基于B方法的组件开发

高丽萍[1],褚 伟[2]

(1.合肥工业大学 计算机与信息学院,安徽 合肥 230009;
2.合肥工业大学 网络研究所,安徽 合肥 230009)

摘 要:现有的组件开发技术的规格说明是非形式化的,这导致了逻辑的非严密性和理解的歧义性,将会严重影响组件复用的效率。B方法是形式化方法之一,已经有功能强大的工具支持软件的形式化开发过程,它通过严格的数学推导和证明来保证软件设计和代码的正确性。为此,将B方法应用于学生信息管理系统的开发,提供了学生组件从需求规格说明、精化到最终实现的开发过程。通过对这一实例的研究可以看出,B方法增强了组件的规范性,对于提高组件复用的可靠性有重大的意义。

关键词:形式化方法;B方法;组件

中图分类号:TP311.52 **文献标识码**:A **文章编号**:1673-629X(2007)07-0028-03

An Approach of Component Development with B Method

GAO Li-ping[1], CHU Wei[2]

(1. School of Computer and Information, Hefei University of Technology, Hefei 230009, China;
2. Institute of Network, Hefei University of Technology, Hefei 230009, China)

Abstract: The fact that current specifications of component development are unformal results in logical non-rigor and comprehensive ambiguity, and it will affect the efficiency of components' reusing seriously. B method is one of formal methods, which supports the software development by the strong tool and guarantee the correctness of software's design and code via strict deduction and certification. For this reason, this paper applies B method to the development of student information management system. This method provides the developing procedure of student component that ranges from abstract specifications to the implementations. Through the research on this case, conclude that B method improves the formalization of specification and reliability of components' reusing.

Key word: formal method; B method; component

0 引 言

组件技术已经广泛应用于软件开发领域。这种技术以前所未有的方式提高软件产业的生产效率,这一点已逐步成为软件开发人员的共识。使用组件最大的好处就是可以将它们动态地插入或卸出应用系统,是真正意义上的软件模块即插即用。由于组件的开发过程不受组件用户的控制,如何开发可靠的组件成为组件开发的一个难题。当前较为流行的组件技术有Microsoft的COM,ORG的CORBAR,SUN的JavaBean以及IBM的OpenDoc等,这些技术成功地实现了对功能性接口规则的定义和规范,公布了对组件建模和应用的规格说明。但这种以自然语言描述的规格说明不可避免地存在缺乏逻辑的严密性、表达不清晰和二义性的缺点,导致最终开发出的组件的可靠性无法得到保证。形式化的软件开发是一种保证软件质量和软件可靠性的方法。B方法作为少数几种具有较强工具支持的形式化方法之一,在开发过程中它通过严格的数学推导和证明来保证程序的正确性,提供了规约环境的基础,使得其生成的模型比用传统方法或面向对象方法生成的模型更完整、一致和无二义性[1]。用B方法开发组件,将会为组件规格说明提供精确语义,规范组件的开发过程,保证系统的正确性和可靠性,以提高组件复用效率。在开发高安全的软件系统中使用形式化方法的重要性是被广泛认可的。形式化方法已被应用于一些案例研究,而且也已应用于一些关键工业中,包括铁路、健康、航天航空等部门。

收稿日期:2006-09-27

基金项目:国家自然科学基金项目(70471046)

作者简介:高丽萍(1982-),女,山西晋中人,硕士研究生,主要研究方向为计算机网络和软件形式化方法研究。

1 B方法介绍

B方法是一种对软件系统进行描述、设计和编码

的形式化方法[2]。B 方法产生于 20 世纪 80 年代早期,在 Z 语言的研究基础上引入了新的技术:谓词的转换和 Dijkstra 的最弱前置条件。它以一种基于 Zermelo - Frankle 集合论的符号表示法来书写,使程序规格说明处于一个统一的数学框架下,减少了出现语义错误的可能性[3,4]。B 方法较其他形式化方法的优点在于它支持整个软件开发的生命周期,首先从需求分析建模到规格说明的书写,然后对模型逐步求精,直到代码自动生成。

