

# OOAnalysis 的设计和实现\*

倪彬 冯玉琳 黄涛

(中国科学院软件研究所计算机科学开放研究实验室 北京 100080)

**摘要** SCOP 是一个面向对象的组合框架软件,OOAnalysis 是 SCOP 中支持系统分析的一个工具系统,本文介绍了 OOAnalysis 的设计和实现. 该工具采用并行/递归式面向对象的分析方法,采用自底向上的系统构造和对象抽象与自顶向下的系统分解和对象精化相结合的方法建立对象模型,强调多层次抽象、并行工作和递归建模. 这个工具有多层次、可视化和自动化的特点.  
**关键词** 面向对象的系统分析,对象构造,对象精化,多级抽象,并行/递归式开发,自动化.

面向对象的技术是当今软件工程发展的主流技术<sup>[1]</sup>,SCOP 组合框架软件是在这一方向的良好探索. 它遵循面向对象的分析、设计和实现的多层次抽象的并行/递归式开发方法. OOAnalysis 用于 SCOP 系统中面向对象的系统分析这一关键步骤.

OOAnalysis 的主要目的是完成对问题空间的分析和系统建模,它的任务是描述系统中的对象,对象的逻辑结构和动态特性,对象间的分类关系、构造关系、映射关系和通信关系,从而建立系统的静态结构和动态模型. OOAnalysis 是支持并行/递归式开发方法的系统分析,与传统的 OOA(object oriented analysis)方法相比,它具有以下几个新的特点:

(1)既支持自底向上的由对象构件构造系统的构造性分析,又支持自顶向下的对象精化的分解性分析,而且把二者有机地结合起来.

(2)支持各级分析员在不同层次协同并行工作,递归求精,从而支持并行/递归式开发.

OOAnalysis 提供了一种简便、快速、灵活的工具来支持系统模型分析. 这种工具最主要的特性包括:多层次(可从不同的抽象层次建立和修改模型)、可视化(提供良好的可视界面,使分析员能清晰简便地完成分析)和自动化(提供自动化的工具以减轻分析员的负担).

使用 OOAnalysis 可以使我们在概括客观事物的最自然的方法框架上定义和交流系统需求,可以使我们在不同的抽象层次上理解问题空间,良好地进行数据封装和隐藏,加强系统的内聚性,减少系统的耦合性. OOAnalysis 得到的模型是一个稳定的模型,在以后的设计和实现中没有质的变化,只有补充和完善. 这种模型的稳定性,为组合软件工程的并行/递归式开发提供了基本的支持.

\* 本文研究得到国家自然科学基金和国家 863 高科技项目基金资助. 作者倪彬,1969 年生,博士生,主要研究领域为面向对象的软件工程技术. 冯玉琳,1942 年生,研究员,博士生导师,主要研究领域为软件工程,程序设计方法学,面向对象的理论和技术. 黄涛,1965 年生,博士,副研究员,主要研究领域为软件工程,程序设计方法学,对象语义理论.

本文通讯联系人:黄涛,北京 100080,中国科学院软件研究所计算机科学开放研究实验室

本文 1996-07-18 收到修改稿

# 1 OOAnalysis 分析方法

## 1.1 分析的原则

在 OOAnalysis 中主要遵循以下几个原则:

(1)构造和分解结合的原则:构造是指由基本对象组装成复杂对象或活动对象的过程;分解是指对大粒度对象进行精化从而完成对系统模型的细化的过程,这两者的结合是组合软件工程中并行/递归式开发的基础.

(2)抽象化和具体化结合的原则:抽象化强调实体的本质、内在属性,而忽视一些无关紧要的属性.在系统开发中,抽象指的是在决定如何实现对象之前,定义对象的意义和行为.使用抽象可以尽可能避免过早考虑细节.[2]具体化是指在精化的过程中,对对象的必要细节加以刻画,有助于确定对象,加强系统模型的稳定性.

