

RGDL 语义结构的面向对象 模型及翻译方法*

陈 维 斌

(华侨大学计算机科学系, 泉州 362011)

摘要 分析报表描述语言 RGDL(Report Generator Description Language)与一般高级语言在语义上的明显差异后,给出一种新的语义描述方法和翻译方法(语义子程序设计). 重点介绍 RGDL 语义结构建模技术、面向对象的语义子程序设计以及目标代码生成系统的总体构成等.

关键词 报表描述语言, 语义结构的面向对象模型, 标准类库

分类号 TP 312

报表生成描述子语言 RGDL 文本^[1],其语法结构比较接近于自然语言,因而每一个语法范畴所对应的语义较之于高级语言的语义要复杂的多. 按 BNF 范式定义的高级语言文法对应的语义,通常只有计算、赋值、判断、转移之类的单一处理,是非结构的,一般被翻译成三元式组或四元式组. 尽管我们仍旧采用 BNF 范式来表示 RGDL 的文法(对于 RGDL 而言未必是一种好的方法),但其终结符的语义已无法分解成单一的处理;相反,是若干个单一处理组成的复杂处理.

例如,表项格式描述语句,其语法规则是<FORMAT>[<参数表>],终结符 FORMAT 的语义是一个基于活动字段格式描述表对指定表项进行格式处理的函数集. 不难看出,其语义与高级语言的语义有着明显的差异,很难用三元式、四元式等表示法来作为 RGDL 语义描述的中间代码形式. 与此相对应,翻译用的语义子程序,如仍旧沿用传统编译技术的语义子程序设计方法,同样难以胜任这种复杂语义的翻译.

因此,本文另辟蹊径,引入面向对象建模及程序设计技术来实现 RGDL 编译系统的语义翻译与目标代码的生成. 首先是静态的语义建模,即对所有语法范畴构造相应的语义结构模型. 其次当进行语法范畴的语义翻译时,激活对应的语义子程序,进行语义模型的标准类适配(模块的逻辑装配),并在中间代码表及系统参数表中各产生一个记录. 最后由目标代码生成系统进行主模块、公用函数模块、公用数据模块的物理装配及优化,生成 MSC/C++ + 7.0 程序.

* 本文 1995-04-27 收到;国家重点科技攻关计划基金资助项目

1 RGDL 语义结构的面向对象模型

1.1 语义结构模型的构造

RGDL 文本提供了报表定义与数据源描述、格式描述、数据操纵和报表管理等四大类计十六个基本语句。它们的语法结构都比较简单,可表示为:语句:〈动词〉〈参数表〉.SOURCE 语句是个例外,它包含 FROM 和 WHERE 两个子句,宜对其文法加以改造,即将 FROM 子句和 WHERE 子句拆分成独立的语句,然后进行拉链处理。可以看出,对一个特定的语句,动词和参数表这两个节点概念标识符(终结符)的语义,构成该语句的完整的语义,其语义结构模型如图 1 所示。

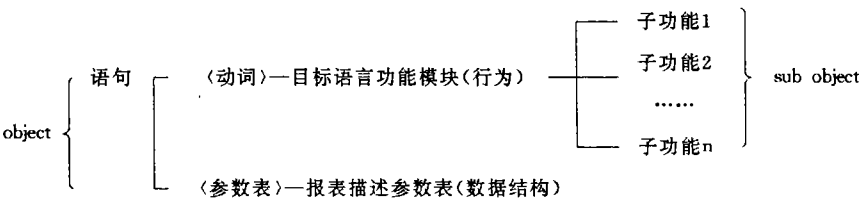


图1 RGDL的语义结构模型

图 1 是一个具有面向对象特征的语义结构模型,动词对应的功能模块(包含一组成员函数)是对象的行为,参数表所对应的各类报表描述子表是对象的数据结构。在实现上,用 CLASS 来封装对象, n 个子功能(若存在的话)可以视为语句这一基类的导出类(子类),它即继承基类的操作和数据,又可适当增加新的行为和数据。

通常,机器翻译系统的 目标语言的语义结构是动态构造的^[2]。由于 RGDL 的语义关系简单,可以将图 1 所示的面向对象模型直接作用于 RGDL 的每一语句,静态地构造其目标语言的语义结构模型。建模有两个目的,其一可用面向对象程序设计技术将模型转化成目标语言的对象类,包括基类、子类、和派生函数,藉此建立标准类库(软件 IC 库);其二为设计语义子程序提供设计标准和方法。

