第18卷 第1期
1997年 3 月
郑州工业大学学报
Journal of Zhengzhou University of Technology
Vol.18 No.1.
Mar. 1997

一种以面向对象及形式化技术为基础的严格的软件开发方法

吴会松

(华北石油勘察设计研究院 计算机应用研究室 062552)

摘 要 随着软件开发规模的迅速扩大，目前流行的软件开发方法(包括当前的软件工程法)已难以满足开发实践的要求，实践要求在软件开发方法学上能有新的进展。鉴于此，本文在面向对象及开式化方法的基础上，系统地提出了一种严格的软件开发方法，力图使开发者的创造性及开发环境的自动化能力都得到充分发挥，使软件开发能适应形势的需要。

关键词 软件开发方法 面向对象 形式化方法 软件重用 程序变换 系统验证

中图分类号 TP311

1 实践对软件开发方法的技术要求

1.1 能最大限度地发挥工具的"自动化能力"及"人的创造性"

用哲学的眼光看，所有的软件过程都存在着这两大因素(有时也是一对矛盾)，区分这两大因素的过程，就是确定人机分工(包括确定人机界面)的过程。对于那种变化不定、很难用模型描述、需要经验和直观推理的工作，原则上应由人来完成。对于软件开发，必须考虑人的因素，这是一个很活跃的成份。一般认为，人具有"万能"的能力，能很快地发现问题的实质，找到解决问题的方法，这是积极的一面；但多数人对单调、重复性的工作没有耐心，对一些问题还总带着固有的看法(也可以说是偏见)，而且人的"运算"速度慢是引起问题的一大原因。因此，合理分布工具的"自动化"与人的"创造性"，充分发挥各部门的作用，使总体效能达到最佳，是软件开发中必须考虑的问题。

1.2 能真实地描述、模拟客观事物，用人类的思维及解决问题的方式处理问题

我们所说的"对客观事物的认识"，包括了事物的关系、事物的内部和外部特征。在软件开发中，突出不同的重点，就能引入不同的软件设计方法；若以突出事物间的关系作为软件开发的出发点，引入的就是面向关系的设计方法(包括相应的逻辑语言)；若以突出事物的观点出发，引入的就是面向对象的设计方法(包括相应的程序语言)。各种方法各有所长。相对而言，"面向对象"更接近人的常规思维习贯，是当今的潮流。

本文根据上述两个原则，系统地提出了一种严格的软件开发方法。它的基本思想是在面向对象的基础上，使逐步求精与软件重用相结合，把基于图形的形式化演算法与逻辑转换系统融为一体，使"自动化"与"创造性"得到充分的发挥。

收稿日期：1995－11－12

2 有关方法学的基本思想

2.1 从上而下的抽象和分解是人们在分析问题时为降低分析问题的难度而自然采取的两个基本思想。它们也很自然的体现在软件开发中。

对软件开发而言，抽象的含义有两种：a. 分析复杂问题时，在分出本质与非本质的基础上，根据问题的实质，清楚地描述问题的结构，进而把非结构化的问题转化成结构化问题；b. 在对系统模块化时，在上层模块中概括低层模块，从而产生上层模块的抽象机制。

程序设计中的模块化和信息就是"分解"的具体体现。从软件开发的过程看，分解其实是一种从上而下的、从抽象到具体的途径之一，是一种逐步细化的过程。"合成"则是其逆过程，它反映了分解后（如概要设计完成后）软件的总体实现的过程（一般是自下而上）。

对软件开发者来说，一个重要的环节就是"软件设计"，这是个对问题的求解过程，也是认知解决问题所需要的知识体系的过程。这应该是一种创造性的劳动。在"抽象"、"分解"与"合成"中充分体现了这种创造性。

软件开发的智力密集度相当高，随着技术工具的增多，"智力密集"所反映的"不断学习"和"创造性"成份也日益增多。软件开发首先是认识客观事物，需求分析就是认识过程的开始。需求分析完成后，认识的过程仍在继续，直到软件"死亡"。实践表明，把认识与实现结合起来（即在对客观事物有了一定的认识后，就开始做一些功能的设计与"施工"，然后再认识，再"施工"…），可以简化总体认识与实现的复杂性，也符合"实践…认识…再实践…再认识"的一般规律。人们都希望有一个能完全符合人类认识规律的指导软件开发的工作模式，能体现开放性的软件开发过程。

