

本研究では、仕様記述言語としてVDM++言語を採用した。また、機械処理による仕様記述の生成を実施可能とするために、表2の要求事項を満たすような記述とした。さらに、このような要求事項を満たしつつ、本研究の例題についてのVDMによる仕様記述を作成するために、表3で示すようなクラスの種別構成とした。

これらを踏まえた、具体的な仕様記述の例を図2と図3に示す。前述の要求事項との対応は、仕様記述中のコメントとして示した。

```

-- このクラスは ログイン画面 についての仕様記述である(ルール1)
class ログイン画面 is subclass of 画面
instance variables
-- UI要素をインスタンス変数として表現する(ルール2)
public IDフィールド : token := mk_token("NULL");
public パスワードフィールド : token := mk_token("NULL");
<省略>
--画面遷移を伴う操作の戻り値は 次の画面のインスタンスである(ルール3)
public ログインボタンを押下する : () ==> 画面
ログインボタンを押下する() ==
(
  --操作に対して、VDM++を実行した際に出力されるログが埋め込まれている(ルール5-1)
  io.print("ログイン画面. ログインボタンを押下する\n");
  let
    sys = システム`システムインスタンス,
    認証情報確認結果 = sys. 認証情報を確認する(IDフィールド, パスワードフィールド)
  in
    if 認証情報確認結果 = true
    then (
      --操作結果に関するログが埋め込まれている(ルール5-2)
      io.print("ログイン画面. タスク一覧画面に遷移する\n");
      return new タスク一覧画面();
    )
    else return self
  )
<省略>
end ログイン画面

```

図2 ログイン画面のVDMによる仕様記述の例

```

-- この操作はユーザによる一連の操作の一例である(ルール4)
public 基本シナリオを実行する : () ==> ()
基本シナリオを実行する() ==
(
  io.print("ユースケース. 基本シナリオを実行する\n");
  現ユーザ. 起動する();
  現ユーザ. ログインする(mk_token("Login"), mk_token("Password"));
  現ユーザ. タスクを選択する(mk_token("顧客 0001"));
  現ユーザ. タスク情報を取得する();
  現ユーザ. 作業内容を登録する(mk_token("Work1"));
  現ユーザ. 交換部品を登録する(mk_token("P0001"), mk_token("Parts1"));
  現ユーザ. 作業内容を取得する();
  現ユーザ. 交換部品を取得する();
  現ユーザ. 作業結果を送信する();
)

```

図3 画面に対する一連の操作をVDMにより記述した例

### 3.2 機械処理による仕様記述の生成

次に、先に作成した VDM による仕様記述から、機械処理による仕様記述の生成を行なった。

本研究では、アクター、画面、システムに関する VDM による仕様記述と、ユースケースシナリオを記述した VDM 仕様記述を実行することで得られるシーケンス情報を入力とし、現場単調者向けの仕様記述(シーケンス図)と技術者向けの仕様記述(画面遷移図とシーケンス図)を生成する手法を提案した。この手法の全体像を図4に示す。

