5A-04

エンタプライズ系業務システムの仕様をVDM++で記述する設計手法の提案

濱野 義満† 宗像 一樹‡ 鈴木 庸介† 野田 恵子†
富士通株式会社† 株式会社富士通研究所‡

1．はじめに

これまでエンタプライズ系の業務システム開発の分野では，業界大手各社ともに設計の標準化に長年取り組んでいるが，未だ業界標準となる設計標準はない．エンタプライズ系分野の標準化に形式手法を活用する取り組みは，仕様フレームワーク[1]の取り組みにより進歩した．一方，人とシステムの接点である画面に対する操作仕様の標準化に関しては事例が少なく，設計段階で画面操作を含めた仕様の記述および検証が属人的に行われ，品質の確保に課題がある．

そこで我々は前論文 [2]で整備した Java による仕様記述方法を VDM++に発展させ，画面の操作仕様を含む仕様フレームワークを開発した．これにより，データ処理仕様をはじめ画面仕様や操作仕様，業務仕様等を含めた仕様実行（仕様アニメーション）を実現することで，設計段階で効率的な形式検証を可能とした．

2．従来の課題

利用者が画面を操作する手順を，本論文では「操作仕様」と呼ぶ．これまでの操作仕様の記述の標準化は十分ではなかった．例えば，発注者ビューガイドライン[3]では，操作仕様の書き方として，操作に対する画面のリアクションを記述することを推奨している（図表 1）．一方で別の問題が発生する．この画面のリアクションは一般的に画面処理に関する仕様書（図表 2）にも記述される．これらは記述内容が類似しており，一部の記述内容は冗長といえる．特に冗長な記述は，大規模な開発では仕様変更の際，修正コストが掛かり，修正漏れによる品質低下のリスクが高まる．さらに，操作仕様は後工程のテスト仕様や操作マニュアル作成にも必要な情報になるにもかかわらず，再利用が難しいため，その都度属人的な対応をすることが多かった．

操作手順	
画面名	コース検索画面
(1)コースIDの入力 ・検索対象のコースIDを入力する. (2)該当するコース情報の検索 ・検索ボタンを押下する. ーコースIDを元に，コーステーブルからコース情報が検索される. ーコース詳細画面に検索結果が表示される.	

図表1. 操作手順の例

画面処理仕様			
画面名		コース検索画面	
ID	イベント名	入力	処理内容
1	検索ボタン押下	コースID	①入力チェック
		コーステーブル	②検索処理 コーステーブルから該当するコース情報を検索する.
			③画面遷移情報の設定 検索結果を画面遷移情報に設定する.
			④画面遷移 コース詳細画面に遷移する.

図表2. 画面処理仕様の例

3．解決策

我々は仕様フレームワーク[1]および前論文[2]の取り組みを発展させ，「テスト仕様」「業務仕様」「操作仕様」「画面仕様」「データ処理仕様」の 5 階層からなる仕様フレームワークを開発した（図表 3）．

各階層の役割と VDM++の記述方法の詳細は図表 3 に示した．本仕様フレームワークの特徴は，従来混在して記述されることが多かった，システム側の画面仕様と，人間側の操作仕様を明確に分けた点である．

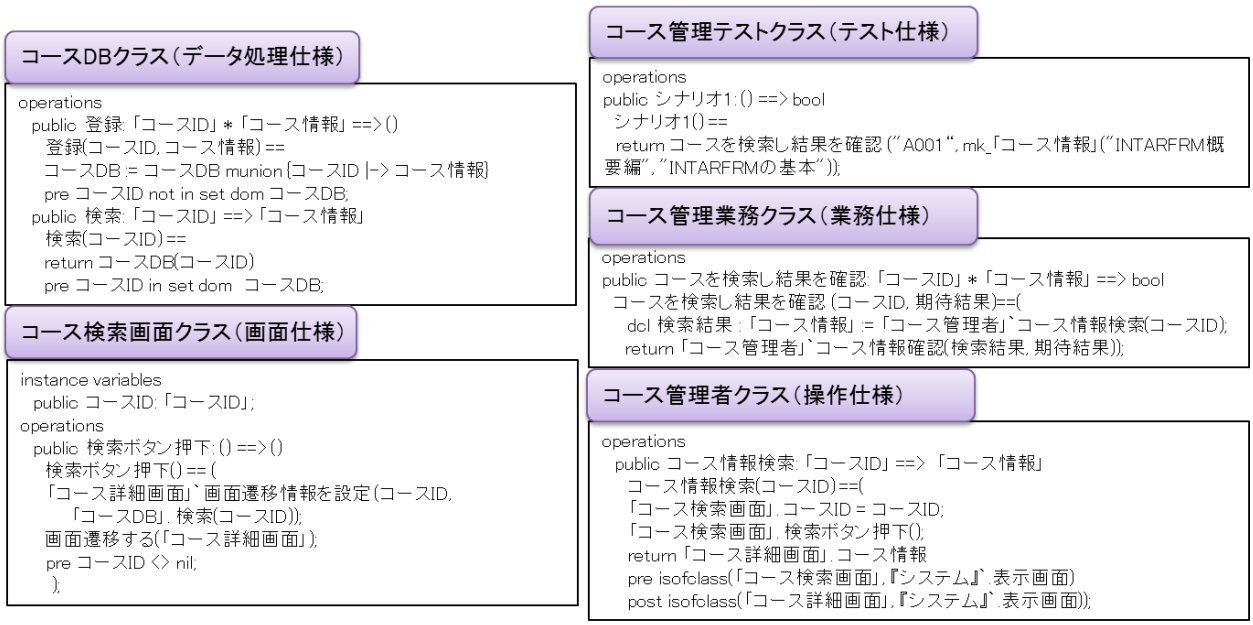
操作仕様は利用者像単位で作成する．例えば権限が異なる担当者と上司，あるいは営業部と人事部に分けることで，操作仕様を明確に区別できる．また，画面仕様との呼び出し関係が明