(3)封装的原则:封装指的是将对象的各种独立的外部特性与内部实现细节分开,封装防止了由于程序的相互依赖性而带来的变动影响.[2]

(4)继承的原则:继承指的是能直接获得已有的性质和特征而不必重复定义它们.继承可以减少冗余度,增加系统的可复用性.

(5)相关的原则:相关包括静态结构的关联,如对象间的映射关系等,也包括动态特性和关联,如通信关系等.这是确定对象和对象关系的重要手段.

## 1.2 分析的步骤和方法

OOAnalysis 采用并行/递归的分析方法.分析在不同的抽象层次递归进行.每一次递归是对高级抽象层次的完善和精化.递归分析过程可以按照高级抽象层次的不同方面,由多个分析员并行完成.

OOAnalysis 分析的一次递归过程如图 1 所示.

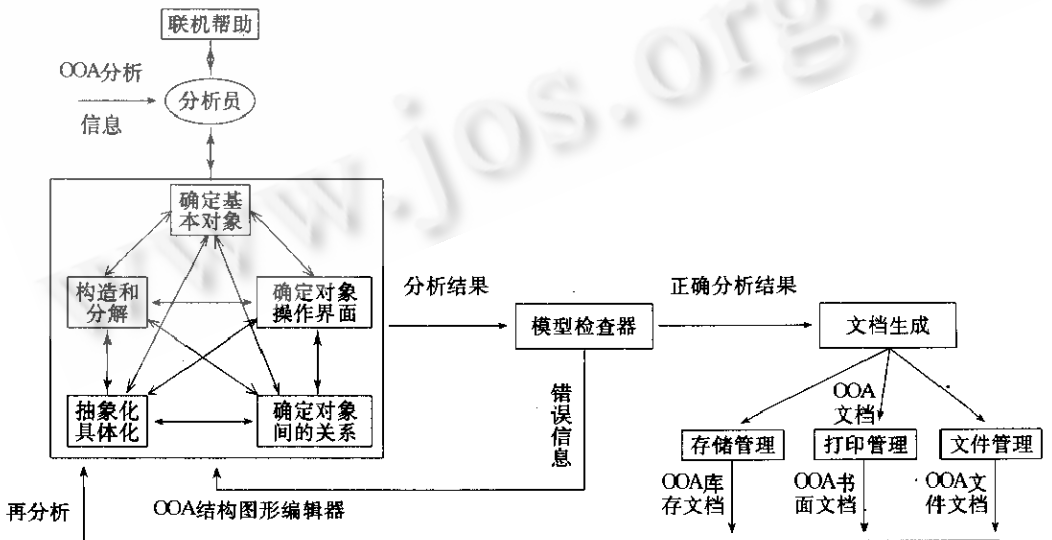


图1 OOAnalysis分析方法图

### (1)确定当前层次的基本对象

一个对象是一个封装和一个抽象:封装是对属性以及这些属性上专有操作的封装;抽象

是对问题空间的抽象,在各种 OOA 方法中,最关键的问题是如何确定对象.在 OOAnalysis 中,我们采用逐步扩充的方法递归地在各个层次确定各自的对象,从而建立不同抽象层次的对象视图.我们有 2 种手段确定对象,①从问题空间、文字资料和图片资料入手.②从上一层分析的结果出发,对上一层分析中得到的对象进行这一层次中必要的精化.这种精化可以是上一层对象私有属性和操作的确定,也可以是上层对象具体实现中需要的新对象的引入.

### (2) 用构造和分解相结合的方法扩充对象模型,确定对象间的聚合关系

从基本对象出发,我们根据构造和分解相结合的原则,把可以从基本对象构造出来的符合当前层次需要的对象自底向上地构造出来;把当前层次需要分解的对象进行分解.这种构造和分解关系有 2 个方面的含义:①描述对象间的静态组装结构,即整体和部分的关系;②描述活动对象和子活动对象的动态组装结构.我们用聚合关系来描述对象间的这种构造和分解关系.

