

一种面向对象的可重用库管理系统的模型

王青 张为民 蔡建平

(北京系统工程研究所 北京 100101)

摘要 研究软件的可重用技术是提高软件生产率、解决软件危机的良好途径。好的重用技术可以带来高可靠、高性能、高质量和高效率的软件新系统。重用是支持软件工程新范型的基本前提。本文介绍了一种支持基于重用的、开放的、不依赖于具体领域的可重用部件管理系统。

关键词 Ada 程序设计语言, 重用, 面向对象, 软件工程, 软件库。

中图分类号 TP31

随着软件规模变得越来越大, 控制软件开发的费用、提高软件开发的效率、改进软件开发质量、保证软件开发的一致性越来越难。造成这种现象的一个重要原因就是软件开发组织对相同或相似系统做着大量重复的开发, 以前的成果和经验没有得到充分的利用。大量的软件开发组织由于技术、文化、合同、法律、经济、政治等诸方面的原因, 无法或者很少重用已有的成果, 改变这种状况的主要措施是提供一种手段, 促进软件开发组织从现在这种“从零开始, 白手起家”的软件开发范型, 尽快地过渡到“过程驱动、面向领域和基于重用”^[1]的软件开发范型。这就是一种新型的基于重用的软件工程范型, 亦即“超大规模程序设计”。

根据这种新型的软件工程范型开发出来的软件系统具有开发效率高、性能稳定、质量可靠、风格一致且可以规模宏大等优点。研究软件的重用技术是研究这种新型的软件工程范型的基本前提。重用现有的信息部件不仅需要建立大量可重用的信息部件, 而且需要对这些信息部件进行有效的管理, 这就是我们提出研究可重用库管理系统的—个基本背景。

1 面向对象的可重用部件库的模型

“过程驱动、面向领域和基于重用”的软件工程范型的中心思想就是可重用的信息部件和管理这些信息部件的可重用库管理系统。

可重用部件库管理系统一般由 3 个层次的内容构成: 可重用部件库的模型; 可重用部件库; 可重用部件库、可重用部件的管理和操作工具。

这里面, 可重用部件库的建模是最重要和基本的工作, 库模型的好坏直接关系到库管理

• 作者王青, 女, 1964 年生, 工程师, 主要研究领域为计算机软件。张为民, 1966 年生, 助研, 主要研究领域为 Ada 语言, 软件工程, CASE。蔡建平, 1960 年生, 副研究员, 主要研究领域为 Ada 语言, Ada 软件工程与环境, Ada 实时嵌入式开发。

本文通讯联系人: 王青, 北京 100080, 中国科学院软件研究所二部

本文 1996-06-27 收到修改稿

的操作的实用性和有效性。

利用面向对象^[2]的设计思想构造的可重用部件库模型,把可重用的部件、工具或子系统看作类或对象,这些类和对象就是库的基本元素。库模型不仅描述了可重用库的构造和元素,同时还描述了这些元素之间的关系。

它打破了传统的软件库^[3],单纯地以类简单的划分和组织软件部件,各软件部件无内部关系进行联系和约束的格局。

1.1 可重用部件库的元素

可重用部件库的元素就是可重用的信息部件。根据面向对象的思想,这些元素被分为2种:类和对象。

类是对一类事物的一般抽象,是对一种事物的分类。类可以很抽象,也可以很具体。在本系统提出的可重用部件库的模型中,用树型层次结构表示类的抽象程度。位于最顶层的类是抽象层次最高的类,最低层的类是最具体的类,中间层次的类抽象级逐渐向低过渡。层次较高的类是位于其下的类的抽象,层次较低的类是位于其上的类的具体化。随着层次由高向低,抽象的级别也由高向低过渡。这种层次关系表达了从一般到特殊的演变过程。

对象是对一个特殊事物的具体表示。它不对一种事物进行分类,而是对类所描述的抽象事物进行具体化或实例化。

每个可重用的部件都是一个类或一个对象。根据面向对象的思想,这些类或对象应该由属性和动作构成。

1.1.1 属性

根据面向对象的设计思想,属性是每个对象固有的特性,它刻画了每个对象私有的、不同于其它对象的性质和特征。在可重用部件库中,每个可重用部件的属性应该是该部件的各个可视的方面。对象可以有预定义的属性,还可以有用户定义的属性。预定义属性是每个对象都具有的,刻画对象一些普通性质的属性。诸如:对象名称、对象的简单描述等等。主要有部件名称;部件分类索引词;部件的应用平台;部件的简单描述;部件的上下文关系;部件的重用接口。

