形式手法を用いた仕様記述支援に関する研究

2003MT017 平松 由紀恵
指導教員　張 漢明

1　はじめに

システム開発では自然言語の要求からプログラム仕様の記述をすることが重要である．仕様に誤りがあると，システム構築後に欠陥が見つかり，大幅な手戻り作業が必要となる．手戻りを発生させる要因の一つとして，厳密な仕様作成がおこなわれていないことが挙げられる．
仕様記述に形式手法を用いることで，より厳密な仕様を記述することができ，手戻り防止に実用的であると考えられている．しかし，形式手法で条件式を記述することが難しく，どのような手順・考え方で記述して行くのかということが提案されていないので，普及していない．

本研究の目的は，洗練化された形式仕様を作成するためのガイドラインを提示することである．自然言語の要求から形式記述への変換手順，各条件の導出法を示す．本研究ではこの手順を分析し，仕様作成のガイドラインとして提示する．本研究では，仕様記述に不可欠な状態不変条件に着目した．適用事例として，南山大学の履修要項をもとにした授業登録システムの仕様を作成する．システムの状態をもとにして必要な操作から条件を考えて記述し，状態の整理をする．また，ガイドラインの妥当性について考察する．

2　形式手法の概要

形式手法とは，数学を基盤としたソフトウェア開発支援技術のことである．数学を基盤として記述すると一意的な記述になるので解釈の違いが生じにくくなる．形式的に記述するためにはシステムの操作を明確にする必要があるので，仕様記述が厳密になるという利点がある．

本研究では，仕様記述に VDM[1] を用いる．VDM は集合論に基づく仕様記述言語である．本研究ではデータ型を値の集合として定義し，操作を関数を用いて記述する．

3　仕様記述のガイドライン

顧客が自然言語で要求を出し，その要求をもとに形式手法を用いて仕様を記述する方法について考える．

仕様は，事前条件・事後条件，不変条件でシステムの条件式を記述して行く．仕様において最も重要なのは不変条件の記述である．事前条件・事後条件よりも不変条件で記述することで，各条件をシステム全体に反映することができるからである．そこで，不変条件を中心に記述するために以下のようなガイドラインを考えた．

要求から操作を関数を用いて記述し，必要な状態を抽出することから始める (**図 1**)．

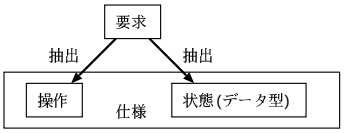

図 1　操作・データ型の抽出

状態不変条件の抽出 (**図 2**) では，操作から事前・事後条件を記述し，状態不変条件となっているものを抽出する．

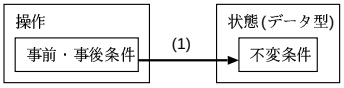

図 2　状態不変条件の抽出

状態不変条件の整理 (**図 3**) では，状態の見直しをおこない，記述の最小化をする．

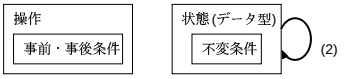

図 3　状態不変条件の整理

操作の事前条件・事後条件の整理 (**図 4**) では，見直された状態不変条件から，操作の各条件を記述し直す．

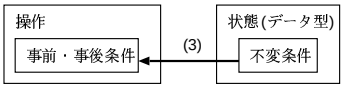

図 4　操作の事前条件・事後条件の整理

3.1　ガイドライン

操作の事前条件・事後条件を考え，その中からシステム全体の条件となっているものを状態不変条件として書き換える．状態不変条件を分析・整理し，整理された状態不変条件を用いて操作の事前条件・事後条件を記述する．

(1) 状態不変条件の抽出

要求から操作の関数と用いる状態 (データ型) を考え，記述する．操作は事前条件・事後条件や不変条件が行間に記述されていると考えることができ，各操作をおこなうために必要な条件を抽出することができる場合がある．抽出された条件はその操作に対する事前条件・事後条件というだけでなく，システム全体に対する条件となっている場合がある．

(2) **状態不変条件の整理**
仕様記述では，状態数をより少なくする必要がある．状態数を少なくすることによって，必要なものだけが抽出されている洗練化された仕様となるからである．
状態不変条件の中には，別の条件の組み合わせから導出できる条件が存在していることがあり，状態数を最小化していく際に省くことができる．

