

应用软件的一个模型^{*}

萧金声

(中山大学计算中心, 广东 广州 510275)

摘 要: 讨论现存软件开发方法的不足, 分析其根由, 并给出一个应用软件模型。最后, 提出根据该模型建立软件开发方法的设想。

关键词: 应用软件; 开发方法; 模型

中图分类号: TP311.52 **文献标识码:** A **文章编号:** 0529-6579 (2002) 01-0103-03

凡开发一个应用软件系统, 人们总选用某种软件开发方法。一经采用, 该方法便会左右开发者的开发思路 and 开发行为, 进而影响到开发过程是否顺利、软件质量如何, 以及日后是否易于维护。可见软件开发方法对于开发应用软件是何等重要, 对开发者的影响又是多么深远。

不幸的是, 从“软件危机”出现的 20 世纪 60 年代末直到跨世纪的今天, 人们始终未能找到一套既理想又适用的软件开发方法。以至出现了一种令人尴尬的局面: 在开发应用软件时, 硬件设备、软件平台、软件工具都十分先进, 而开发方法却始终不尽人意。

这篇文章试图探讨个中原因, 以便寻求解决的途径。

1 结构化方法的失误^{*}

谈论软件开发方法, 势必首先提及结构化方法。它的使用地域广, 时间跨度也长。仅就中国而言, 结构化方法已被使用了 20 多年, 而且 1988 年发布并实施的软件开发国家规范^[1] 也以此法为基础。时至今日, 它仍然拥有大量的使用者, 其影响不可谓不大。

然而, 这个方法总是给开发者带来许多困惑, 使人难以依循。其实, 早在 20 世纪 80 年代初期, 人们已陆续发现了结构化方法的若干缺陷。诸如不适应需求的变化; 文档费时而收效不大; 可重用性差; 维护困难等。导致这些问题的原因是多方面的, 但笔者认为最关键的因素是该方法的发展过程中存在重大失误。

大家知道, 结构化软件开发方法由结构化程序设计方法延拓而来。当年, 人们推崇结构化程序设

计方法是因为对程序的本质形成了一定的认识。一个著名的书名:

算法 + 数据结构 = 程序

就表达了人们对程序的看法, 即“程序就是在数据的某些特定的表示方式和结构的基础上对抽象算法的具体描述。”^[2] 然而, 结构化开发方法的出现并非基于对应用软件的认识, 而是面对“软件危机”的应急之举, 作了直接延拓。

诚然, 应用软件也是程序, 只不过大一些。所以, 把程序设计方法直接延拓过去似乎无可厚非。殊不知, 这从小到大的量变也带来了质变, 诸如由小程序的一人使用到应用软件的多人使用、从小程序的全部运行到应用软件的局部运行等。如果不对这些质变有所认识, 怎么可能有好的、有针对性的开发方法呢? 这就是结构化开发方法的要害所在。

2 面向对象开发方法的进步

同结构化开发方法的形成类似, OO 开发方法是在 OO 程序设计方法的基础上发展起来的。从 1988 年到如今, 正式发表的 OO 开发方法几近 20 套。在延拓 OO 程序设计方法的思路的同时, 这批方法还不同程度地渗入了创建者们各自对应用软件的新认识。例如: ①事件驱动观念。对这一观点使用得最明确的当推 Fusion 方法^[3]。它安排一种系统操作对应一种输入事件: 输入事件发生, 就执行相应的系统操作。②某些系统操作有顺序约束。OMT^[4] 中的动态模型就是对顺序约束的描述; Fusion 方法中的生命周期模型也是这个作用。③OO-SE^[5] 中所述 Objectory 方法的需求阶段有一使用情况模型 (Use Case Model)。它涉及了使用者对应用软件的个别要求。

^{*} 这里只是讨论认识过程中的问题, 并不针对任何人。

收稿日期: 2001-08-25; 作者简介: 萧金声 (1938—), 教授, 男; E-mail: xiao380@public.guangzhou.gd.cn

根据以上不完全的讨论, 可以看出, OO 开发方法的创建者们已不同程度地意识到开发应用软件时, 有一些程序设计方法中所没有的东西。这是可喜的进步。只是这些认识分散在不同的方法中, 没有形成一个整体性的认识。即便是集多家之长的一体化建模语言^[6] (UML) 也仍然没有涉及对应用软件的明确认识。

可以这样说, 有如此众多的 OO 开发方法出现的事实本身, 正好说明人们对应用软件没有统一的认识。要解决好开发方法问题, 首先应该统一对应用软件的認識。

3 七元组模型

开发应用软件的一个首要问题是确定建立一个怎样的软件。这个任务的完成就在需求分析阶段。然而, 不同的开发方法却告诉人们不同的分析模型和步骤。这源于开发方法的创建者们对应用软件的确定持有不同的见解。

应用软件究竟是什么, 或者说怎样才算确定了一个应用软件? 这是一个从外观、从整体去考察应用软件时应能回答的问题。因此, 应该着眼于应用软件的系统行为。

根据自身的经验和体会, 笔者认为:

