

仕様プランの分類と知識表現について

5N-2

関本 理佳 海尻 賢二

信州大学 工学部

E-mail: rika@cs.shinshu-u.ac.jp

1 はじめに

近年ソフトウェアのライフサイクルにおいて、保守の占める労力と時間の割合は増加の一途をたどり、ますます大きな問題となってきた。ソフトウェア保守において何よりも最初に必要なのは、ソフトウェアを十分に理解することで、この理解を支援することにより効果的な保守が可能になると考えられる。我々はこのような観点から、対象プログラムからそのソフトウェアが何をするのかという仕様レベルの情報を抽出することによりプログラム理解を支援する、リバースエンジニアリングツールの研究・開発を行なっている[?]

従来のリバースエンジニアリングツールのほとんどは、構造的観点からのアプローチが多く、ソースコードのレベルにとどまっていた。これをさらに高いレベルに引き上げ、意味的情報を抽出するためには、アプリケーションドメインの知識・理解が必要となる。また、認識結果を仕様としてドキュメント化するためには、仕様記述言語に関する知識も不可欠である。

そこで我々は、このような知識を獲得するために、アプリケーションドメインをGNUのテキストユーティリティに限定し、まず実際にその仕様を作成してみた。そして、この仕様プロセスの分析に基づき仕様に関する知識を仕様プランとして体系化した。本稿では、この仕様プランの分類と知識表現について報告する。なお、仕様記述には現実世界のモデル化のために適したZ言語[?]を用いた。

Classification of Specification Plans and
Their Knowledge Representation
Rika SEKIMOTO, Kenji KAIJIRI
Faculty of Engineering, Shinshu University
500 wakasato,nagano 380, Japan

2 仕様プランの分類と知識表現

図1に仕様プランの分類と構成を示す。

仕様プランとは、仕様記述において多用される標準的なパターンともいうべきものである。ある特定のドメインのみで利用される仕様プランを、ドメイン仕様プラン (domain specific specification plan) と呼ぶ。これを除く基本仕様プラン (basic specification plan)、原始仕様プラン (primitive specification plan)、関数仕様プラン (function specification plan) の3つは、複数のドメインで共通に利用できる汎用な仕様プランである。

各仕様プランの知識には、スキーマ定義の他に、以下の情報が付加されている。

Name:	プランの名前
Kind	プランの種類
Description	自然言語による内容記述
SupperSchema	抽象スキーマへのポインタ
Composition	本プランが利用している 下位の仕様プランへのポインタ
Parameters	総称パラメータの記述
Constrains	利用における制約の記述

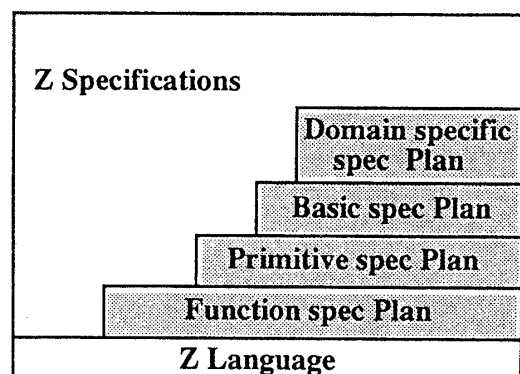


図 1: 仕様プランの分類と構成

2.1 ドメイン仕様プラン

今回対象としたテキストユーティリティは、ファイルの内容に対する種々の取り扱いについてのコマンドの集まりである。例えば、与えられたファイルの行数、単語数、文字数を表示する `wc` コマンドや、ファイルの最初の数行を表示する `head` コマンドなどがある。

ドメイン仕様プランには、ファイルの識別子からその中身を取り出す `FileStorePlan` や、バイト列を行単位に分割されたシーケンスにする `MakeLinesPlan` 等がある。

2.2 基本仕様プラン

仕様記述の基本となる基本仕様プランは、原始仕様プランと関数仕様プランの知識をベースにしてスキーマ知識を表現した仕様に関する知識である。

入力シーケンスに、ある一定の要素数ごとにあるパターンのシーケンスを挿入する `InsertPlan` や、入力されたシーケンス中のあるパターンを他のパターンで置換する `ReplacePlan` 等がある。`ReplacePlan` の知識表現の例を以下に示す。

Name: `ReplacePlan`

Kind	basic specification
Description	replace string1 by string2
SupperSchema	NIL
Composition	(DistributedConcatenate Separate Combine)
Parameters	X
Constrains	*UNDEFINED*

```

Replace
-----
input? , output!: seq X
string1? , string2?: seq X
-----
output! = ^/( (input? sep string1? )
               comb string2? )
  
```

2.3 原始仕様プラン

原始仕様プランは、単一シンボルを利用してのスキーマ知識を表現した、プリミティブな仕様プランである。

入力されたシーケンスの要素数を数える `CountPlan` や、あるシーケンスから指定された位置の要素を取り出

す `PickupElementPlan`、指定された区間の要素を取り出す `CopySequencePlan` 等がある。

2.4 関数仕様プラン

関数仕様プランは、一般的に利用できる関数を定義したもので、ツールキットとして利用される。

シーケンスをあるパターンが現れるごとにカットする `Cut` や、あるセグメントにわかれたシーケンスに対し、ほかのシーケンスをそのセグメントのターミネーターとして結合させる `Combine` 等がある。

3 おわりに

アプリケーションドメインを GNU のテキストユーティリティに限定し、その Z 言語による仕様記述を分析することにより、仕様に関する知識の抽出を行ない、仕様プランとして体系化した。本稿では、この仕様プランの分類と知識表現について述べた。

本稿で提案した仕様プランを、現在我々が開発中の”プラン認識に基づくリバースエンジニアリングツール”における仕様化のプロセスに利用することにより、理解の対象となるプログラムを、形式仕様として表現することが可能となる。また、本プランは Z 言語を利用して仕様を記述していくフォワードエンジニアリングにも利用可能である。但し、現在ではドメインをかなり限定しているため、他のドメインについての適用可能性の考察が不可欠である。

今後は、仕様プランの利用法についての分析を進め、本仕様プランのライブラリを利用して、実際にどの程度のコマンドに対し、どのような仕様記述が作成できるか検証していく予定である。また、プランライブラリにおけるプランの探索法についての考察も行なう。

参考文献

- [1] 関本理佳他: プログラム理解のための仕様記述の分析, 情報処理学会ソフトウェア工学研究会資料, 94-SE-98-4 (1994).
- [2] J. M. Spivey : *The Z Notation - A Reference Manual*, Prentice Hall (1989).