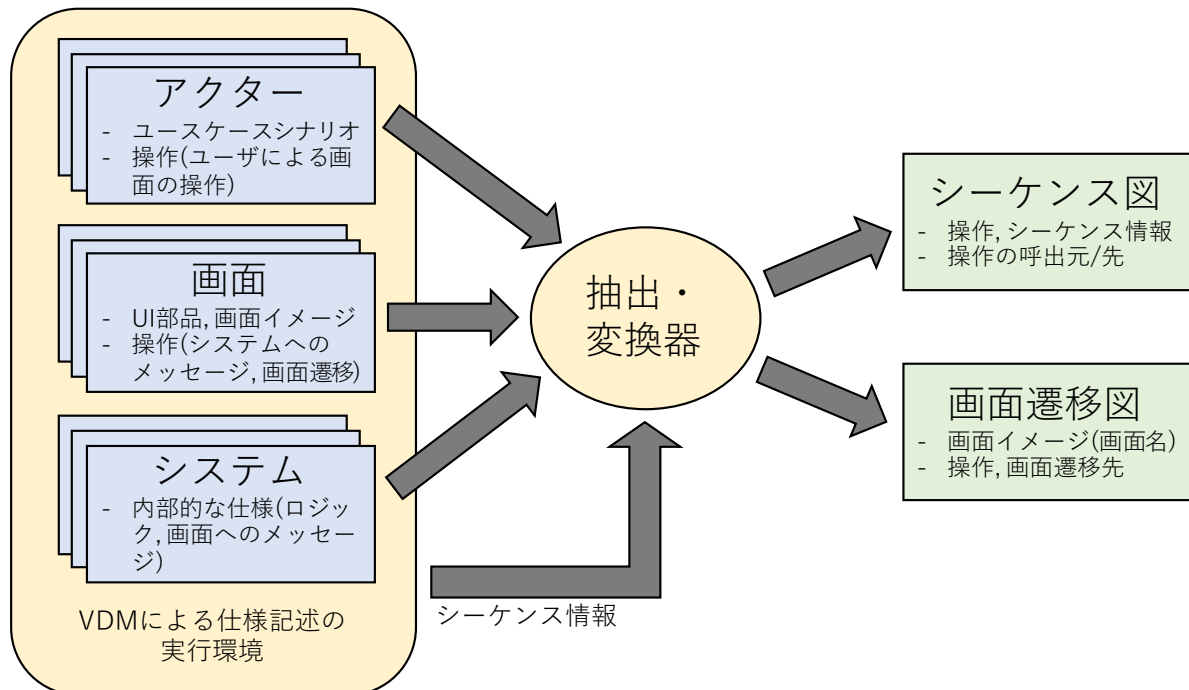


図4 本研究の機械処理による仕様記述生成手法の全体像

#### 3.2.1 シーケンス図の生成

本提案手法によるシーケンス図は、前述のシーケンス情報に対してシーケンス図への変換処理を行うことで得られる。本研究の実装では、シーケンス情報の取得に Overture Tool [3] を利用した。これにより得られるシーケンス情報の例を図5に示す。この例では、コメント文により図2と図3におけるVDM仕様記述のルールとの対応も示している。また、シーケンス図への変換処理は、シーケンス情報から、クラス間の操作呼び出し情報を抽出し、さまざまな図の生成ツールである PlantUML [4] が解釈可能な形式のファイルを生成する手段と、そのファイルを入力として PlantUML を実行し、画像データとしてシーケンス図を得る手段で構成される。さらに、この変換処理は、読み手に合わせて、シーケンス図に掲載する情報を取捨選択することができる。例えば、現場担当者はシステムに関する情報は不要であるので、操作呼び出し情報の抽出処理において、システムクラスの操作呼び出し情報は除外する、といったことである。これらにより、生成されたシーケンス図の例を図6に示す。この例からは、図5で示したシーケンス情報の各行がシーケンス図の各メッセージに対応していることがわかる。

```

ユースケース. 基本シナリオを実行する #図3 ルール4
ユーザ. 起動する
ホーム画面. アイコンを押下する
ホーム画面. ログイン画面に遷移する
ユーザ. ログインする
ログイン画面. IDを入力する
ログイン画面. パスワードを入力する
ログイン画面. ログインボタンを押下する #図2 ルール5-1
システム. 認証情報を確認する
システム. 認証OK
ログイン画面. タスク一覧画面に遷移する #図2 ルール5-2
システム. 顧客名一覧を返す
システム. 顧客名一覧
<省略>
ユーザ. 作業結果を送信する
メンテナンス作業画面. 作業結果送信ボタンを押下する
システム. タスク情報を送信する
メンテナンス作業画面. タスク一覧画面に遷移する