在语义翻译阶段,由语义子程序实现类的中间形式;在目标代码生成阶段,由生成系统抽取标准类库中的实在类替代中间形式,装配成 MSC/C++7.0 语言程序。

1.2 实例

下面给出制表语句 DRAW 的语义结构模型以及对应的类说明。

(1) DRAW 的语义结构模型(图 2):

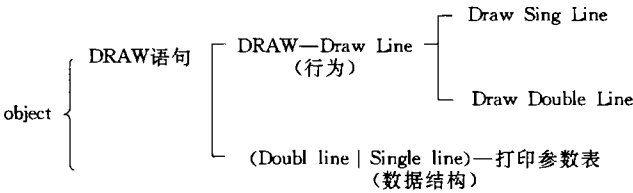


图2 DRAW语句的语义结构模型

(2) Draw 语句的类说明:

```
class Draw{
    Struct print _ table;
    .....
public
    Drawline(){...};
    .....
}
class Double Line:public Draw {
    protected
        .....
    public
        Draw Single Line();
        .....
}
class Single Line:public Draw {
    protected
        .....
    public
        Draw Double Line();
        .....
}
```

2 语义子程序设计

2.1 语义子程序的功能

传统编译技术的语义子程序是基于产生式的. RGDL 编译系统的语义子程序是基于语义结构 OO 模型的, 主要任务是进行模型的“标准类”适配和全局数据标识, 也即从标准类库中选择对应的类及子类, 将主要信息登记到中间代码表内(表 1); 若有全局数据(没有封装在类中的数据), 还需要在系统参数表中进行登记(表 2). 中间代码表和系统参数表构成语义翻译阶段的产品——RGDL 程序的中间代码, 提交给生成系统进行物理装配. 表 1 与表 2 的 n 个记录, 分别对应源程序的 n 个语句(包括被拆分的子句); 表 1 的子类表、公用函数表和调用参数表这三个项目的值; 均是指向相应子表的指针; 表 2 用“ \times ”和“ \vee ”记号, 分别标识对应子表的使用与否.

表 1 中间代码表

| 语句 ID | 基类名 | 子类表 | 公用函数表 | 调用参数表 |
|-------|----------|-------|-------|-------|
| DRAW | Drawline | * prt | * prt | * prt |
| : | : | : | : | : |

表 2 系统参数表

| 语句 ID | 状态参数表 | 活动字段格式参数表 | 数据加工参数表 | 打印参数表 |
|-------|--------|-----------|----------|--------|
| DRAW | \vee | \vee | \times | \vee |
| : | : | : | : | : |

2.2 语义子程序模型设计

由上面分析可知, 每一个语义子程序应具有“标准类”适配、系统参数标识、标准类库管理(搜索)和表格管理等四项基本功能, 一个明显的特征是, 不同的语句所对应的语义子程序之差异仅在于适配规则和标识规则的不同. 例如, DRAW 语句的适配规则描述为: 在标准类库中搜索名为 Drawline 的类及所有子类, 将其信息记入中间代码内; SORT 语句(排序)的适配规则描述为: 在标准类库中搜索名为 Sort 的类及所有子类, 将其信息记入中间代码内. 因此, 我们设计了如图 3 所示的语义子程序模型. 该模型将 RDGL 的语义这一总体概念映射成一个基类, 各语句的语义子程序的差异部分, 被映射成标准类适配及系统参数标识所对应的 n 个子类; 公共功能部分, 则通过继承基类的“表格管理”、“标准类库管理”这二个成员函数的操作

