

文章编号: 1005-8451 (2003) 11-0013-03

结构化程序设计方法与面向对象的 程序设计方法的比较

王德军 郝永芳

(武汉大学 网络中心、武汉 430070)

摘 要: 随着计算机的硬件及通讯技术的发展, 计算环境发生了深刻的变化。计算环境的变迁和不断增长的软件需求对程序设计方法学提出了一个又一个的挑战, 程序设计方法学也在挑战中前进。首先回顾软件工程程序设计方法的发展历史, 指出结构化和面向对象是软件工程程序设计方法中的2个核心思想, 分析、探讨了结构化程序设计方法与面向对象的方法的区别, 并就如何在实践中正确应用给出了一些建议。

关键词: 程序设计方法; 面向对象; 结构化

中图分类号: TP311.11

文献标识码: A

Comparing structured programming method with object-oriented programming method

WANG Dejun, HAO Yongfang

(Network Center of wuhan University, Wuhan 430070)

Abstract: With the development of computer hardware and communication technology, the computing environment has been profoundly changed. The changing computing environment and the growing software requirement made greatly challenge to software engineering. It was reviewed the history of software engineering, pointed that structured and object-oriented methods were two kinds of software design method, given deeply analysis to structured and object-oriented method, and proposed some advices about how to apply those two methods to practical use.

Key words: programming method; object-oriented; structured

1 引言

随着计算机的硬件及通讯技术的发展, 计算环境发生了深刻的变化, 计算环境的变迁和不断增长的软件需求对程序设计方法学提出了一个又一个的挑战, 程序设计方法学也在挑战中前进。计算机发展经历了3个主要的阶段: 大型主机、客户/服务器以及网络计算。与此相对应, 软件工程的程序设计方法的发展可分为4代。

1.1 第一代面向过程的程序设计方法

面向软件系统的信息流程图, 采用面向过程的程序设计语言或面向进程的程序设计语言, 实现软件设计流程图所描述的信息处理过程的功能, 称为面向过程的程序设计方法或面向进程的程序设计方法。这种方法适用于设计小规模 of 专用软件包, 软件

的通用性、重用性和扩展性差。

1.2 第二代面向模块的程序设计方法

结构上将软件系统划分为若干功能模块或实体, 分别采用模块化程序设计语言, 如: pascal 编程实现, 再由各模块联结, 组合成相应结构的软件系统, 称为面向模块的程序设计方法或模块化程序设计方法, 也称为面向实体的程序设计方法。这种方法适用于设计模块化、结构化程序, 可提高软件系统的模块化和结构化水平, 设计和组装较大规模的软件系统, 有助于提高软件的通用性、重用性和扩展性。

1.3 第三代面向对象的程序设计方法

所谓对象是指具有一定结构、属性和功能的实体, 采用对象和对象类, 以及对象之间的相互通信的消息, 描述客观世界中的各种事物及其相互关系, 建立面向对象和消息的具有层次结构的世界模型。面向对象的程序设计方法基于上述面向对象世界模型。采用面向对象的程序设计语言, 如 c++、smalltalk 等编

收稿日期: 2003-05-12

作者简介: 王德军, 在读硕士研究生; 郝永芳, 工程师。

程实现。这种方法具有通用性,适用于广泛应用领域的大规模软件系统设计。有助于提高软件的可重用性、扩展性和移植性,提高编程效率和程序自动化水平。

1.4 第四代面向智体的程序设计方法

面向智体的程序设计方法是面向对象的程序设计方法的发展。

在程序设计方法的发展演变历程中,结构化和面向对象思想是最核心的思想方法。结构思想体现了人们抽象思维和复杂问题分解的基本原则与要求,而面向对象则反映了客观世界由对象组成这一本质特点。