用户定义属性指用户定义的、刻画对象一些特殊性质的属性。从软件工程学的角度出发,在可重用部件库中,可重用部件的用户定义属性可是可重用部件的各级文档。用户定义的属性可多可少,可有可无,完全可以依赖于面向领域的具体的可重用部件库。用户定义属性使得可重用部件的属性极大地丰富和灵活起来。

1.1.2 动作

动作是施加于对象之上的操作,是对象属性得以表现的载体。外界只有通过动作才能看见对象的属性,才能对对象进行操作和使用。

同样,动作也有预定义动作和用户定义动作。预定义动作是系统定义的动作,是每一个对象都可以执行的动作。在本模型中,定义了与浏览可重用部件和使用可重用部件有关的预定义动作。具体是:

- 浏览可重用部件的属性,包括预定义属性和用户定义属性;
- 提取用户感兴趣的、可重用部件的属性内容;
- 拷贝可重用的信息片段

拷贝完整的可重用部件的属性

- 根据可重用部件的依赖关系,抽取可编译运行的闭包的程序单元.可重用部件之间的依赖关系,可以通过可重用部件的描述模板来定义.
- 快速检索需要的可重用部件.

用户定义动作是用户定义的与具体对象有关的动作,它可以是任意可执行的动作.一个对象可以定义自己的动作,同时还继承其父类定义的动作,它自己定义的动作还将被其子类所继承.子类和父类的继承关系,由动作调用机制体现出来.

1.2 可重用部件库各元素之间的关系

本模型中各元素之间共有 4 种关系,即包含关系、继承关系、实例关系、依赖关系.

包含关系描述了一般到特殊的关系,如果元素 B 是元素 A 的特殊化,则 B 是 A 的子类,亦即 B 包含于 A.继承关系指子类可以继承父类乃至祖父类定义的所有动作.实例关系指每个对象都是某个类的实例,如果元素 A 是类,元素 B 是类 A 的对象,则 B 和 A 就是实例关系.依赖关系指当部件是一些程序编译单元时,各有关编译单元之间必定有编译依赖关系.因此,要重用某个编译单元时,必须要同时重用它所依赖的其它编译单元.

2 面向对象的可重用部件库管理系统

有了可重用部件库的框架模型以后,本文提出了基于这个模型的可重用部件库管理系统.它可以对可重用部件库进行管理维护、浏览、检索和使用.该系统主要可分为以下 6 个部分:可重用部件的描述模板;可重用部件库管理维护子系统;可重用库浏览子系统;动作调用机制;可重用部件使用子系统;可重用部件检索子系统.

2.1 可重用部件的描述模板

如何描述一个可重用的部件、并将其放进库中的适当位置是构造可重用部件库的一个关键因素.在本系统中,提供了一个描述模板,用来定义可重用部件的属性和动作.在该模板中,可以对可重用部件进行以下描述:部件名称;部件分类索引词;部件的应用平台;部件的简单描述;部件的上下文关系;部件的重用接口;用户定义属性;用户定义动作.

前面已经介绍,在我们的可重用部件库中,将每一个可重用的部件看成是一个类或对象.根据面向对象的设计思想,每个类或对象都应该有属性和动作,通过描述模板,用户将每个欲放入重用库的部件,刻画成一个类或对象,并使之具有对象化的特征.

2.2 可重用库管理维护子系统

可重用库的管理维护子系统,主要完成建库和对库的常规维护,以及对可重用部件的描述、插入和删除等操作.

在前面我们已提出,这种基于面向对象的思想构造的可重用部件库,将可重用的部件看作是库中的基本元素,所以在建库和描述可重用部件时,除了描述它的属性和动作以外,还要描述各元素之间的关系.这些关系构成了可重用部件库中各元素之间的内在联系.建立和描述好的可重用部件库,马上可以通过可视化的浏览机制,以树型结构显示在系统的主客户区中.