```

図5 VDMによる仕様記述のシーケンス情報の例

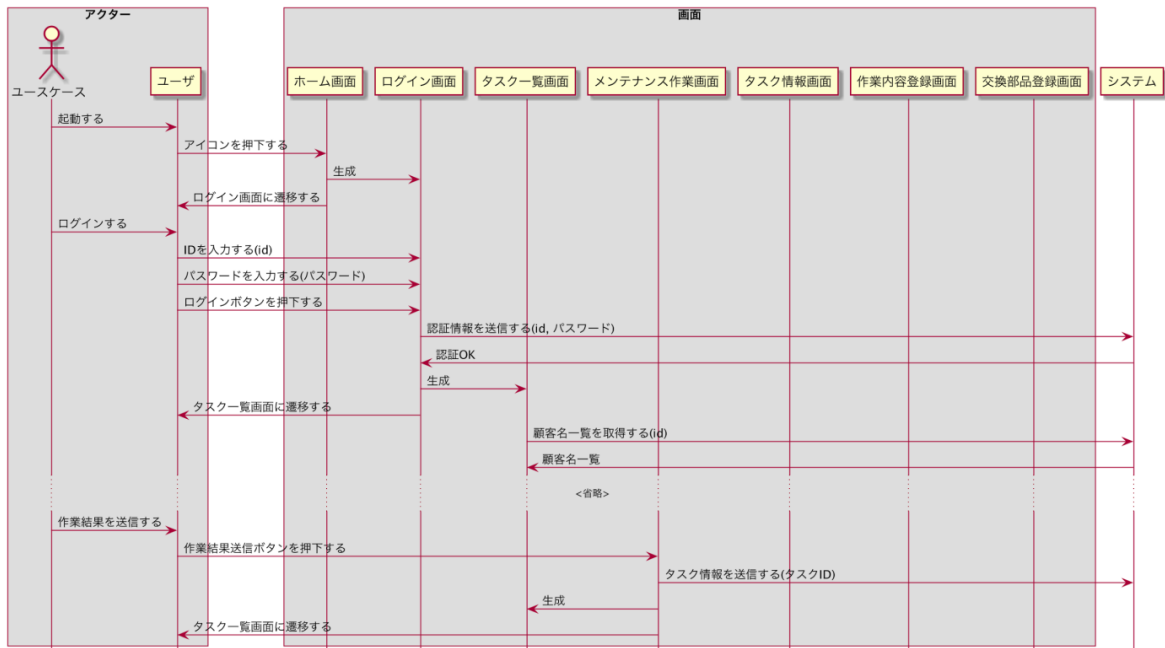


図 6 機械処理により生成された仕様記述(シーケンス図)の例

### 3.2.2 画面遷移図の生成

本提案手法による画面遷移図は、VDM による仕様記述に対して変換処理を行うことで得られる。本研究の実装における画面遷移図への変換処理は、VDM による仕様記述のうち、画面の仕様にあたるものを抽出し、さらに、それらに記述されている UI 要素と画面遷移に関する情報を抽出した上で、PlantUML が解釈可能な形式のファイルを生成する手段と、そのファイルを入力として PlantUML を実行し、画像データとして画面遷移図を得る手段で構成される。なお、画面遷移図の生成は、アクティビティ図生成機能を利用した。これらにより、生成された画面遷移図の例を図 7 に示す。この例では、図中の吹き出しにより図 2 における VDM 仕様記述のルールとの対応も示している。