### (3) 用抽象化和具体化相结合的方法扩充对象模型,确定对象间的子类关系

我们用抽象化的方法,抽取一组相关对象的共性,构造抽象对象作为这一组对象的抽象父类,这一组对象作为这一抽象类的子类.同时,我们也应用具体化的方法,把一个抽象对象分类成一组具体的子类对象.通常子类继承父类的所有属性和操作.它们同时还有自己的属性和操作.在我们的分析方法中允许多重继承.但是,我们不允许循环继承.子类关系描述了对象间的这种抽象和具体的关系.

### (4) 确定对象间的实例连接关系

实例连接关系描述一个对象实例与另一个对象实例的必要的映射关系.在确定实例连接关系的时候,只要恰当就尽量在较高的抽象层连接对象.把连接的多重性和参与性结合起来考虑,存在 4 种对应关系:1 对 1、1 对多、0 对 1、0 对多.在 OOAnalysis 中,我们只允许二元的实例连接.多元的实例连接可以通过传递性来表示,也可以引入一个新的对象来表示.对于多对多的实例连接,如果存在连接自身的属性,那么我们引入一个新的对象来表达这种连接.我们允许对象到自身的实例连接,但不允许 2 个对象之间有多于一个实例连接.

### (5) 确定对象间的消息连接关系

消息连接关系描述了对象间的通信关系.这种消息传递触发了对象的操作,从而推动系统运行.消息连接可以是单向的,也可以是双向的.对象收到消息后,根据内部状态、事件、响应表决定要进行的操作.

### (6) 确定当前层次对象的操作界面

#### • 确定对象的属性

分析过程中,要把每个属性附加到问题空间中与该属性所最直接描述的客观事物相对应的对象中.属性应该是原子层次上标志的.属性要区分是公有属性还是私有属性.同时,要对其取值范围、限制、计量单位和精度等作出非形式化的描述.在当前层次我们只需要确定对象的公有属性,其私有属性及内部细节可以留待下一层分析完成.

#### • 确定对象的操作

操作是对象接到消息后要进行的处理.对象通常有 3 类操作:自身维护操作;计算操作和监控操作.操作要区分是公有操作还是私有操作.分析过程中,要对操作的参数、返回值等作出非形式化描述.在当前层次我们只需要确定对象的公有操作,其私有操作及内部细节可

以留待下一层分析完成.

- 确定对象的动态行为特性

对象的动态行为特性表现为对象的生命周期. 在 OOAnalysis 中, 我们用内部状态、事件、响应表来刻画. 内部状态、事件、响应表的每个表项描述了对象在一个特定的状态, 接收到某个消息后要作出的反应. 内部状态、事件、响应表实际上就是一个对象活动的状态转换图, 这个图表达了对象的动态行为约束.

(7) 检查分析模型的一致性

在模型建立以后, 常常需要对模型进行一致性检查, 以便及早得到不一致性和不必要的复杂性等问题的警告. 发现这些问题后, 要返回到相应的步骤进行再分析和修改, 消除这些问题.

(8) 生成正文文档

在模型建立后, 我们已经得到了分析模型的图文档. 我们还需要生成包含完整的模型信息的正文文档, 提供给 SCOP 的其它工具使用.

### 1.3 分析的结果

OOAnalysis 的分析结果包含 2 个部分: ①OOA 分析图, 该图存储在对象库中, 可以复用和再分析. ②OOA 分析的正文文档, 它由 OOA 分析图产生, 为其它工具使用.

## 2 OOAnalysis 工具设计

### 2.1 设计思想

(1) 充分支持 OOAnalysis 并行/递归的分析方法

工具是为方法论服务的. OOAnalysis 工具要支持分析员在不同的抽象层次上并行地进行工作. 同时, 也要保证这些工作能够通过递归有机地结合起来.

(2) 支持友好、简便和灵活的操作界面

OOAnalysis 工具是一个人机交互工具, 友好的图形界面是工具的一个重要环节. 我们充分利用 Motif 的 API(application program interface), 实现一个友好的 GUI(graphic user interface), 提供给分析员一个好的可视化界面, 增加分析的清晰度和简便性.

(3) 尽量减轻分析员的负担

自动化的手段可以大大减轻分析员的负担, 体现出工具的优越性. 多样化的输出模式, 可以使工具适用于各种应用场合.