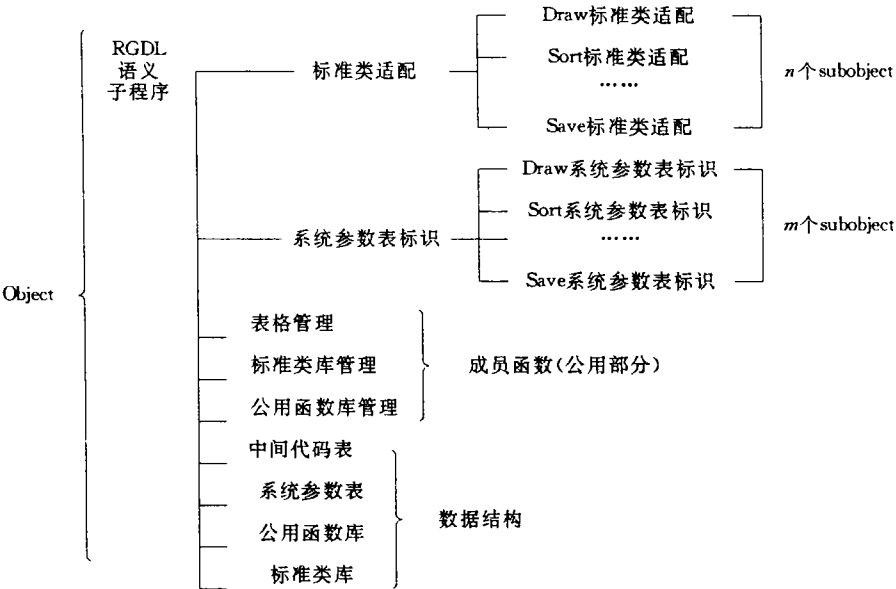


图3 RGDL的语义子程序模型

来实现。

3 目标代码生成系统

RGDL 编译系统的目标代码生成系统,本质上是一个程序自动生成器。它用来完成由类说明集和系统参数标识构成的中间代码,向 MSC/C++7.0 目标程序的转换,其构成如图 4 所示。

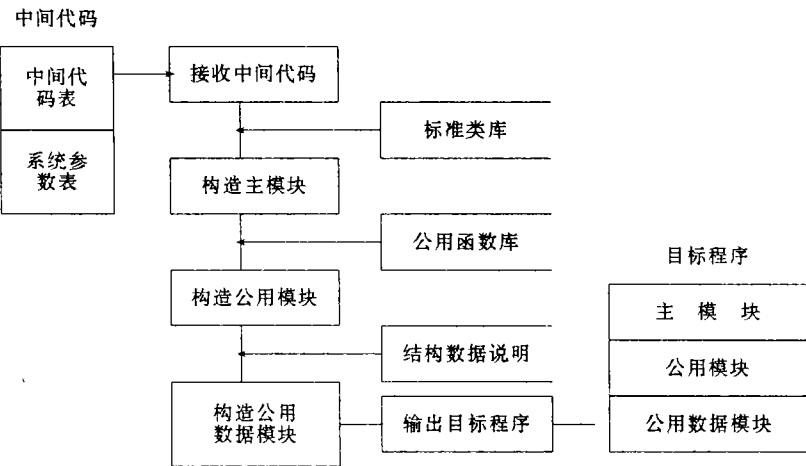


图4 生成系统构成图

在图 4 中的构造主模块，是根据登记在中间代码表中的类名(包括子类名)，从标准类库中提取相应的类及子类说明,按顺序写入目标程序中,构成主模块,并在主模块中生成一个 main 函数,进行类定义以及形成类的行为调用的控制流程。构造公用模块,是根据登记在中

间代码表中的公用函数名,从公用函数库中提取相应的函数子模块,进行全局优化后,构成公用模块.而构造公用数据模块:则是根据系统参数表所标识的各个数据名,从结构数据说明文件中提取相应的结构类型说明,据此定义全局数据,进行全局优化后,构成公用数据模块.

参 考 文 献

- 1 陈维斌,陈启泉,吴清江.报表生成描述子语言 RGDL 文本.华侨大学学报(自然科学版),1994,15(3): 34~347
- 2 卞世力,姚天顺,金 鸿.一个从中间语言生成目标语言的原理和方法.软件学报,1994,5(9):1~7
- 3 陈火旺,钱家骅,孙永强.编译原理.北京:国防工业出版社,1984.132~172
- 4 Atkinson L, Atkinson M, Mitchell E. Microsoft C/C++ 7.0 使用指南.曹晓峰等译.北京:清华大学出版社,1993.328~363

Object-Oriented Model and Translation Method of the Semantic Structure of Report Description Language

Chen Weibin

(Dept. of Computer Science, Huaqiao Univ., 362011, Quanzhou)

Abstract After analysing semantic difference between report description (RD) language and common high-level languages, a new method of semantic description and translation is given. Emphasis is placed upon the modelling technique of RD language; the design of object-oriented semantic subroutine; and also the global constituent of object code generating system.

Keywords report description language, semantic structure, object-oriented model