(1) 应用软件是等待外界使用的软件系统。如果外界没有请求, 它就会一直等待; 每当外界提出某种请求时, 它就因此请求而作相应的响应。这就是应用软件的工作方式。

(2) 为接受请求和响应作答, 应用软件必须同外界通讯。这就是输入事件和输出事件。

(3) 外界向应用软件提出请求有 2 种方式: 一般通过输入事件, 称为事件驱动; 少数情况是在建立系统时约定的定时启动, 称为时钟驱动。

(4) 应用软件对外界每一种驱动力的响应, 就是一种系统操作。

(5) “使用” 应用软件的是一个群体, 其中包括人和/或其他软、硬件系统。他(它)们对应用软件的使用要求通常不会完全一样。所以, 这个使用群体可以分成若干组。同一组中的个体对应用软件有相同的请求集。

(6) 应用软件的某些系统操作可能必须遵从某种顺序, 因而要求相应的输入事件也遵从这种顺序, 否则系统不予响应。

(7) 应用软件所处理的数据构成特定的应用数据集。基于以上认识, 可以把应用软件 (AS) 看作

1 个七元组:

$$AS = \{AOC, IE, OE, CD, GIE, SO, OD\}$$

其中, AOC (application object classes) 是应用对象类的非空有穷集。反映应用域中的数据; IE (input events) 是输入事件的非空有穷集。反映外界对系统的通讯; OE (output events) 是输出事件的非空有穷集。反映系统对外界的通讯; CD (clock drivers) 是时钟驱动的有穷集, 可空。反映指定的定时启动; GIE (groups of interactive entities) 是交互实体(使用者)组的非空有穷集。反映系统的使用环境; SO (system operations) 是系统操作的非空有穷集, 且 $|SO| = |IE| + |CD|$ 。反映系统对驱动力的响应; OD (order diagrams) 是顺序图的有穷集, 可空。反映部分系统操作的顺序约束。

这样, 当七元组中的每个集合确定下来之时, 一个应用软件的功能也就完全确定了。从而, 不同应用软件之间的差异就在于它们有各自的七元组。

4 结 语

有了七元组模型之后, 建立一套合理的开发方法便不会太困难了。开发方法的核心就是在需求分析阶段确定这 7 个集合; 此前的调查阶段要为确定七元组而搜集原始需求; 此后的设计和实现则是兑现七元组的工作。

根据这个思路, 我们建立了一套新的 OO 开发方法: 金石方法。在近 2 年的实践中我们高兴地看到, 这个方法的分析模型对于后续的设计来说, 既是充分的, 也是必要的。这正好说明, 七元组模型符合应用软件开发客观实际。

参考文献:

- [1] 中国标准局. 计算机软件开发规范 GB8566-88[S]. 北京: 中国标准出版社, 1989.
- [2] NIKLAMS W. Algorithms + Data = Programs[M]. NJ: Prentice-Hall, 1976.
- [3] COLEMAN D, AMOLD P, BODOFF S, et al. Object-oriented development — the fusion method[M]. NJ: Prentice-Hall, 1994.
- [4] RUMBAUGH J, BLAHA M, PREMERLANI W, et al. Object-oriented modeling and design[M]. NJ: Prentice-Hall, 1991.
- [5] JACOBSON I. Object-oriented software engineering[M]. MA: Addison-Wesley, Reading, 1992.
- [6] FOWLER M, KENDALL. UML distilled-applying the standard object modeling language[M]. MA: Addison Wesley, Reading, 1997.

(转第 109 页)

Vapor liquid Equilibrium of 1, 2, 4-Trimethyl Benzene or 1, 3, 5-Trimethyl Benzene+t Butanol

OUYANG Gang feng, HUANG Zhong qi, KANG Bei sheng

(School of Chemistry and Chemical Engineering, Sun Yat sen (Zhongshan) University, Guangzhou 510275, China)

Abstract: Vapor liquid equilibrium data were determined in a CP-1 still at 99.992 kPa for 2-propanol+1, 2, 4-trimethyl benzene and 2-propanol+1, 3, 5-trimethyl benzene respectively. A thermodynamic model proposed by Hayden-O'Connell for accounting the nonideal behavior of the vapor phase was used for calculating the activity coefficient of liquid phase. All the experimental data were tested by the method proposed by Herington. The results show that they conform to the demand of thermodynamic consistency. Wilson equation was selected to correct composition and the activity coefficient of the components in liquid phase. The calculated values of the vapor composition are in good agreement with experiment data.

Key words: vapor liquid equilibrium; activity coefficient; thermodynamic consistency

(接第 104 页)

A Model of Application Software

XIAO Jin sheng

(Computer Center,

Sun Yat sen (Zhongshan) University, Guangzhou 510275, China)

Abstract: The deficiency of extant software development methods is discussed, and the origin of the deficiency is analysed. An application software model is also presented. Finally, an idea to set up a software development method by that model is advanced.

Key words: application software; development method; model

(接第 106 页)

Studies of Blue Emission of n-Type GaN

LI Xin, PENG Lin-feng, HUANG Qi-wen, LAI Tian-shu, LIN Wei-zhu

(State Key Lab of Ultrafast Laser Spectroscopy //Department of Physics, Sun Yat sen (Zhongshan) University, Guangzhou 510275, China)

Abstract: The spectral characterized techniques were used to study the mechanisms of blue emission in undoped n-type GaN. A model for 427~496 nm (2.5~2.9 eV) emission is presented. The blue emission is considered as a contribution from the donor states originated from the interstitials.

Key words: GaN; donor; blue emission