### 2.2 设计规范

- OOA 分析图文档规范

OOAnalysis 的主要过程是建立结构化 OOA 的分析图. 这个分析图由图页组成. 图页是图中相对独立的部分抽出来形成的子图. 在一个页中, 其它页通过子图图标来标识, 页面的关系通过依赖关系来表示. 子图图标可以随意的 ZOOM 和跳转. 每一个图页由图元组成, 图元的规范如图 2 所示.

- 正文文档规范

正文文档给出了分析结果的所有细节. 这样的文档在软件的工程开发过程中是必不可少的. 它还为其它环境中对分析结果的复用提供了接口. OOAnalysis 的正文文档是由其图

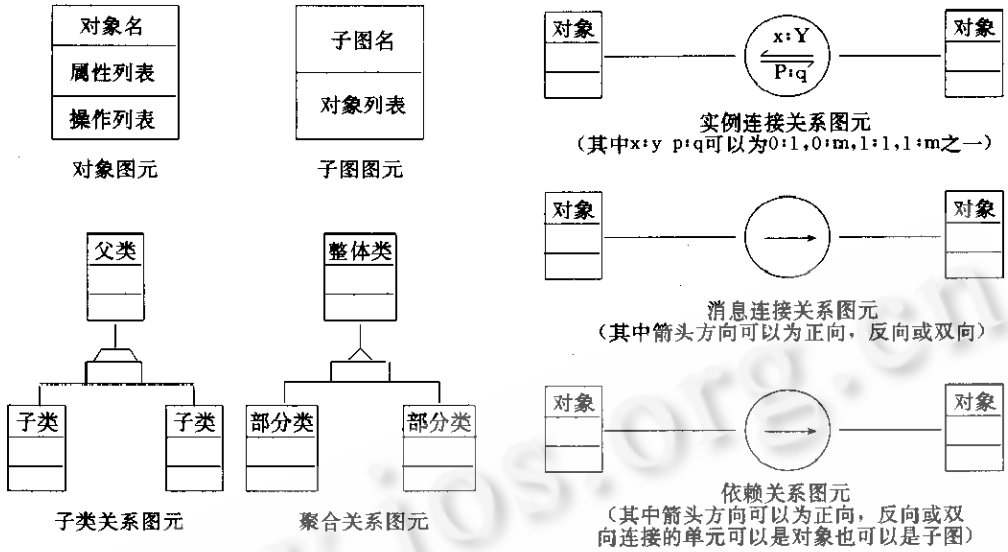


图2 OOA分析图图元规范

文档自动生成的,它是一种结构化的自然语言表述,主要规范如下:

```

<DOCUMENT> ::= OOA TEXT DOCUMENT <CLASSSPECIFICATION>+
              (SUBGRAPHSPECIFICATION)* ENDOOA TEXT DOCUMENT
<CLASSSPECIFICATION> ::= CLASSSPEC <CLASSNAME>
              <SUBGRAPH CLAUSE> <DESCRIPTION CLAUSE>
              <ATTRIBUTE CLAUSE> <COMPONENT CLAUSE>
              <OPERATION CLAUSE> <SUPERCLASS CLAUSE>
              <INSTANCE CONNECTION CLAUSE>
              <MESSAGE CONNECTION CLAUSE> <DEPENDENCY CLAUSE>
              <STATE EVENT RESPOND CLAUSE>
              ENDCASSPEC
<SUBGRAPHSPECIFICATION> ::= SUBGRAPHSPEC; <SUBGRAPH NAME>
              <SUBCOMP CLAUSE> <DEPENDENCY CLAUSE>
              ENDSUBGRAPHSPEC

```

### 2.3 系统的对象模型

图3给出了一个概要的OOAnalysis工具对象模型图。

### 2.4 关键技术和算法

#### 2.4.1 结构化图形编辑器

- 多层次、多侧面的可视技术

在结构化的图形编辑器中,我们利用多层次对话框来隐藏不同层次的信息;采用不同的颜色来抽象不同的关系;采用图标 ZOOM 及自动跳转来反映不同的抽象层次的递归及连接关系,从而多层次、多侧面地达到可视化效果. 支持递归/并行的分析方法. 在 OOA 分析图中,每个图元都对应一组多层次对话框,每个图元都可以 ZOOM 到低级抽象层次,每个层次都可以 ZOOM 回高级抽象层次.