(3) **操作の事前条件・事後条件の整理**
記述された状態不変条件を用いて，各操作の事前・事後条件を記述する．不変条件を用いることで条件記述を 1 ヶ所にまとめることができ，条件の参照が容易になる．

4 授業登録システム

3 節で提示したガイドラインを用いて，南山大学履修要項から南山大学の授業登録システムの仕様を作成する．具体例として，登録操作について考える．

(1) **状態不変条件の抽出**
登録操作の条件の一部を以下に記述する．

- 登録する科目は，全て違う時間に開講される
- 登録する科目は，時間割にある科目である
- 登録する科目は，開講される科目である
- …

システム状態不変条件は，操作の事前条件・事後条件から考えることができる．事前条件・事後条件は操作の実行前後における仕様の満たすべき条件を記述しており，システム全体に対する条件になっている場合がある．
登録操作を VDM で記述したものを**図 5** に示す．

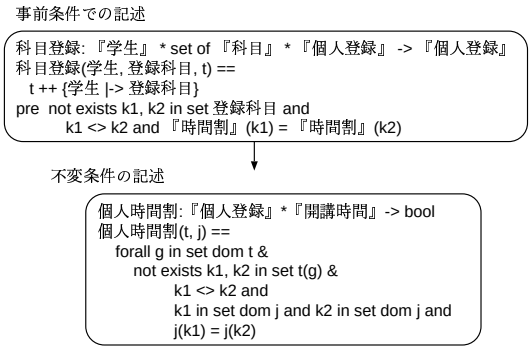

図 5　登録操作の VDM 記述

図 5 の上の pre 以下が事前条件の記述である．それを状態不変条件に変換したものが**図 5** の下の記述である．

(2) **状態不変条件の整理**
条件の組み合わせによって導出される条件の例として，以下の例を挙げる．

- 登録されている科目は，時間割にある科目である
- 時間割にある科目は，今学期の開講科目である
- 登録されている科目は，今学期の開講科目である

登録されている科目を A，時間割にある科目を B，今学期の開講科目を C とすると，上から A ⇒ B，B ⇒ C，A ⇒ C の関係になっている．(A ⇒ B ∧ B ⇒ C) ⇒ (A ⇒ C) が成り立つので，上の二つの条件から三つ目の条件が導出できる．よって，このような関係式が成り立つ場合には三つ目の記述は省略することができる．

(3) **操作の事前条件・事後条件の整理**
状態不変条件を用いて登録操作の事前条件・事後条件を記述したものを**図 6** に示す．‘登録する科目は，開講される科目である’ という事前条件は，(2) で導出できる条件として考えられたので記述する必要がない．

```
科目登録: 『学生』* set of 『科目』*『個人登録』->『個人登録』
科目登録(学生, 登録科目, t) ==
  t ++ {学生 |-> 登録科目}
 pre 個人時間割(t, 開講時間) &
     登録科目_時間割(t, 開講時間) &
     ...
 post 個人時間割(t, 開講時間) &
     ...
```

図 6　登録操作の事前条件・事後条件

5 考察

提案したガイドラインの妥当性について考察する．

ガイドラインの妥当性
授業登録システムを例とし，本研究で提案したガイドラインの妥当性の検証をした．自由に仕様を記述する際には条件が仕様の各所に散らばり，状態数や条件数が膨大になりがちである．しかし本研究で提案したガイドラインに従って記述することでシステムの条件を 1 ヶ所にまとめることができ，4 章の (2) での状態の最小化に役立った．手順が示されているので，4 章の (3) では条件記述を容易におこなうことができた．ガイドラインを用いることで要求の見直しをすることができ，より洗練化された仕様記述をおこなうことができると考えられる．

6 おわりに

自然言語の要求から形式化するための手順を分析し，仕様記述のガイドラインを提示した．授業登録システムを例として，条件記述の手順を示し，ガイドラインの妥当性について考察した．

謝辞
本研究を進めるにあたり，熱心にご指導くださった野呂昌満教授，張漢明助教授，蜂巣吉成講師，有益なアドバイスをくださった野呂研究室大学院生のみなさまに深く感謝いたします．

参考文献

[1] J. Fitzgerald，P. G. Larsen 著，荒木啓二郎，張漢明，荻野隆彦，佐原伸，染谷誠 訳：ソフトウェア開発のモデル化技法，岩波書店 (2003).