B 方法使用抽象机封装系统状态和行为,是一种基于对象的描述方式。抽象机的形式语义由 AMN 通过广义代换语言(GSL)提供。B 方法采用分层模块的方式构造大型的软件系统,其基本思想:抽象机的数学模型要经过多次精化,一个抽象机的精化要基于其他抽象机的规范,直到最后的精化结果(实现)能够在计算机上执行[2]。这样减少了大型系统开发分解到各个功能模块的复杂性,提供了重用以前性能良好的模块的可能性。

目前有 B - cored 的 B - ToolKit 和 Atelier - B 两种功能强大的工具,它们支持 B 方法开发软件的全部过程,包括 AMN、精化和实现的建立,语法分析,类型检查,大部分的正确性证明以及代码的自动生成,当前可生成的程序语言有 C,C + +等。

2　B 方法开发组件的可行性分析

组件是独立部属的单元,它跟所在的环境以及其他组件完全分离,因此组件必须封装自己的全部内部特征。AMN 中结构化的机制与面向对象方法很相似[3],以对象为基础,封装了各个变量和操作;信息隐藏原理使"用户"只能通过操作来改变机器的状态。B 方法的这些特征为组件开发提供了技术支持。

Bertrand Meryer 认为:没有契约的复用组件是困难的[1]。目前组件技术中契约的设计是比较困难的,而 B 方法解决了这个困难。在软件开发初期的规格说明中,B 方法形式化地描述组件契约内容,并进行形式化验证。而在每个开发阶段,每个设计描述都需要检测其与初始规格说明的一致性。

将 B 方法全部应用于软件各个部分是没必要的,而且实践的复杂性和困难度都会增加。因此采取了 B 方法与组件技术相结合的方法。一种结合途径就是 B 方法建立组件需求模型,在此基础上进行开发,弥补现在组件技术所欠缺的正确性规范。

3　应用实例

为了说明 B 方法的组件开发的步骤和方法,文中用一个具体的应用实例来解释说明。首先给出学生成绩查询系统中学生组件的规格说明。这里使用了 B - Core 的 BToolkit(工具集)对此组件建模,精化到最后的代码实现。

3.1　学生组件的规格说明

B 方法中,软件的抽象规格说明用抽象机(AMN)来描述。一个抽象机的形式化描述包含了若干个子句,MACHINE、SETS、VARIABLES、INVARIANT、CONSTANTS、PROPERTIES、DEFINITIONS 子句构成了系统变量的状态和属性等静态行为的描述。

学生组件的属性规格说明如下(其中封装了学生个人信息全部属性:学生姓名、学号、性别以及年龄):

```
MACHINE
    Student
SETS
    STUDENT;
    SEX = {male, female}
VARIABLES
    student, number, age, sex
INVARIANT
    student⊆STUDENT ∧
    number∈ student→NAT ∧
    age∈ student→NAT1 ∧
    sex∈ student→SEX
INITALIZATION
    student: = ø||
    sex: = ø||
    age: = ø||
    number: = ø||
```

MACHINE 表示组件名。SETS 子句定义两个集合:第一个延期集合 STUDENT,表示所有可能的(现在和未来的)学生;第二个枚举集合 SEX,表示学生的性别(male 或者 female)。VARIABLES 子句引进该机器的状态变量名,变量 student 是集合 STUDENT 的子集,它包含了那些已经将有关信息有效地存入数据库中的学生,number 表示学生的学号,age 对应学生的年龄,sex 代表学生的性别。INVARIANT 子句由若干合取项组成,在之后的开发过程中检查某个操作是否正确,我们必须证明相关的规范是否维持了这些不变式。

目前的组件技术通过插入一些断言描述接口的逻辑属性,包括前置、后置条件和不变式。这种契约方法只能保证操作的部分正确性[5]。为了获得操作的完全正确性,B 方法引入了 Dijkstra 的最弱前置条件方法,在 B 方法中描述为:[S]P,其中 P 是输出结果的逻辑状态,S 是可执行的操作,则[S]P 表示 S 执行结束后能够保证 P 的初始逻辑状态。

OPERATIONS 子句表示这个机器中的操作,描述

了系统的动态行为。每个操作被描述为一个前置条件和一个原子行为,前置条件表述一种必不可少的条件,在不满足它的情况下操作就不能执行,是激活操作的必要条件;原子行为通过一种建立在谓词转换器技术基础上的广义代换的方式来形式化表示[4]。