- 自动布线技术

在结构化的图形编辑器中,我们采用自动布线技术来完成图元之间的关系连线,该算法自动绕过连线之间的图元障碍,连接相应的图元. 该算法是线探索法<sup>[3]</sup>的改进算法,其主要思想如下:设 A, B 为图上 2 点,图中有不可穿越的垂直线集合 C<sub>v</sub> 和不可穿越的水平线集合

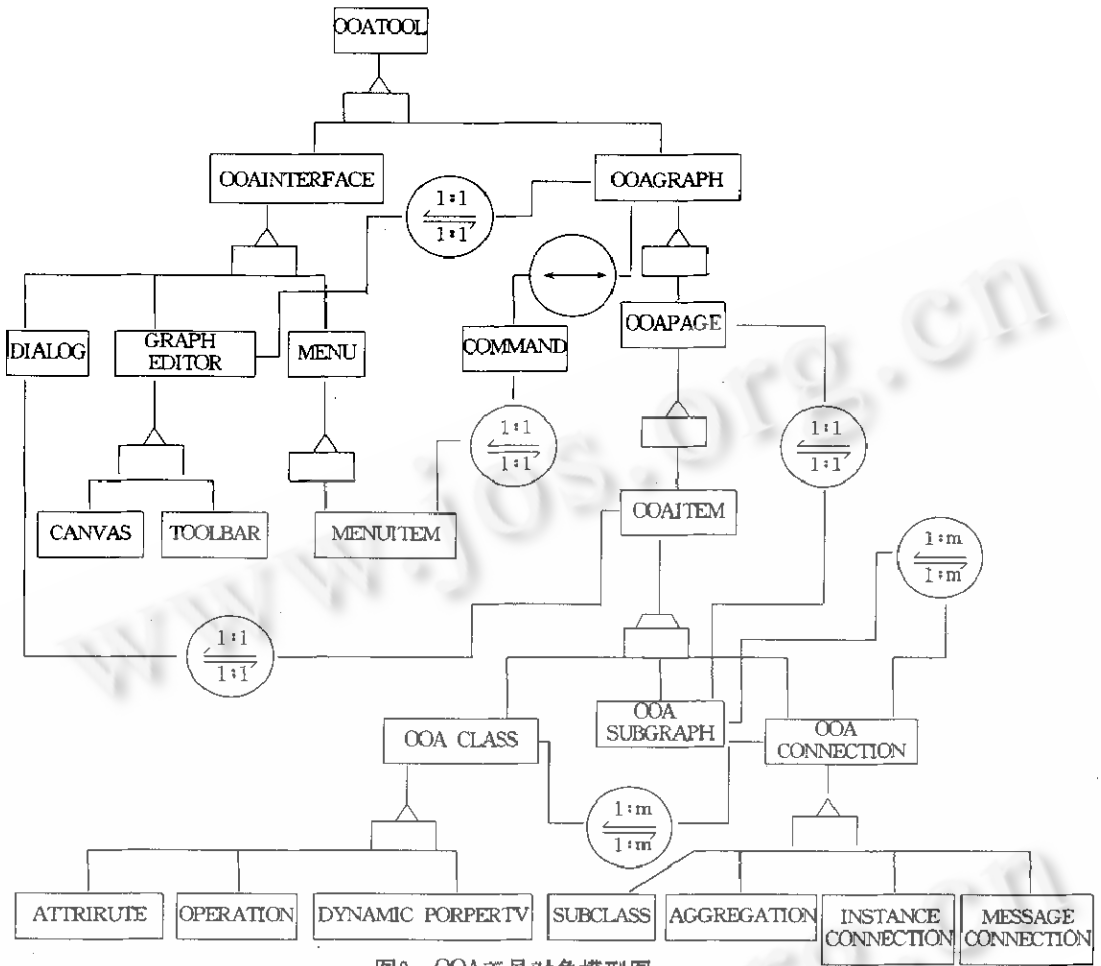


图3 OOA工具对象模型图

$C_H$ . 问题是用垂直水平线连接  $A, B$ , 连线绕过障碍线集合.