階層	役割	VDM++記述方法※
テスト仕様	テストケースを記述する	「操作」に具体的なテストデータを指定し，業務シナリオを呼び出す.
業務仕様	業務シナリオを定義する	「操作」に業務シナリオを記述する．引数に業務シナリオ全体の入力と出力を記述する．「操作」の内容に，関連する複数の操作仕様の呼び出しを記述する.
操作仕様	利用者像毎に画面の操作を記述する	「操作」に利用者の画面操作を記述する．引数に利用者の入力，戻り値に利用者が期待する結果を記述する．「操作」に記述できる対象は，画面仕様の画面項目と画面イベントに限る.
画面仕様	画面項目と画面イベントを記述する	「変数」に画面項目，「操作」に画面イベントを記述する．引数と戻り値は共に指定しない．「操作」の内容に，データ処理仕様の呼び出し処理等を記述する.
データ処理仕様	データ処理仕様を記述する	「操作」にデータの具体的な処理内容を記述する．（一般的な形式仕様記述に対応する）

※便宜上，instance variablesを「変数」，operationsを「操作」と記載

図表3. 仕様フレームワークの階層

Proposal of design method that describes the specification of enterprise business systems in the VDM++
† YOSHIMITSU Hamano, YOSUKE Suzuki, KEIKO Noda, FUJITSU LIMITED
‡ KAZUKI Munakata, FUJITSU LABORATORIES LIMITED



図表4. VDM++による仕様記述の例

確であると同時に階層が異なるので，仕様の境界線が明確である．さらに一連の操作仕様をつなげるのが業務仕様である．業務仕様の定義内容は前論文[4]に示したように，業務フロー図に変換することができる．この記述方法に従ったVDM++の記述例を図表 4 に示す.

４．効果

本仕様フレームワークにより，従来混在して記述することが多かった画面処理に関する仕様を，画面仕様と操作仕様に分けて明記することで，従来の仕様記述の冗長さが解消され，特に大規模開発では，仕様の保守性が高まり，品質向上とコストの削減に繋がる

さらに，VDM++の仕様記述は，Overture[5]などのツールにより，仕様実行が可能となる．

これまで，操作仕様を含めた画面の仕様に関する仕様の記述内容の正しさを確かめるためには，人手によるレビューが不可欠であった．また，レビューで見落とされた設計上の不具合は，テスト工程で発見されることが多く，仕様書の修正にまで手戻りが発生し，大きなコスト増の要因となっていた．

仕様実行が可能になることで，操作仕様を起点に画面から連携する各種仕様の不具合を，設計段階で検出することができ，大幅な品質向上及びコストの低減につながる．

５．今後の課題

実用化にはいくつかの課題が残る．VDM++は形式手法言語の中では分かり易いといわれるが，まだ開発者への認知度が低く，開発現場に普及するにはこれまでの設計手法から段階的に移行するための施策が必要である．また，利用者向けには VDM++の知識がなくても仕様の中身を確認するため，従来の設計書へ変換するツールの整備が必要と考えている．

さらに，VDM++の仕様記述から実際のプログラムやテストプログラムを自動生成することで，これまで属人的に行ってきた開発・テスト工程の大幅な効率化が期待される．今後，仕様フレームワークの品質を高めるとともに，自動生成に関する調査を開始する予定である．

６．おわりに

当仕様フレームワークにより，従来属人的に記述されてきた画面に関わる仕様の全体を VDM++で記述することができた．さらに，仕様実行により，これまで不可能だった設計段階でのテストにより仕様バグの抽出が可能となる．エンタプライズ系分野において，形式手法を本格的に活用する基盤が構築できたといえる．

参考文献

[1] 佐原伸：形式手法の技術講座, ソフト・リサーチ・センター, 2008

[2] 濱野義満, 銀林純：モデル駆動開発とテスト駆動開発を統合する仕様記述方法の提案, 情報処理学会全国大会, 2B-01, 2015

[3] IPA/SEC：発注者ビューガイドライン, http://www.ipa.go.jp/sec/softwareengineering/reports/20080710.html

[4] 濱野義満, 銀林純：業務フロー図に含まれる仕様を形式表現する手法の提案について, 情報処理学会関西支部大会研究報告, B-04, 2014

[5] Overture Tool：http://overturetool.org/