

仕様記述の役割を意識したフォーマルメソッド学習

西岡 貴史¹ 大森 洋一² 荒木 啓二郎²

概要: ソフトウェア開発環境の変化により, 仕様保守も重視されるようになってきている. こうした環境変化に対応するために, 実行可能であるフォーマルな仕様を用いて, 要求変更に対する対話的な仕様修正を行い, 実現可能性や要求充足を確認する過程を学習する教材を作成した. 学習プログラムでは教材の仕様記述の本質である発注者と開発者の対話を重視し, それぞれの役割を理解するためのロールプレイを行う. 大学の演習における試行の結果, 改善された部分とともにいくつかの課題が見つかった.

キーワード: フォーマルメソッド, VDM-SL, オンライン教材, 仕様記述学習

1. 仕様保守の重要性

Software as a service (SaaS) に代表されるクラウドコンピューティングの普及, 組込みシステムにおけるオンライン環境の一般化などにより, ソフトウェア開発者が開発したシステムを手放すことなく, 運用者と一体となってソフトウェアにより実現されたサービス提供を行う形態が増えている. 開発者の立場からみると, このような環境は開発プラットフォームを限定できる, ソフトウェア更新による新サービスを短期間で普及できるといった利点がある. その反面, 利用者の乗り換えコストが小さい, 他社の新サービスも短期間で普及する可能性があるなど市場環境は厳しくなる面もあり, 競争力を保つためには, ソフトウェアのライフサイクル全般に渡って機能追加や不具合修正といった保守作業が重要となる.

こうした環境変化につれてソフトウェア開発における仕様の位置づけも変わりつつある. 従来の仕様がソフトウェアライフサイクルの初期で確定し, 以後の工程では参照されるだけであったのに対し, ソフトウェアのライフサイクル全般に渡って保守される中間生成物のひとつとしての性格が強まっている. これは, 仕様が単に開発者によって参照されるドキュメントではなく, 運用時の不具合の判定や実装や修正作業見直しなど後工程で活用される割合が高まることを意味しており, 仕様記述についても要求との関係

だけでなく, 保守との関係も学習するのが望ましい.

2. 仕様記述の学習

仕様記述やモデリングを対象とした学習については多くの先行研究がなされている. 国内に限定しても, ノウハウの公開による意欲の改善 [2], 未参照変数の発見やグループ学習による識別子の命名学習による品質改善 [7], ユースケースのツールによる形式的検証 [4], インспекションを用いた UML モデルの欠陥学習 [1] などがある. 比較的長期間に渡る国際的なソフトウェア開発教育事例として DOSE がある [8]. DOSE は分散開発を行う教育プログラムであり, 仕様の不具合を防ぐために フォーマルメソッドの一部である契約による設計が有効であったことが報告されている [5].

これらの先行事例を踏まえて, 実行可能な仕様を記述可能なフォーマルメソッド記述言語である VDM-SL を 2 種類の演習教材を準備した. 本件における我々のポジションは以下である.

主張 ソフトウェア仕様を実行可能とすることによりライフサイクル全体での可用性が高まる

実行可能な仕様はツールによる短時間での検証や実行結果の実装との突き合わせなど活用範囲が広まる. ただし, 実行効率より保守性や可読性が重視され, 複数のリリースで維持される点がプログラムと異なる.

3. 演習事例

フォーマルな仕様記述の一種である VDM-SL の文法に関する学習教材を <http://www.aofa.csce.kyushu-u.ac.jp/aofa-case-text.php> に用意した.

¹ 九州大学 電気情報工学科
Department of Electrical Engineering and Computer Science, School of Engineering,
Kyushu University
² 九州大学 大学院システム情報科学研究所
Faculty of Information Science and Electrical Engineering,
Kyushu University

3.1 文法教材

VDM-SL 学習教材のリストを図 1 に示す。

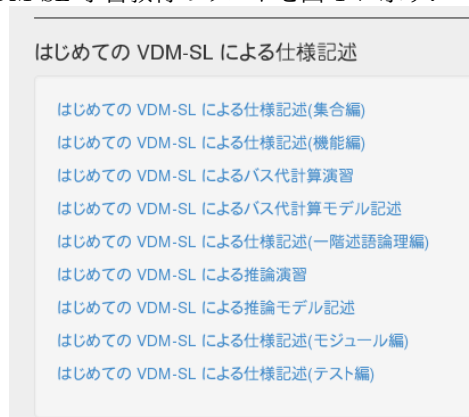


図 1 VDM-SL 学習教材リスト

Fig. 1 List of Learning Material for VDM-SL

これらの教材は VDM-SL のすべての文法要素をカバーするとともに、以下の工夫を行った。

- 集合、機能、写像などを小分けに記述している
企業内の講習などを想定し、ひとつの学習単位を短時間 (1 時間) になるように分割し、演習時間に応じて内容の取捨選択ができるようになっている。VDM-SL の熟練者が指導する前提ではあるが、ソフトウェア開発の専門家向けに全体で 8 時間程度、大学の学部生など入門者向けには 15 時間程度である。
- 個別の学習要素に対する演習と独立した演習を組み合わせている
VDM-SL の検証ツール OvertureTool[3] やオンライン IDE VDMPad[6] を活用し、実行可能な仕様を評価して確認しながら学習できるようにした。例題はこれらのツールでそのまま実行可能であるが、一部修正が必要な例題を用意している。