为了算法描述的方便, 我们首先给出一些简单的定义.

定义 1. 如果由点  $P$  作一条垂直线, 与之相交的  $C_H$  的元素称为  $P$  的水平障碍. 由  $P$  作一条水平线, 与之相交的  $C_V$  的元素称之为  $P$  的垂直障碍.

定义 2. 通过  $P$  的垂直线段(终止于  $P$  的水平障碍)称为  $P$  的垂直逸出线. 同理可以定义  $P$  的水平逸出线.

定义 3.  $P$  的垂直(水平)逸出线上一点  $E$ , 若满足如下条件:

- (1) 一条垂直障碍已不是  $E$  的垂直障碍, 且
- (2) 该垂直障碍之间的任何垂直线段都不是  $E$  的垂直障碍.

则称  $E$  为该逸出线上的一个逸出点.

定义 4. 由  $A$  衍生出的逸出线及其逸出线上的逸出点进一步衍生出的逸出线集合定义为  $\{L_{AH}, L_{AV}\}$ , 其中  $L_{AH}$  为水平逸出线集合,  $L_{AV}$  为垂直逸出线集合. 同理可以定义  $\{L_{BH}, L_{BV}\}$ .

算法思想: 首先从  $A, B$  引出一条水平逸出线和一条垂直逸出线, 构成最初的  $\{L_{AH}, L_{AV}\}$  和  $\{L_{BH}, L_{BV}\}$ . 若它们中由一对元素相交, 交点为  $(x, y)$ , 则自然立即找到一条连接路径  $(A,$

$(x, y), B)$ . 否则, 在  $\{L_{AH}, L_{AV}\}$  和  $\{L_{BH}, L_{BV}\}$  上分别求逸出点  $\{E_A\}, \{E_B\}$ , 然后, 作出这些逸出点的新的逸出线, 并加入到  $\{L_{AH}, L_{AV}\}$  和  $\{L_{BH}, L_{BV}\}$ . 当出现一对元素相交时, 就找到了连接路径. 如果没有任何一对元素相交, 则可重复上述过程直到得到一对相交的逸出线或  $\{L_{AH}, L_{AV}\}$  和  $\{L_{BH}, L_{BV}\}$  都没有新的元素产生(布线失败)为止. 连接路径可用回溯法找出. 为了提高搜索效率, 我们给出一些优先搜索的条件, 如优先的方向、距离等.

#### • 图合并技术

为了使并行工作分析员的分析结果最终能够合并成为一个整体. 我们提供了图的合并技术支持. 高级分析员可以通过拖放将 2 个分析图按 ZOOM 合并或重叠合并的方式合二为一.

#### • 系统的灵活性技术

在图形编辑器中, 我们采用了拖放技术<sup>[4]</sup>、快速工具按钮技术、拷贝、剪裁和张贴技术以及 UNDO 技术. 实时配置技术的使用更增加了系统的灵活性.

### 2.4.2 自动化工具

作为一个计算机辅助分析工具, 我们还提供了一些自动化工具.

(1) 自动模型检查工具主要用来完成模型的一致性检查.

对 OOA 分析图的模型检查主要包括如下检查(其中一部分是警告性的检查): 类、属性、操作的名称唯一性; 类的非孤立性; 类的足够复杂性; 抽象层次之间的一致性; 继承的一致性和对象相关的一致性.

(2) 自动文档生成算法主要用来完成 OOA 图到正文文档的自动生成, 生成的结果可以浏览和作必要的修改.