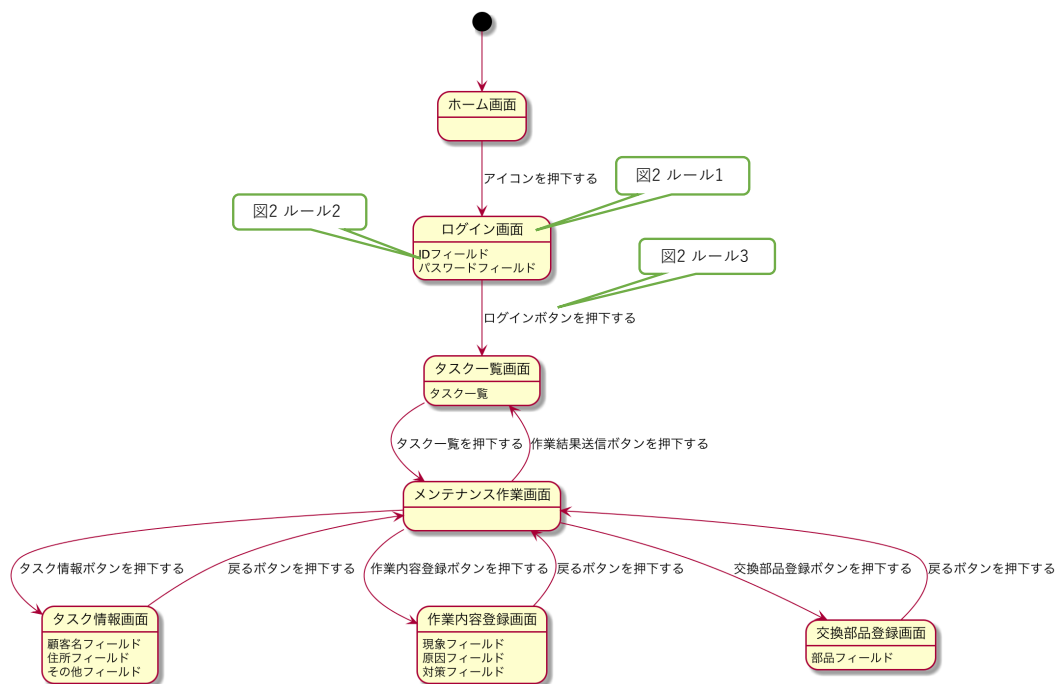


図 7 機械処理により生成された仕様記述(画面遷移図)の例

この画面遷移図からは、業務システムの画面遷移に関する仕様を俯瞰し、全体像を理解することができる。例えば、本研究の実装では、画面毎に仕様記述がされているが、画面遷移の仕様を理解するためには、それらの仕様記述を横断的に読む必要がある。本図ならば、画面遷移に関する仕様がひとつの図にまとまっているので、理解することが容易である。

## 4. 評価と考察

### 4.1 生成機能について

本研究では、本提案の第一歩として、一般的な仕様記述の形式であると考えられるシーケンス図と画面遷移図による、現場担当者と技術者向けの仕様記述生成機能を実装した。ただし、読み手にどのような情報をどのような視点や形式で見せるべきかについては、別途追求が必要であると考えられる。

しかし、本研究の実装による仕様書生成機能は、本提案で期待する初期の効果を確認することができたと言える。具体的には、VDMによる仕様記述から、機械処理により仕様記述（シーケンス図と画面遷移図）を生成することで、複数の仕様記述間の不整合を防ぐことができた。シーケンス図においては、掲載する情報を取捨選択することにより、読み手の視点に合わせた仕様記述を生成することができた。例えば、現場担当者はシステムに関する情報は不要であるので、操作呼び出しログの抽出処理において、システムクラスの操作呼び出しログは除外する、といったことである。普段の業務では、さらに、図による画面イメージといった情報を含めて、操作マニュアルのような形で示すこともあるが、後述する提案手法の拡張によって、それは実現可能だと言える。画面遷移図においては、開発者が、業務システムの画面遷移に関する仕様を俯瞰し、全体像を理解することを促す仕様記述を生成することができた。具体的には、本研究の実装では、画面毎に仕様記述がされているが、画面遷移の仕様を理解するためには、それらの仕様記述を横断的に読む必要がある。本図ならば、画面遷移に関する仕様がひとつの図にまとまっているので、理解することが容易である。

さらに、本仕様書生成機能は、仕様に関する一通りの情報が集約されたVDMによる500行程度の仕様記述から、さまざまな視点や抽象度で表現した複数の仕様記述を機械処理によって生成することができた。つまり、仕様書の作成・維持にかかる作業の削減効果を示すことができたと言える。

### 4.2 提案手法について

本提案手法の目指すことは、読み手のさまざまな視点に合わせた仕様記述を生成し、かつ、複数の仕様記述間の不整合を防ぐことである。そこにたどり着くためには、本提案手法を拡張することや、本提案手法を開発プロセスに適用するといったことが必要であると考えられる。