学生组件提供三个操作:添加学生记录操作 add(num,sx,ag),删除学生记录操作 delete(stu),修改学生记录 modify(stu)。下面给出了 add(num,stu)操作的完整规格说明,其他操作省略:

```
OPERATIONS
    add(num,sx,ag) =
        PRE
            STUDENT - student≠ø∧
            num∈NAT∧
            sx∈SEX∧
            ag∈NAT1
        THEN
            ANY newstu WHERE
              Newstu∈STUDENT - student
            THEN
              student: = student∪{ newstu } ||
              number(newstu): = num ||
              age(newstu): = ag ||
                sex(newstu): = sx
            END
        END;
    reportdelete(stu) = ……
    reportedit(stu) = ……
```

num、sx 和 ag 是形式输入参数,num 表示新增加学生的学号,sx 表示他的性别,ag 是他的年龄,我们为这个操作加上一个“实质性的”前条件:STUDENT - student≠ø∧num∈NAT∧sx∈SEX∧ag∈NAT1(NAT 表示从 0 开始的自然数,NAT1 表示从 1 开始的自然数),规定了输入参数的类型,只有满足这些条件的情况下才可以合理地激活与之对应的操作,即添加新学生的操作。操作执行最终结束后满足不变式。BToolkit 对上面的 AMN 进行语法分析和类型检查,并可以完成全部的证明义务。

3.2 精化和实现

精化是一种用于将软件系统的“抽象模型”(其规范)变换到另一种更具体一些的数学模型(实现)的技术[2]。主要的精化方法是对抽象机的操作进行变换和对系统功能进行细化。在上面抽象机的精化中,对 Student 组件分解,划分为多个功能模块。这些功能模块由部件库中基本功能抽象机(例如 basic_io,file_dump 等)经过聚合得到,并翻译成可执行代码。从初始模型到实现要进行多次精化,在每一次精化和最后的实现过程中,抽象机都要整体性地重新构造,并形式化验证其满足初始规范,保证设计和实现的正确性。这个过程这里不再给出。

3.3 组件接口的产生

组件最重要的特征是具有独立于应用的接口。每个组件通过接口提供服务并向其他构件请求服务,因此构件接口定义在组件开发中成为关注的焦点。组件可以直接提供接口,或者实现一个客户可访问的对象,而由该对象间接提供接口。这里由组件间接提供接口(即面向对象的接口)。

在 B-Toolkit 中,B-InterfaceGenerator 可以对已经分析过的 IMPLEMENTATION(实现)自动产生组件接口的规格说明和代码,成为系统和用户通信的桥梁。接口的规格说明如下:

```
INTERFACE
        Student
OPERATIONS
        add,
        delete,
        edit
END
```

接口中只保留了 IMPLEMENTATION 的 OPERATIONS 的行为名称,并没有具体实现,实现过程保留在 IMPLEMENTATION 中,这样做符合我们对接口的常规认识。

组件接口的特性和操作需要服从使用方面的规格说明,使组件间的通信成为可能。该规格说明即要表明客户为使用接口需要做些什么,又要指出提供者为满足接口承诺的服务需要实现些什么[5]。这些规格说明应当尽可能地形式化,以从中获取必要的信息和进行形式化验证,避免接口规格说明的歧义性和模糊性。接口的规格说明已在相对应的操作规格说明中得到描述。

4 结束语

文中提出了一种组件技术和形式化方法 B 的结合方式。用 B 方法为组件建模,从中开发组件,充分利用了形式化的优点,保证软件系统的正确性,大大提高了软件复用的可靠性和无二义性。形式化方法具有精确性和可验证性的优点,随着软件规模和复杂度的增加,形式化方法将被越来越广泛地应用于软件技术中。

参考文献:

[1] Pressman R S. 软件工程:实践者的研究方法[M]. 第 5 版. 梅 宏 译. 北京:机械工业出版社,2002.