2.2 对形式化方法的说明

我们所说的形式化方法是指：a. 由数学直接推导出的技术方法（包括一些概念）；b. 实现、使用这些技术的方法学途径。这两者的关系往往是一对多的：一种形式化的技术往往有多种应用途径。但各途径的良莠程度相差很大，因此，应用中最关键的往往是找出一条合理的途径。

形式化方法的主要作用是用规范化语言编写开发文档及各部门的接口，并能在抽象的基础上对设计决策及关键算法作出检查及验证，在此基础上产生软件的体系结构。目前软件工程上的形式化可分成以下几类：

a. 非形式化及半形式化方法。包括在一些领域仍很流行的结构化分析、设计法，它们往往是基于图形的软件开发方法。

b. 基于形式化概念的方法。

c. 严格的软件开发方法。

d. 具有数学化的规范语言、提供可支持定理证明及检查开发环境的完全形式化的开发方法。

目前在软件开发的需求阶段使用最多的还是结构化分析方法，但普遍认为未来的主流是面向对象方法[3,5]。笔者认为，如能将面向对象方法与形式化方法结合起来，必然会产生更好的效果。

3 基于面向对象和形式化方法的、严格的软件开发方法

笔者在这里提出一个基于面向对象和形式化方法的、严格的软件开发方法（见图1），它

在形式化技术的基础上对"自动化"与"创造性"做了合理的折中，并集成了目前公认比较优秀的软件开发技术。在这种方法指导下的软件开发具有很好的形式化基础。该方法的软件开发步骤有三个：以面向对象为基础的需求分析及建模，生成目标软件体系结构，软件变换及重用。

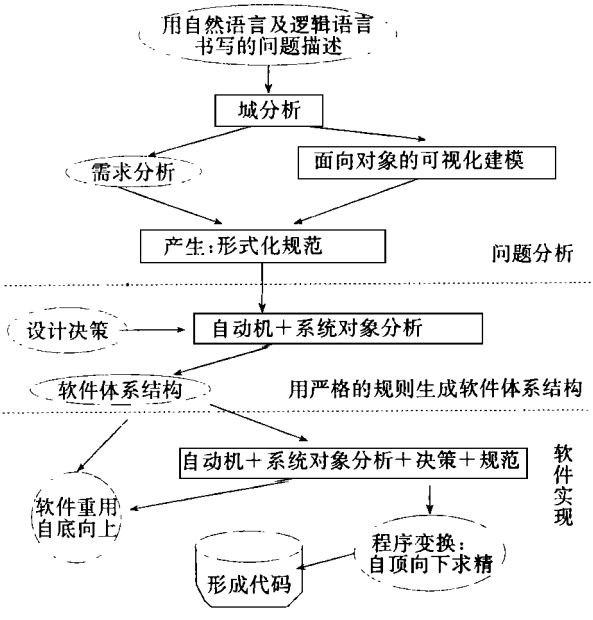

图 1　软件开发方法示意简图

3.1　以面向对象为基础的需求分析及可视化建模方法

需求分析的根本目的，是抽象出系统的特征，用准确、严格的方式描述需求，为目标系统建立模型。鉴于此，这里在面向对象的基础上，提出了基于域分析而进行的需求分析及建模方法，包括以自然语言为基础的识别域元素的方法[2]、面向对象的可视化建模方法以及逻辑化的全局需求规范化方法。