2 软件工程程序设计方法的出发点

从程序结构来看,每个子问题形成整个程序结构的一个构件,这个构件称为一个模块。程序的算法结构,就是一个由模块连接成的层次结构。在软件工程中,把这种设计方法归结为软件工程设计方法学。该方法学的基本表述为:自顶向下,逐步求精,模块化层次结构设计。

程序设计方法的本质是问题的抽象与分解,各种程序设计方法的区别在于其分解的因子不一样,处理数据对象及相关操作的方法不一样,也就是出发点不一样。

3 结构化程序设计方法

结构化程序设计方法包含以下内容。

3.1 结构化技术

结构化技术包括结构化分析(SA)、结构化设计(SD)、结构化程序设计(SP)3方面内容,对应于软件开发时期的分析、设计和编码阶段。

3.2 结构化分析

结构化分析是70年代中期由DeMarco和Yourdon等倡导的一种基于功能分解的分析方法,即使用数据流程图、决策表、决策树等工具,来建立一种符合用户需求的结构化说明书。

3.3 结构化设计

结构化设计是一种面向数据流的设计方法,也就是采用最佳的可能方法设计系统的各个组成部分以及各成分之间的内部联系的技术,目的在于提出满足系统需求的最佳软件的结构,完成软件层次图或软件结构图。

4 面向对象的方法

面向对象技术:面向对象技术包括面向对象分析(OOA)、面向对象设计(OOD)及面向对象程序设计(OOP)3部分内容。

OOP是在结构化程序设计的基础上,于80年代初涌现的一种程序设计方法,但其真正显示力量和被产业界所重视还是最近几年的事。封装是整个OOP方法的基础,主要用于在数据段外围构造保护层,以限制外界变化的影响,所有的数据访问都由保护层内的过程间接处理。应用程序员不必再按照将程序设计语言逐句拼装的方式来构造整个软件,只需组合、重用由系统程序员开发、可供他人用来装配的软件集成块即可。例如,Visual Basic(VB)是一种面向对象的程序设计语言,与传统DOS下的Basic或Quick Basic最大的差别在于它运用了面向对象的概念。VB建立了一个事件驱动的环境,供用户直接调用。程序设计人员只要专心数据的运算处理,其余诸如Windows应用程序下所见的滚动条、按钮、下拉式菜单和对话框等,都已经有了对象供用户进行调用,而且每个对象又都有许多事件、属性和方法,供用户填入适当值或程序码,从而形成一个应用程序。

5 结构化程序设计方法与面向对象的程序设计方法的比较

结构化程序设计方法由瑞士沃斯教授最早提出为数据结构+算法=程序。这种观念自60年代的软件危机之后对程序设计方法产生了深远的影响。然而随着软件规模的不断增长,也愈来愈暴露出它的不足和缺陷。在传统的结构化程序设计中,数据存储分离与程序代码还存在着一些弊端。所有的代码被书写成结构化的,而不是模块。因为数据元素可以从任何代码中访问,它可能在开发人员不知道的情况下被修改。这可能会导致非常难于调试的运行错误。此外,程序维护可能成为一项重要的任务。理解结构化编程中修改一行代码所造成的全局影响是非常困难的。最后,依赖于开发人员控制代码和数据结果会导致较低的可重用性。事实上,采用结构化程序设计方法设计出的程序,其结构是建立在数据结构上的,数据结构是现实世界中对象间本质关系的抽象反映。而程序所要处理的现实世界中的对象是最容易变化的部分。这种变化很难被吸收于局部或某一层,而且涉