3.2 仕様変更教材

前節の演習により VDM-SL 文法は理解していることを前提として、日本語による記述と実行可能な VDM-SL 仕様からなる、ハンバーガーショップの状態変化を対象とした学習教材を用意した。VDM-SL による仕様そのものはコメントも含め 150 行、付随するテストは 80 行程度である。

実行可能な VDM-SL 仕様を用意したのは、演習時間の短縮という側面もあるが、現実のソフトウェア開発においてほとんどを占める実装後の仕様変更を模擬するためである。また、実行可能な仕様記述のスタイルという観点からも、関数型のように宣言的に記述することで統一し、指導する側の負担を減らしている。

学生に対する演習では、隣に座っている者同士で組になり、発注者役と開発者役に分かれて要求の変更 → 仕様の修正 → 修正の確認という手順で演習を進め、実装は実行可能な仕様により代用した。

記述結果の評価は仕様では難しい絶対的な品質評価の代わりに、参考文献 [2] 同様に仕様変更例を用意し、自分の記述と比較してもらうようにしている。

4. まとめと考察

- 対話的なロールプレイを行うための教材を用意した
仕様記述における発注者の役割を意識する効果はあった。しかし、対象の学生の習熟度が高くないこともあって十分な演習時間がとれておらず、実施効果の定量評価はまだできていない。
- 発注側の要求選択や仕様決定に関する教材も必要
思いついた要求変更に対し、優先候補を選択するのに予想以上に時間がかかった。設計上の決定を行うにも学習や技能習得が必要である。
- 自律的な学習には記述したモデルの評価基準が必要
フォーマルな仕様はツールを利用して無矛盾性を示すことは容易であるが、特に初学者にとって、仕様が要求に対して十分であることを確信するのは難しい。このとき、プログラムと異なり複数の観点からの評価基準を提示するのが望ましい。

謝辞 本研究の一部は JSPS 科研費 24220001, 基盤研究 (S) 「アーキテクチャ指向形式手法に基づく高品質ソフトウェア開発法の提案と実用化」の成果による。

参考文献

- [1] 金城龍弥, 植山淳雄: インспекション成果物を利用したモデリング学習支援システムの開発と評価 (学生セッション, 一般), 電子情報通信学会技術研究報告. KBSE, 知能ソフトウェア工学, Vol. 105, No. 652, pp. 47–52 (2006).
- [2] 小木曾禎, 遠山紗矢香, 湯浦克彦: 学生向けモデリング演習支援システムの開発と評価, 研究報告コンピュータと教育 (CE), Vol. 2011, No. 5, pp. 1–9 (2011).
- [3] Larsen, P. G., Battle, N., Ferreira, M., Fitzgerald, J., Lausdahl, K. and Verhoef, M.: The Overture Initiative – Integrating Tools for VDM, *SIGSOFT Software Engineering Notes*, Vol. 35, No. 1, pp. 1–6 (2010).
- [4] 松井駿介, 奥田博隆, 野呂 惇, 岡田康治, 加藤 真, 渡辺大貴, 松浦佐江子: モデリング能力育成を目的としたユースケース記述の自動評価手法, 電子情報通信学会論文誌. D, 情報・システム, Vol. 97, No. 3, pp. 461–472 (2014).
- [5] Nordio, M., Mitin, R., Meyer, B., Ghezzi, C., Nitto, E. D. and Tamburrelli, G.: The Role of Contracts in Distributed Development, *3rd International Conference of Proceedings of Software Engineering Approaches for Offshore and Outsourced Development*, pp. 117–129 (2009).
- [6] Oda, T., Araki, K. and Larsen, P. G.: VDMPad: a Lightweight IDE for Exploratory VDM-SL Specification, *FormliSE 2015* (Plat, N. and Gnesi, S., eds.), Florence, In connection with ICSE 2015 (2015).
- [7] 式見 遼, 若林智徳, 松浦佐江子: WebStudy: ソフトウェア開発技能育成を目的としたプログラムの実行・評価機構をもつ Web 教科書の開発, 電子情報通信学会論文誌. D, 情報・システム, Vol. 96, No. 10, pp. 2464–2475 (2013).
- [8] Zurich, E.: A distributed project for a globalized world, <http://se.inf.ethz.ch/research/dose/>.