2.3 可重用库浏览子系统

本文提出的可重用部件库管理系统,基于 MS WINDOWS 3.1 的开发环境.它的浏览

机制非常友善和方便,良好地体现了用户界面可视化的技术。

本系统提出的可重用部件库的模型^[4,6]中,用树型结构表现各可重用部件之间的关系,它可以很明显地表示出各部件的层次关系和继承关系。

基于本系统提出的可重用部件库的模型,浏览子系统的宗旨是允许用户随意方便地浏览可重用部件库中的任意一个可重用部件。它以树形方式将可重用部件库显示在系统的主客户区,每棵子树可以任意扩展和收缩。而且,用户可以用鼠标或键盘直接选择定位到任意一个可重用的部件。

选取一个可重用部件库后,可以马上在系统的状态行显示出该部件的简单描述。同时还能执行以下操作:弹出该部件的动作菜单,并选取和执行相应的动作;浏览该部件的属性;在系统菜单条选取相应动作,执行对该部件的描述修改,并可在当前部件位置插入和删除可重用的部件。

在部件的动作菜单中,同时可以列出它所继承的其父类所定义的动作。选择动作后,由动作调用机制自动发出和执行这些动作。

如果在可重用部件上,构造一个知识推理库,本系统还可以提供一个推理机(Inference Engine),根据重用工程师提供的推理逻辑,指导用户应该浏览的下一分支,亦即“导航式浏览”。这样,当用户浏览可重用部件库时,如果发现某个可重用部件存在有附加的推理模块,就可以通过本系统提供的推理机,利用推理模块中的知识进行推理,决定下一步的浏览方向。这种“导航式浏览”机制为可重用部件库的用户提供了一些智能化的辅助决策手段。

2.4 可重用部件库的动作调用机制

根据面向对象的思想,一个对象的动作是指可以施加于该对象上的操作,它一般可以分为2类:预定义动作和用户定义动作。

在本文提出的可重用部件库管理系统中,对象就是每一个可重用的部件,所以这里提到的动作也就是可施加于可重用部件上的操作。

在本系统中,可重用部件的预定义动作主要有以下3种,即浏览可重用部件的属性,包括预定义属性和用户定义属性;提取用户感兴趣的、可重用部件的属性内容;根据可重用部件的依赖关系,抽取可编译运行的闭包的程序单元。可重用部件之间的依赖关系,可以通过可重用部件的描述模板来定义。

预定义动作是由系统实现的,每个部件都是可以执行的动作。

可重用部件的用户定义动作是该可重用部件和其父类自己定义的,只有它们自己才能执行的动作。那么,子类如何调用父类定义的动作呢?这是本系统所要实现的动作调用机制。

动作调用机制是指用一个动作链来表示子类、父类以及祖父类所定义的动作之间的关系,对于使用可重用部件库的用户,动作的定义和发出都是透明的。他并不关心动作是在哪定义以及如何定义的,他只需弹出部件的动作菜单,选取感兴趣的动作,双点鼠标即可执行该动作了。而且用户还不用关心究竟动作是本部件定义的,还是它继承而来的。对于继承而来的带参数的动作,动作调用机制自动用当前部件的属性匹配需要的参数。

这样,通过动作调用机制就可以很方便地执行任何可以施加于部件上的操作了。使得该系统的设计完全符合面向对象的设计思想。

2.5 可重用部件使用子系统

从理论上说,对可重用部件的使用分 3 个层次,即拷贝可重用的信息片段;拷贝完整的可重用部件的属性;拷贝可重用的部件及重用环境。

2.5.1 拷贝可重用的信息片段

所谓拷贝可重用的信息片段,即是我们通常所说的现场阅读.用户可以通过属性浏览器,浏览阅读部件的各种属性,然后将感兴趣的内容拷贝下来。

2.5.2 拷贝完整的可重用部件的属性

所谓拷贝完整的可重用部件的属性,即是我们通常所说的借阅.同样用户可以通过属性浏览器,浏览阅读部件的各种属性,然后将感兴趣的属性完整地拷贝下来。

2.5.3 拷贝可重用的部件及重用环境

拷贝可重用的部件及重用环境是使用可重用部件的最高层次.熟悉高级程序设计语言的人们都知道,一个可运行的软件系统,通常由多个相对独立的编译单元组成,这些编译单元之间就存在着编译依赖关系,而对于 Ada 这样的非常严格的程序设计语言来说,不仅有编译依赖关系,还存在编译次序的依赖关系,

