

一种面向柔性的业务对象模型研究 *

申利民, 王显峰, 蔡萃燕

(燕山大学 信息科学与工程学院, 河北 秦皇岛 066004)

摘 要: 针对目前信息系统中面向用户层次柔性程度不高的现状, 分析了信息系统面向用户的柔性, 将信息系统的柔性分为业务控制柔性、业务功能柔性、业务过程柔性和业务组织柔性。提出了利用适应性对象模型支持上述柔性的思想, 进而提出了柔性业务对象的概念, 建立了柔性业务对象模型, 给出了该模型的形式化描述, 并分析了其对柔性机制的支持。最后给出了其实现流程并用实例说明了该模型的实用性和有效性。

关键词: 信息系统; 面向用户的柔性; 柔性业务对象模型; 适应性对象模型

中图分类号: TP311

文献标志码: A

文章编号: 1001-3695(2009)03-0908-03

Research on flexibility oriented business object model

SHEN Li-min, WANG Xian-feng, CAI Cui-yan

(College of Information Science & Engineering, Yanshan University, Qinhuangdao Hebei 066004, China)

Abstract: Aiming at the problem that the end-user flexibility of information systems was too low, firstly, analyzed the user level flexibility of information system, which was divided into business control flexibility, function flexibility, process flexibility and organization flexibility. Then, proposed the idea of support the above flexibility based on adaptive object model. Moreover, defined the conception of flexible business object and proposed flexible business object model, which was formalizing described and the support for flexibility was discussed. Finally, gave its implementation process, and provided a modeling example to verify the practicability and effectiveness of the proposed model.

Key words: information system; user-level flexibility; flexible business object model; adaptive object model

随着社会信息化水平的提高, 信息系统已经渗透到社会的各个领域, 人们对信息系统的要求也越来越高, 已经由原来的被动接受、使用转变成现在的主动发现和个性化定制。这就要求信息系统能够即时满足用户的各种需求变化和多样性的要求。

柔性软件^[1,2]是指一定范围内能够满足和适应用户需求变化的软件系统, 它使用柔性来描述软件适应变化的能力。根据用户对软件操控能力的不同, 柔性软件在三个层次上体现其柔性: a) 面向低级用户的柔性。它允许通过软件提供的简单操作来调整软件的功能(如设定功能参数、自定义查询等), 所进行的操作不需要任何程序设计的专业知识。b) 面向高级用户的柔性。它允许通过软件提供的设置和操作界面来完成一些复杂关键的调整操作(如初始化、业务规则定义、业务过程定义等), 实现该层次的柔性需要较高的计算机知识和领域知识。c) 面向开发级用户的柔性。它允许对软件进行深层次的调整, 增添软件的内置行为和改变软件的构成(如构件替换、调整对象等), 实现该层次柔性需要用户具有较强的业务知识和程序设计知识。

目前的信息系统开发大部分是采用传统的面向对象技术以一种相对静态的方式对信息系统进行建模, 它在设计阶段用类来描述对象模型, 在实现阶段由程序员编码实现。这样建立的业务模型是相对静态的, 大大限制了软件的柔性, 当业务实体的属性、行为发生变化或有新的业务实体出现时, 都会引起代码的重新编写、编译, 甚至版本更新。这意味着业务模型发生改变时, 信息系统不能很好地适应用户要求, 从而使人的主

动性也无法得到很好的满足。因此, 必须开发一种能够由用户驱动来动态适应需求变化的柔性信息系统。

1 面向用户的柔性分析

信息系统必须能够反映企业或组织的业务过程和业务规则。而随着企业和组织自身的不断发展, 业务过程、业务规则和业务实体等的变化是不可避免的。即使这些业务过程和业务规则的核心没有发生变化, 对信息系统的不同用户而言, 对它们的理解和要求也会发生变化。因此如何提高用户层次的柔性, 尤其是面向高级用户和开发级用户的柔性是信息系统成功与否的关键所在。

典型的企业信息系统模型一般包括业务控制模型、业务功能模型、业务过程模型、业务组织模型。与之相对应, 信息系统的柔性(图1)可以分为业务控制柔性、业务过程柔性、业务功能柔性、业务组织柔性。



图1 信息系统的柔性

1) 业务控制柔性(business control flexibility, BCF) 它表现为在一定范围内对业务基本过程的调整。当系统的应用目标发生变化时, 可以由用户通过对业务功能进行重新选择或对业务功能间的流向进行调整以满足用户的需要。