(下转第 34 页)

oDetailsNode. Text = name

即是在 List. xml 文件中创建一个节点＜name＞，并将变量 name 接收的值赋给标签＜name＞＜/name＞中的数据内容。

3) List. xml 负责存储(输出)由用户界面输入(调用)的数据，其代码如下：

```
<? xml version="1.0" encoding="gb2312"? >
<xml><Knowledgebase>
<item>
<id>1</id>
<name>感冒</name>
<Posttime>2004 - 5 - 14 10:28:05</Posttime>
<causeofill>受凉</causeofill>
<disease>头疼，发热</disease>
<medicine>感冒灵，一天三次，一次一粒。</medicine>
<remarks>注意休息，祝早日康复！</remarks>
</item>
<item>
<id>2</id>
.....
</item>
</Knowledgebase></xml>
```

代码说明：标签＜Knowledgebase＞，＜id＞，＜name＞，＜causeofill＞，＜disease＞，＜medicine＞，＜remarks＞等都是由用户定义的，从标签的命名上直观地反映了标签包含的数据内容的具体含义，这正是 XML 的优点之一，譬如：＜id＞＜/id＞中的数据内容是表示这是第几条记录，＜name＞＜/name＞中的数据内容是表示这条记录的名称是知识库中哪个部分的名称，＜causeofill＞，＜/causeofill＞，＜disease＞，＜/disease＞，＜medicine＞，＜/medicine＞，＜remarks＞，＜/remarks＞中的内容表示的是记录具体内容。用户可以通过交互界面的查询功能搜索知识库中的知识规则，如果能找到，即匹配查找成功，那么输出这条记录，如果由多条记录同时满足，则列表给出或给出一个相似度最高的数据，如果未能找到，即匹配查找不成功，则显示没有这项记录，并询问用户是否要添加所查记录[7]。例如，用户输入“发热 头疼”，经过查找匹配，系统可以找到相应的记录，则输出：

“病因：受凉

病症：发热，头疼

用药：感冒灵，一天三次，一次一粒

备注：注意休息，祝早日康复!!”

4) Del. asp 是用户删除已经添加到知识库中的记录的交互界面，其代码如下：

```
Set objRootsite = objXML. documentElement. selectsinglNode ("Knowledgebase")
  AllNodesNum = objRootsite. childNodes. length - 1
  response. write AllNodesNum
  objXML. DocumentElement. RemoveChild(objRootsite)
  objRootsite. parentnode. removechild(objRootsite)
  response. write "ok"
objXML. save(strSourceFile)
```

4 结束语

XML 在实际应用中的重要作用越来越明显，作为一种数据交换的标准，它已经逐渐地流行起来[1]。它允许各个领域自定义自己行业内部的通用标记，并且易于阅读和编写，为网络上交换数据提供了很好的标准。

如前面所说，一个完整的专家系统主要包括知识库、数据库、推理机制、解释机制、人机接口和知识获取等功能模块。文中着重探讨了如何用 XML 建造专家系统的知识库以及其相关操作。

参考文献：

[1] 黄 理，曹林有，张 勇，等. ASP. NET/XML 深入编程技术[M]. 北京：北京希望电子出版社，2002.

[2] 程慧霞，李龙澍，倪志伟，等. 用 C + + 建造专家系统[M]. 北京：电子工业出版社，1996.

[3] 张全寿，周建峰. 专家系统建造原理及方法[M]. 北京：中国铁道出版社，1992.

[4] Ennsen L. WEBSPHERE 环境下 XML 与 XSL 编程[M]. 北京：机械工业出版社，2001.

[5] Baartse M. ASP 与 XML 高级编程[M]. 康 博译. 北京：清华大学出版社，2002.

[6] Freeman A, Jones A. Microsoft. NET XML Web 服务程序设计[M]. 北京：清华大学出版社，2003.

[7] 胡海璐. XML Web Services 高级编程范例[M]. 北京：电子工业出版社，2003.

(上接第 30 页)

[2] Abrial J R. B 方法[M]. 裘宗燕译. 北京：电子工业出版社，2004.

[3] 邹盛荣，郑国梁. B 语言和方法与 Z、VDM 的比较[J]. 计算机科学，2002(10)：136 - 138.

[4] 肖美华，薛锦云. 形式化方法 B 及其程序规约机理[J]. 计算机工程 2004，30(16)：16 - 18.

[5] Szyperski C，Gruntz D，Murer S. 构件化软件：超越面向对象编程[M]. 第 2 版 . 王千祥，曹东刚，左继宏 译. 北京：电子工业出版社，2004.