本系统是一个基于重用的部件库管理系统,这个库中的基本元素是可重用的信息部件,从重用的角度出发,这些信息部件中最重要的就是程序编码.为了重用,可重用部件的编码准则之一就是应该具有较高的抽象层次.因此,一个可重用的信息部件通常都是一个相对独立的抽象的编译单元,亦即面向对象的设计思想中的信息隐蔽和数据抽象.基于此,一些相关的编译单元之间就必然存在依赖关系.那么,在希望重用一个可重用的信息部件时,就必然要同时用到它所要依赖的其它的可重用的部件.这就是我们现在所说的重用环境.为了能够正确地使用可重用的信息部件,必须要将该部件的完整信息以及相关的环境全部拿走.亦即需要重用一个完整的程序闭包。

本系统抽取的程序闭包,不仅可以完全包纳所有相关依赖的程序单元,而且可以根据可重用部件的构造者所描述的依赖关系的依赖次序安排闭包中各程序单元的前后编译次序.这样,它不仅支持诸如 C, BASIC 这样的灵活的、要求不严格的高级程序设计语言,还支持象 Ada 这样的基于面向对象思想的严格的高级程序设计语言。

2.6 可重用部件检索

部件查找是以可重用部件的索引词为关键字,查找与之匹配的可重用部件.这种手段可以使用户快速定位到他所需要的可重用部件。

3 结束语

重用的基础是建立大量可理解的、符合重用准则的可重用的面向各种具体领域的信息部件.而重用的手段则是必须建立结构良好的可重用部件库,以及基于这种库模型结构的可重用部件库的管理系统。

很多大系统的开发需要快速、高效而且要求高质量、高可靠和高性能.利用现有的、可理解的、已证明是可靠并且高质量、高性能的软件部件,按照超大规模程序设计的软件工程范型进行开发,是满足这些大型系统要求的良好途径。

致谢 本系统的设计和完成是在北京系统工程研究所副总工程师姜静波同志的直接领导和

参与下进行的,在此我们表示衷心的感谢!

参考文献

- 1 姜静波,郭勇,蔡建平. 面向对象的可重用库框架. 国防系统分析与软件, 1995,8(3):29~36.
- 2 Grady Booth. Software components with Ada. 1987.
- 3 Utrecht. Software re-use. Workshops in Computing, 1989.
- 4 A Methodology for the Design of Reuse Engineered Ada Components. Ada Letter, 1991.
- 5 Designing a class hierarchy for domain representation and reusability. In: Proceeding of Tools'89, 1989.
- 6 Greg Cicca. Reuse-system: software repository tool concepts. Ada Letters, 1991.

A SYSTEM MODEL OF OBJECT-ORIENTED REUSE COMPONENTS LIBRARY MANAGER

WANG Qing ZHANG Weimin CAI Jianping

(Beijing Institute of System Engineering Beijing 100101)

Abstract Researching on the reuse-technology of software is the better way of increasing the software productivity. The good reuse-technology can produce the new software system with high reliability, high performance, high quality and high efficiency. Reusing is the basis of the new software engineer development model. This paper introduces a new reuse components library manager system that reuse-base, open and non-dependent special domain.

Key words Ada programming language, reuse, object-oriented, software engineering, software library.

Class number TP31

1997 国际未来软件技术学术研讨会

International Symposium on Future Technology (ISFST-97)

征文通知

第二届国际未来软件技术学术研讨会将于1997年10月29~31日在厦门市举行.会议得到日本软件工程师协会、联合国大学国际软件技术研究所的支持.会议的主题是“面向21世纪的软件技术”.它包括以下内容:大规模软件系统的工程化;混合和分布/并行系统;重向工程和逆向工程;领域工程;软件结构;面向对象和重用组件技术;软件过程和管理;多媒体和CSCW;智能浏览和Internet;技术传播;计算机辅助软件工程;人机交互等.

论文截稿日期:1997年7月15日 发出录用和修改通知单:1997年8月15日

收到正式录用稿全文:1997年9月15日

征文要求:论文的理论研究和开发成果未曾公开发表过,论文需一式6份,用英文书写.被录用稿件的作者至少有一位必须参加会议.

来稿请寄:100080 北京2719信箱 中科院北京软件工程研制中心 钟锡昌 研究员

电话及传真:010-62560341 E-Mail:XCZhong@sun.ihep.ac.cn