2) 业务功能柔性(business function flexibility, BFF) 它表现为当系统中的某个业务功能不能满足新的需求时, 可以由用

收稿日期: 2008-05-16; 修回日期: 2008-07-07 基金项目: 河北省自然科学基金资助项目(F2008000861)

作者简介: 申利民(1962-), 男, 黑龙江人, 教授, 博导, 主要研究方向为柔性软件工程(shenlmm@sina.com); 王显峰(1983-), 男, 河北人, 硕士, 主要研究方向为柔性软件工程、企业信息系统; 蔡萃燕(1982-), 女, 山东人, 硕士, 主要研究方向为数据库加密技术。

户通过简单的配置来调整业务功能和业务功能的表现形式。

3)业务过程柔性(business process flexibility, BPF) 它表现为当利用 BFF 无法满足用户对业务功能的要求时,通过用户对完成业务功能的业务过程进行配置来满足用户的需求。

4)业务组织柔性(business organization flexibility, BOF) 它表现为当企业或组织的业务组织结构发生变化(部门合并、从属关系、角色变更、约束条件变化)时,用户可以通过对组织模型的配置满足用户的需求。

2 适应性对象模型的核心思想

所有有价值的软件都在一定程度上反映了现实世界,其数据结构 and 代码无不蕴涵了该软件所涉及领域的相关知识。传统的面向对象程序设计方法直接使用类对业务进行建模,将大部分的业务知识封装到了代码中,而这些封装到代码中的知识在运行时是无法改变的,当需求的动态性导致领域知识需要发生变化时,必然要求程序员修改代码,这往往会引入不可预料 的错误。另外,一些具有动态性要求的需求^[3]也不允许通过修改代码的方式来满足需求。如果将这些领域知识以元数据的形式存储,将领域知识与程序代码分离,由计算机维护这些领域知识,就能够在运行时即时满足用户需求的变化,适应性对象模型^[4,5]就是在这一思想的基础上提出的。

适应性对象模型将类的属性、行为和关系存储为元数据^[6],在运行时解释执行这些元数据并建立对象模型。当对象模型需要发生变化时,只需要改变相应对象模型的元数据描述,就可以在运行时改变系统的行为,这就意味着用户拥有了更强的创建、定制、扩展系统功能的能力^[7]。将对象的知识保存在元数据中由计算机来维护,使得基于适应性对象建模的系统能有效地应对需求的动态性问题,同时降低软件的实施成本。笔者将适应性对象模型的思想与业务领域中的概念相结合,使之能较好地支持信息系统面向高级用户和开发级用户的柔性。

3 面向柔性的业务对象模型

3.1 柔性业务对象

柔性业务对象是一种描述特定领域内业务概念的语义对象,它以一种高层次的观点来表达业务领域的核心知识,它与具体的计算设施和技术实现细节无关,允许在运行时由高级用户(领域专家)动态地进行调整,调整的过程是可控的、可逆的、和谐的和容易的。业务对象的柔性调整具有以下特征:

- a)调整是可控的,意味着调整被限制在一定范围之内。
- b)调整是可逆的,意味着调整能以几乎相同的代价恢复。
- c)调整是和谐的,意味着调整不会导致失效或质量及可靠性的降低。
- d)调整是容易的,意味着调整的过程需要较少的时间和费用。

例如银行业的账户、电子商务中的订单等都是业务领域的概念,都可以采用柔性业务对象来描述。而数据存取、异构平台通信等涉及的对象往往不是柔性业务对象的描述范围,因为它们不是业务处理和业务过程的核心内容。由于柔性业务对象具有以上所述的特征,与普通的业务对象相比,它能更容易与业务概念保持一致性,更能深刻地体现业务的本质。

3.2 柔性业务对象模型

柔性业务对象模型(图 2)描述了特定领域内业务对象的构成,以及这些业务对象之间关系的对象模型。它描述了业务用例的实现,是对业务角色和业务实体之间如何联系及协作以执行业务的一种抽象,允许在运行时由高级用户(领域专家)对其进行动态调整以适应用户需求的变化。