3.1.1　自然语言与域分析中相对应的概念

自然语言与域分析中相应的概念

自然语言中有这么几种必用的概念：宾语、名词、代词、及动物词、副词、形容词。这里先给出这些概念与域分析中概念的对应关系：

实体……若是指能完成的行为，对应主语；若指能用来完成或能接受的行为，对应宾语；…

动作……指能改变系统状态的操作，与之相应的是主语、及物动词及宾语；

关系……与之相关的是事实陈述及不及物概念。

3.1.2　面向对象的可视化建模方法

结合域识别，这里用面向对象的图的描述方法建模。

a. 识别对象的策略

找出意义明确的实体，用聚集抽象的方法给对象分层；用统一的名称代替别名；去除非最终产品的对象实体。

b. 识别对象行为的策略

若对象间出现了请求关系时，那么被请求的对象必须有能接受该请求的接口；若对象需要处理该对象外部的事件，那么这个对象必须有能接受这些事件的接口。

c. 识别特定对象的、与特定处理相关的属性，根据其值确定对象的状态。

d. 对象状态转换建模

根据有穷自动机原理，描述对象状态、事件、传递及因果关系。

e. 认识特定对象所属的类

根据面向对象理论中的分类原理，担负出具有相似性质及行为的对象，利用继承、聚集的方法组织不同的类。

为满足上述处理，目前已提供可视化语言来处理文档及建模，并能用多视点方式处理对问题的全面描述。

3．1．3 逻辑化的全局需求规范化方法

对于全局的需求，这里用时态逻辑的规范进行定义。对于对象本身的待精化问题，也应定义成一个子问题。在整个的需求分析阶段，需要根据时态逻辑建立目标模型的时态语言。要在尽可能早的阶段进行容易操作的系统建模，这样可以加强甲乙双方的交流、降低需求分析的复杂度，加地所处理问题的了解。

3．2 用严格的规则生成软件体系结构

在需求分析及建模后，我们得到的是个形式化的初始规范。其后续的实现有多种。按照形式化处理的思想，可以由规范加设计决策在各开发的子阶段导出其实现。在设计决策中可以充分体现开发者的创造性。目标软件的体系结构根据目标系统的模型及设计决策逐步生成，其实现必须是严格的，这应该在两方面体现：a．体系结构与需求规范应保持其一致性；b．体系结构内部应保持其一致性。为验证、维护这种一致性，需要一种基于蕴含式证明的 "真值维护系统" (用于证明并保证某规范是另一规范的实现、某规范应满足某些特定的性质等) [9]。

这里要特别说明一下软件体系结构的基本设计导向：

a．根据设计决策确定软件的体系结构及开发策略；例如，对于一般的财务凭证处理，可以采用 "对收款凭证、付款凭证、转帐凭证分别编号" 的方式处理，也可以采用 "对收款凭证、付款凭证、转帐凭证统一编号" 的处理方式。

b．尽可能保证基于系统模型的软件体系结构具有良好的可操作性及可实现性。

3．3 以程序变换和软件重用为基础的实现技术

在软件实现中，本方法采用了自顶向下和自底向上的两种方法，它们与以程序变换和软件重用为基础的实现技术相对应，在软件开发阶段起到了互补作用。

a．自底向上…软件重用[6]

为了加快开发进度、减少重复劳动，合理地重用已有软件是必须的。因此，对于过去产生的可靠软件 (部件) ，开发环境应提供实用手段，使开发者能够在开发中选用能满足需要的已有构造当前开发软件的部件，否则就用自顶向下逐步求精(程序变换) 的方法开发之。见图 2。

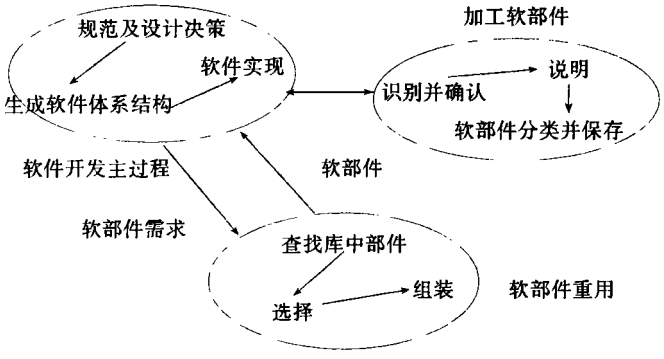

图 2 软件重用的基本过程

显然，软件重用包含了 "分解" 与 "合成" 两部分内容，它们作为互补手段在软件重用中发挥了重大作用。

b．自顶向下求精…程序变换

因为软件重用不能解决所有问题，因此还需要与之对应的自顶向下求精的程序变换技术。在变换时，逐步的向软件体系结构中加入相关的设计决策。

3．4　在开发中对性能指标分解的一般策略

一般而言，性能指标的分解依赖于功能指标的分解，可以认为是在功能规范上的性能指标。我们在做逐步分解时，把功能指标和性能指标分离，再根据功能需求的分解，产生新的子功能的性能指标。

4　本方法应具备的开发工具

在一般情况下，无论是方法产生于工具之前，还是根据已有工具提出新的方法，最后都得通过工具来达到方法的实现。所以，任何方法都离不开工具。对于在上面提出的开发方法，我们还需要根据已有条件集成一个软件工具箱，产生一个集成的环境。

4．1　需求分析工具[2,9]