### 2.4.3 多种输出模式

为了满足 OOAnalysis 工具的不同应用要求, 我们使分析的结果有多种输出模式. 首先, 分析的结果可以作为对象存入对象库中. 分析图在对象库中作为 OOA 分析图类的对象存放, 整个图按照系统模型中定义的对象结构存储在对象库中. 正文文档作为文档类的子类 OOA 分析文档类的对象存储在对象库中. 这些对象和类可以通过其它工具浏览和查询. 对象库中的分析图和正文文档主要是以进一步分析和复用分析为目的的. 其次, 分析的结果也可以作为文件存放. 文件存储的目的是为了在其它环境和方法中能够复用分析结果, 增强对外接口的灵活性. 最后, 用户也可以打印分析图和正文文档. 我们采用了 POSTSCRIPT 技术, 使硬拷贝不依赖于特定的打印设备.

## 3 总 结

从 80 年代起, 人们提出了许多面向对象的分析方法. 其中代表性的有 G. Booch 的 OOD (object-oriented design)<sup>[5]</sup>; Rumbaugh 的 OMT (object modeling technique)<sup>[2]</sup>; Peter Coad 和 Ed Yourdon 的 OOA (object-oriented analysis)<sup>[6]</sup>; ESA 的 HOOD (hierachical object oriented design)<sup>[7]</sup> 和 Wirfs-Brock 等的 RDD (responsibility driven design).<sup>[8]</sup> 这些方法各有特色, 适用于不同的应用环境, 很难说孰优孰劣.

同这些方法相比, OOAnalysis 主要有以下特点: 整个分析过程在不同抽象层次递归完成, 每个递归内部可以是并行进行的. 既支持自顶向下的系统分解和对象精化, 又支持自底

向上的系统构造和对象抽象.既能刻画系统的静态特性又能刻画系统的动态特性.把对象间的关系提高到和对象相同的层次来刻画.能建立多层次、多侧面的可视化模型.具有自动布线、自动模型检查、自动文档生成等自动化功能.灵活的图形界面接口和图形编辑器.灵活的对外接口和多种输出模式.

在完成 OOAnalysis 以后,我们觉得还有许多工作可以进一步研究,例如:其它分析方法的结果到 OOAnalysis 的自动转换.加强刻画活动对象的特性,如时序性、死锁检测等;形式化的约束和模型检查.

### 参考文献

- 1 冯玉琳,黄涛.面向对象的组合软件技术研究.中国科学院软件研究所学术报告,1995.
- 2 Rumbaugh J, Blaha M, Premerlam W *et al.* Object-oriented modeling and design. Englewood Cliffs NJ: Prentice-Hall, 1991.
- 3 庄文君等.集成电路布线设计自动化.上海:上海交通大学出版社,1986.
- 4 Motif Programmer's Guide. Ver 1.2 2nd ed. 1992.
- 5 Booch G. Object-oriented design and application. Redwood City: Benjamin/Cumrings, 1991.
- 6 Coad P, Yourdon E. Object-oriented analysis. 2nd ed., Englewood Cliffs NJ: Prentice-Hall, 1991.
- 7 HOOD User Mannel Issue3.0 WME/89-353/JB. HOOD Work Group. European Space Agency, September, 1989.
- 8 Wirfs-Brock R, Wikerson B, Wiener L C. Designing object-oriented software. Englewood Cliffs NJ: Prentice-Hall, 1990.

## DESIGN AND IMPLEMENTATION OF OOAnalysis

Ni Bin Feng Yulin Huang Tao

(Laboratory of Computer Science Institute of Software The Chinese Academy of Sciences Beijing 100080)

**Abstract** SCOP is the component software that supports object-oriented composition software engineering. OOAnalysis is a tool in SCOP. This paper introduces the design and implementation of OOAnalysis. This tool adapts the parallel/recursive object-oriented analysis method, combining the bottom-up object construction and the top-down object reification together and emphasizing multi-level abstraction, parallel work and recursive modeling. It is an automatic tool with visualization effects in multi-level way.

**Key words** Object-oriented analysis, object construction, object reification, multi-level abstraction, parallel/recursive development, automation.