本提案手法の今後の拡張として、例えば、本研究で作成したVDMによる仕様記述であっても、「システムが持つインターフェースの一覧」のような仕様記述は簡単な拡張で実現できると考えられる。また、前述したVDMによる仕様記述の要求事項を拡張し、図による画面イメージと組み合わせ、静的プロトタイプ[5]のような仕様記述を生成することや、業務に関することがらを含めたVDMによる仕様記述を作成して、業務フロー図[6]のような仕様記述を生成することの実現方法は容易に考えられる。

さらに、対象とするシステムを増やすことも可能だと考えられる。本研究では典型的な業務システムを対象にしたので、VDMによる仕様記述に対する要求事項として、画面に関する仕様などを設けたが、組み込みシステムといった、別のシステムを対象とする場合は、前述のものとは異なる要求事項を定める必要がある。しかし、本提案手法の考え方は、別のシステムを対象とする場合にも活用できる。

本提案手法を開発プロセスに適用するためには、ステークホルダーに応じた視点を選定すること、視点に合わせた仕様記述を生成するための手法の検討と実装を行うこと、VDMによる仕様記述を更新したら、機械処理による仕様記述を再生成するといった実際の運用についてのこと、読み手からのフィードバックに基づき提案手法を改善するといったことを定める必要がある。

## 5. おわりに

本研究では、形式手法のひとつである VDM によって仕様記述を行い、その仕様記述から機械処理によって、ステークホルダーのさまざまな視点に合わせた仕様記述を生成することにより、仕様書の作成・維持作業の支援手法を提案した。

本研究では、機械処理によって画面遷移図とシーケンス図を生成する、本提案手法に基づいた仕様記述作成機能の実装を行なった。これらの仕様記述は、技術者や現場担当者に役立つ一般的な仕様記述の形式であると考えた。本研究の実装は、本提案手法の第一歩となるものであるが、その特徴である、複数の仕様記述間の不整合を防ぎつつ、現場担当者や技術者といったさまざまな読み手に適した仕様記述を生成可能であるということを示し、また、今後の拡張の可能性を示すことができた。また、関心事の異なるステークホルダーに合わせて多くの文書を作成・維持することは大変だが、形式手法はそのような課題に対しても有効であるということを示すことができた。

今後は、本提案手法の拡張によって、多様なステークホルダーとのより深い議論や妥当性確認のための仕様書を生成できるようにすることや、技術者ではない読み手による評価などを行い、提案内容をより良いものにするを通じて、本提案を要求獲得の道具として使えるものに育てていきたい。

## 6. 今後の展望

下記に、本研究における今後の展望を列挙する。

- よりさまざまな仕様記述への生成機能を構築
- 他のシステムへの適用を通じた手法の洗練
- 技術者ではない読み手による評価

## 謝辞

本論文への取り組みは、日本科学技術連盟主催のソフトウェア品質管理研究会の 2016 年度 SQiP 研究会 演習コース II の研究成果を基に行ったものです。本論文の執筆に当たり、九州大学大学院 荒木啓二郎先生、2016 年度 SQiP 研究会 演習コース II 研究員の宮本陽子氏に大変お世話になりました。厚くお礼申し上げます。

## 参考文献

- [1] John Fitzgerald, Peter Gorm Larsen, Paul Mukherjee, Nico Plat, and Marcel Verhoef, 酒匂 寛(訳), VDM++によるオブジェクト指向システムの高品質設計と検証, 翔泳社 2010.
- [2] 荒木啓二郎, 張漢明: プログラム仕様記述論, オーム社, 2002.
- [3] Overture Tool <http://overturetool.org/>
- [4] PlantUML <http://plantuml.com/>
- [5] 川西 裕幸, 栗山 進, 潮田 浩, UX デザイン入門, 日経 BP 社 2012
- [6] 高安 厚思, システム設計の謎を解く, ソフトバンク クリエイティブ株式会社 2013