

問題向け設計言語開発ツールキットの開発と適用 \*

6 C-9

別府祥之 登内敏夫 中島震†

NEC C&Cメディア研究所‡

beppu@ccm.cl.nec.co.jp

1 はじめに

問題向け設計言語、Domain-Specific Language(DSL) [4]を中心とする AP 構築支援ツールは、特定問題領域のアプリケーションの開発生産性と品質を向上させる。しかし、AP 構築支援ツール自身の開発工数と保守拡張性という問題が生じる [2] [5]。この工数を削減するために問題向け設計言語作成ツールキット (DSL ツールキット) 「Rosetta」を開発し、NMS-AP 構築支援ツール FOG の開発に適用した。Rosetta とその効果について報告する。

2 FOG(NMS-AP 構築支援ツール)

FOG はネットワーク管理システム (NMS) を応用領域とし FScript と呼ぶ DSL を中心とした AP 構築支援ツールである [1] [2]。FOG は図 1 で示すように FScript による記述と NMS に特有な外部情報から、C++ のアプリケーションを自動生成する。FOG を用いると、NMS-AP の開発効率を向上するとともに、プラットフォームに依存しない AP の開発が可能になる [3]。

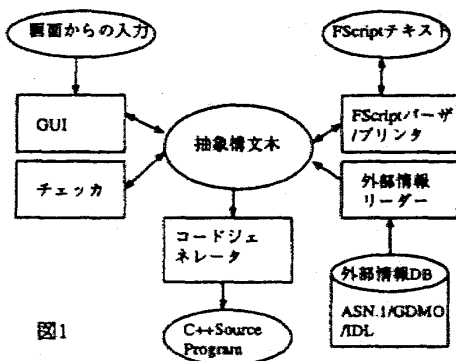


図1

3 Rosetta(DSL ツールキット)

DSL ツールキットとは DSL を中心とする AP 構築支援ツールの作成に有用なツール (DSL ツール) 群のこと

である。既存の DSL ツールの例としてはパーザ開発に有用な Lex/Yacc や JavaCC [7] がある。

3.1 Rosetta

Rosetta は、FOG に適用することを目的とする DSL ツール群である。ツールの可搬性を考えて Java で作成することにした。Rosetta を構成する各ツールに対して、適用部分、目的、機能について説明する。

(1.) 文法クラスチェッカ

適用部分: 言語設計フェーズ

目的: 具象構文の文法を定めた時点でトップダウンパーザが作れることを確認する。

機能: 文脈自由文法を入力とし、与えられた文法が LL(1) であるかどうか判定する。

(2.) 抽象構文木自動生成ツール (ASTG)

適用部分: 抽象構文木設計、プログラム開発フェーズ

目的: AP 構築支援ツールを作成する場合、抽象構文木を表すクラスが必要である。これらのクラスを自動生成する。

機能: 抽象構文木記述言語 ASDLJ で書かれた抽象構文木定義から抽象構文木を表す Java のクラスを自動生成する。生成された抽象構文木の内部状態をダンプするツールも自動生成する。なお ASDLJ は ASDL [6] の拡張となっている。

3.2 Rosetta を用いた AP 構築支援ツールの開発

Rosetta を用いた支援ツール開発手順を示す。

(1.) リファレンスモデル分析フェーズ

問題領域の設定 問題領域に特有な概念、操作を抽出する。

(2.) 言語設計フェーズ

DSL の具象構文の設計を行ない、文法クラスチェッカでチェックする。

\*A Domain-Specific Language Development Toolkit

†Yasuyuki Beppu, Toshio Tonouchi, Shin Nakajima

‡C&C Media Research Laboratories, NEC Corporation

### (3.) 抽象構文木設計フェーズ

コード生成に十分な情報を含む抽象構文木を ASDLJ により設計する。

### (4.) プログラム開発フェーズ

GUI、パーザ、チェッカ、ジェネレータを作成する。パーザのデバッグに ASTG が生成したダンププログラムを利用する。

## 3.3 Rosetta を適用の効果

AP 構築支援ツールの開発に Rosetta を適用すると次の利点がある。

#### 1. DSL の拡張、修正に簡単に対応できる。

AP 構築支援ツールの開発過程で、DSL に新しいデータ型や関数が必要になることがある [4]。そのたびに具象構文の文法と抽象構文木に追加、修正が必要になるが、Rosetta のツール群を用いることにより簡単に対応できる。

#### 2. 抽象構文木の最新の仕様をプロジェクト参加者全員が理解できる。

ASDLJ による抽象構文木の記述は非常にコンパクトでかつ理解しやすいものなので、プロジェクト参加者が常に最新の抽象構文木の仕様を理解できることにより (3.) (4.) のフェーズの分業がスムーズ行なる。

## 4 Rosetta の FOG への適用

Rosetta を FOG の開発に適用し以下の効果を得た。

### (2.) 言語設計 フェーズ

FScript の文法を文法クラスチェッカを用いてチェックすることにより、早期に FScript の文法を固定することができた。

### (3.) 抽象構文木設計フェーズ

ASTG により、FScript の抽象構文と外部情報テーブルを表すクラスを自動生成した。ASDLJ による 0.2KL の入力に対して、自動生成した Java のクラスは 8KL であった。

FScript の新しいデータ型の導入に伴い数回にわたり追加、修正が行なわれたが ASDLJ の簡単な修正で対応できた。

### (4.) プログラム開発フェーズ

FScript のパーザ (JavaCC [7] を利用)、外部情報リーダーのデバッグに ASTG が生成したダンプツールを利用した。チェッカ、コードジェネレータは、ASTG の生成クラスの API を用いて作成した。FOG の各モジュール作成担当者が抽象構文木の最新の仕様を理解していたので、FOG の統合テストがスムーズに行なえた。

## 5 終りに

今回は FOG に適用するという明確な目的のもとに Rosetta を作成し効果をあげた。今後も実際の DSL 開発に有用なツールを作っていくたい。

謝辞 DSL ツールキットの FOG-2 への適用に関して議論してくださる、C&C メディア研究所 友部主任、伝送共通技術本部 三木氏に感謝します。

## 参考文献

- [1] M.Miki, M.Tanaka, M. Tomobe, S.Nakajima : A Scripting Language for Network Management Applications and its Related Tool, *Proceeding of GLOB-COM'97*, pp.1714-1718, IEEE(1997)
- [2] 中島, 友部, 三木: 問題向け設計言語ベースの AP 構築支援ツール, オブジェクト指向最前線, 情処・O O'97 シンポジウム, 朝倉書店, pp.11-14(1997)
- [3] 友部, 中島, 三木: 問題向け設計言語によるアプリケーション構築支援環境の構築, 日本ソフトウェア科学会 第 14 回大会論文集, pp.585-588(1997)
- [4] Paul Hudak, *The Promise of Domain Specific Languages, Keynote Address of Conference on Domain-Specific Languages*(1997)
- [5] Arie van Deursen and Paul Klint: Little Language: Little Maintenance?, *Proceeding of DSL '97*, pp.109-127,(1997)
- [6] Daniel C.Wang, Andrew W.Appel, Jeff L.Korn, Christopher S.Serra: The Zephyr Abstract Syntax Description Language, *Proceedings of Conference on Domain-Specific Languages*, pp.213-227, USENIX(1997)
- [7] Sun Microsystems: JavaCC Documentation, <http://www.suntest.com/JavaCC/DOC>