及整个程序结构。

面向对象的程序设计方法的出现为解决这些日益凸显的困难带来了曙光。在面向对象的程序设计方法中,数据结构已不再是程序的基石。取而代之的是抽象数据类型、类和对象这些概念。数据和操作数据的算法不再分离,它们被封装在一起,更接近现实世界本来面貌。类的封装性将大量的实现细节封装于内部。它带来的好处是降低了程序结构的复杂性。使程序更易于修改,也更易于实现代码的重用。类的继承性和多态性是2个最重要的特征。继承性使类能反映现实世界的层次结构,多态性反映了现实世界的复杂多样。OOP允许开发人员通过继承同时重用代码和数据。通过从先前确定的对象中继承,开发人员可以更迅速地构造复杂的应用程序。由于编写新的代码总是会有带入错误的潜在可能,重用经过测试的代码可以使产生额外错误的可能性达到最小。他们使程序的描述能力进一步提高,使真正的代码重用成为可能。如在软件开发过程中,用于界面的工作量往往达到70%甚至更多。因而出现了各种用于开发界面的函数库。问题是用户很难对之进行修改扩充。如果这么做,需要读懂函数库的源代码并进行修改。这通常是困难的,且很容易引进错误。如果提供的是类库,只需简单地继承一个类,重新定义或增加新的函数。

6 在实践中的正确应用

面向对象的程序,最根本的目的就是使程序员更好地理解和管理庞大而复杂的程序,它在结构化程序设计的基础上完成进一步的抽象。

使用面向对象的程序设计方法绝非是要摒弃现有的结构化程序设计方法,相反,它是在充分吸收结构化程序设计优点的基础上,引进了一些新的、强有力的概念,从而开创了程序设计工作的新天地。面向对象的程序设计方法把可重复使用性视为软件开发的中心问题,通过装配可重用的部件来生产软件,而不是像目前编程所用的那样,通过调用函数库中的函数来实现。这里要注意,我们是基于应用程序这一个层次来阐述这些问题的,事实上,在对象内部的实现上,我们常常使用过程式的结构化程序设计方法,也常常调用C/C++函数库中的很多有用的函数,然而从程序的总体结构上说,它是由一系列对象构成的,对象之间能够以某种方式进行通信和协作,从而实

现程序的具体功能。

结构化编程的基本思想就是把大的程序划分为若干个相对独立、功能简单的程序模块。它以过程为中心,强调的是过程,强调功能和模块化。通过一系列过程的调用和处理完成相应的任务。

面向对象编程以对象为中心,是对一系列相关对象的操纵,发送消息给对象,由对象执行相应的操作并返回结果,强调的是对象。

程序设计人员进行面向对象编程时,不再单纯地从头到尾一行一行地编写代码(但不意味着不写代码),而是考虑如何创建对象,利用对象来简化设计,使开发软件相对容易。由于面向对象程序设计具有诸多特点,能大大提高软件的开发效率。对于已经习惯于面向过程的设计人员来说,有一个转变观念和思维方法的过程。

程序设计方法关注软件生命周期的各个环节,从需求分析、总体设计到编码、测试和维护。在选择设计方法时还要注意到该设计方法各个阶段的工具,根据开发团队的知识情况来决定采用何种设计方法。

7 结束语

程序设计方法的发展经历了4代,结构化和面向对象是软件工程的程序设计方法中最本质的思想方法,结构化体现了抽象思维和复杂问题求解的基本原则,面向对象则深刻反映了客观世界由对象组成这一本质特点。各种程序设计方法的一个重要区别在于问题分解的出发点不同(因子),思维模式不同。计算机中数据结构和过程是密切相关的,结构化方法将数据结构和过程分开考虑,面向对象的方法组合数据和过程于对象之中。理论上,对象式实际方法将产生更好的模块内聚和耦合特性,使得软件更易于重用与维护,但在实践中程序设计方法需要工具 and 环境的支撑,需要考虑软件生命周期的各个环节,在选择程序设计方法时,需要综合考虑这些因素。

【参考文献】

- [1] Ronald J. Norman. Object-Oriented Systems Analysis and Design [M]. New York: Prentice-Hall, inc., 1996.
- [2] 陈世鸿. 软件工程原理及应用[M]. 武汉: 武汉大学出版社, 2002.
- [3] 郑人杰. 实用软件工程[M]. 北京: 清华大学出版社, 2001.