这是以 WINDOWS 或 UNIX 为平台建立的一套需求分析工具，它具有的基本功能是：

a．可以在自然语言的说明中提取域元素做面向对象的分析；

b．多视的建模功能（其模型应包括多种对象模型）；

c．维护需求获取过程一致性的功能。

4．2　系统设计工具箱[2,9]

所谓"系统设计"，是指从系统模型到软件体系结构的转化过程。很显然，这个过程的自动化程度越高越好（不需要太多的"创造性）。目前已经能够提供结构化的基于 Statecharts 的规范化生成工具。为了兼容传统的结构化设计，还可以同时采用由数据流图生成结构化规范及结构图的转化工具。基于高阶定理证明器的"证明开发环境"对系统设计有着不可低估的。

为了支持面向对象的原则，这个阶段选用的开发工具应支持层次化的求精和抽象。

4．3　软件重用工具[6]

针对软件体系结构所需的各个软件部件，需要提供软件重用的手段以大幅度提高开发效率。目前可以提供基于域分析和规范的软件部件的浏览、提取的工具，以及在规范化的基础上充实软部件库的"补充工具"。

4．4　系统验证工具[11,12]

严格的软件开发方法要求能对软件开发的各个步骤进行验证（包括传统意义上的验证）。目前已有的"命题时态逻辑证明器"已能够完成抽象的需求规范和设计决策的验证。基于高阶定理证明器的"时实系统验证工具"对及时验证软件系统也有重要作用。

4．5　软件开发环境

面向对象是本方法的基础，因此还应提供一个能支持面向对象的（如能进行类库存取、分类封装等）、能有机的把上述工具结为一体的基本环境，从而在整体上支持面向对象框架法的开发。这只要以当前已有的面向对象系统（如 Visual Basic、Visual FoxPro 等）为平台，在 OLE 的支持下是不难做到的。

5　结语

本文提出的严格的软件开发方法，与现有的软件工程方法有很大不同。传统方法因受开发工具（包括硬件及软件）的限制，在"自动化"和"创造性"方面都受到很大限制。本方法在这两方面都有很大程度的扩展。本方法在注重人的创造性的同时，还强化了有相当数学基础的"自动化"。与完全形式化的开发方法相比，本方法采用的面向对象、多视点技术及自动建模等为充分发挥软件人员的创造性提供了有益的条件。

就软件开发方法本身来说，它应提供足够多的开发策略及工具，以达到选择及效率上的最佳

值。目前的支持工具多为离散的"散兵"，还需要做进一步的集成优化，以适应更复杂的软件开发。另外还有一些理论上的问题需要进一步的突破，如软件规范一致性程度的极限等。

参考文献

1 冯玉琳、钟萃豪、陈友君·程序设计方法学·北京：科学技术出版社，1989.

2 陈晓桦、陈火旺. 从自然语言描述的需求提取形式规范. 中国计算机学会第八届年会论文集，1992.

3 李师贤、乔琳。面向对象软件工程开发的管理，计算机工程及应用。1995 (2)

4 陈火旺、罗朝辉、马庆鸣. 程序设计方法学基础. 长沙：湖南科技出版社，1987.

5 费翔林、张帆。面向对象抗议法综述，小型微型计算机系统。1995 (9)

6 毛新军、齐治昌。软件重用研究与应用。计算机科学，1994，(4)

7 罗铁庚、齐治昌。PC/CASE：支持软件开发过程的 CASE 工具，计算机科学，1994，(6)。

8 谭庆平、陈火旺。基于类型理论的递归元程序设计，软件学报，1994，5 (8)。

9 王献昌、陈火旺。真值维护系统的语义研究。中国科学 (A)，1993，23 (11)。

10 杜中平、刘春辉、张宏霞。面向对象的应用程序生成器，小型微型计算机系统。1995 (3)

11 Kerong Ben，Chen Huowang，Wang Binshan PLT Sequent calculus system. Science in China A，1995，38 (1)

12 Hu Chengjun，Wang Ji，Chen Huowang . Towards a formal and mechanical Verification of real-timesy systems. Proceedings of CICS'95，1995.

A Software Exploitation Method Based on Faling to User And Formalize Techneque

Wu Huisong

(North institute of oil exploration and design)

Abstrct In this psper，a strictly software exploitaion method was given. So that exploiter's Creation and the automation of exploited cirmstanle Can be fuu used，and software exploitation Can provide the need for user.

Keywords software exploitation method face to user using software once program transformation，system assay.