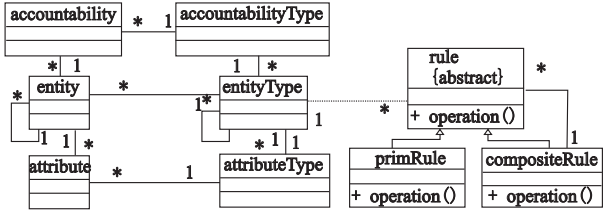


图2 柔性业务对象模型的基础结构

柔性业务对象模型是一个自描述的对象模型,它分为核心层、扩展层和用户接口层三个层次(图 3)。其中,核心层体现了领域的核心业务和功能,是比较稳定的。扩展层是对领域变化部分的封装,并提供了改造、变更和扩展的接口,这使得业务对象模型处于开放状态。用户接口层是对核心层和扩展层面面向用户的描述,该层屏蔽了模型的具体实现,使领域专家能方便地对其进行调整。

定义 1 柔性业务对象模型为一个六元组,FBOM = {BS, ES, TS, CS, VS, AS}。其中:BS 代表业务对象的基础信息集合;ES 代表业务对象的稳定信息集合;TS 代表业务对象的易变信息集合;CS 代表业务应用的上下文集合;VS 代表业务对象的视图集合;AS 代表业务对象的适配操作集合。

定义 2 基础信息集合为一个四元组,BS = {ID, ON, UN, UD}。它描述了与业务对象相关的基本属性。其中:ID 为业务对象的惟一标志;ON 表示业务对象的名称,该名称是高级用户或领域专家所熟悉的;UN 为创建该业务对象的用户名称;UD 反映了对业务对象最近修改的时间。

定义 3 稳定信息集合为一个三元组,ES = {SA, SH, SR}。它是对特定领域的核心业务和功能的描述,这些信息是在领域分析后得到的,是领域知识的集中体现,相对比较稳定。其中:SA 为特定领域内业务对象所共有的属性集合;SH 为特定领域内业务对象共有的行为集合;SR 为特定领域内业务对象之间的共有关系集合。

定义 4 易变信息集合为一个三元组,TS = {DA, DH, DR}。它是对特定领域业务的不同理解和对相同理解的不同表现形式的描述,属于可选信息,在软件运行过程中很可能需要根据用户的需求进行调整。其中:DA 表示业务对象可能变化的属性集合;DH 表示业务对象可能变化的行为集合;DR 是对业务对象之间可能变化的导向关系、约束条件等的描述。

定义 5 业务应用上下文集合 CS = {c_i | i ∈ N}。它描述了业务对象应用的具体环境和适用场景,方便了用户对业务对象的选择。其中:c_i 是对第 i 个业务对象应用场景的描述信息;N 为自然数集。

定义 6 视图集合 VS = {⟨a_i, v_i⟩ | i ∈ N, a_i ∈ SA ∪ DA}。描述了业务对象的表现形式,业务对象通过视图与业务角色相关联,一个业务对象可以有多个视图,不同的业务角色通过相应的视图对业务对象进行操作(图 4)。根据业务对象的构成和表现要求,可采用相关的呈现模式^[8]来实现。视图简化了对领域概念的表示,是业务对象与应用间的接口。其中:⟨a_i, v_i⟩

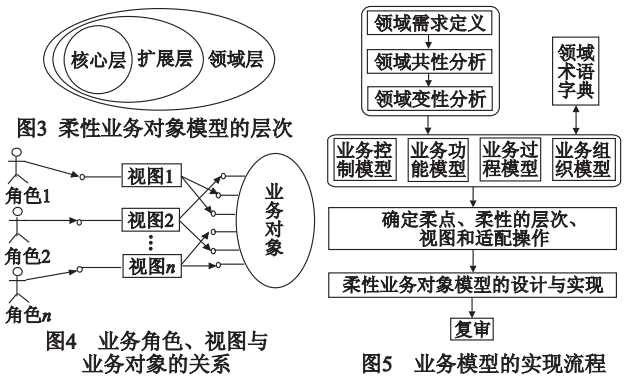
表示业务对象特定的属性 a_i 对应的视图 v_i ; N 为自然数集。

定义 7 适配操作集合 AS 是对系统中现有的业务对象进行配置和利用现有业务对象生成新的业务对象的过程,它的目的是满足新的需求和应用上下文。适配操作主要是对业务对象的属性、行为、视图和业务对象之间关系的配置。

根据软件柔性的度量要素^[1],如果采用传统的面向对象技术以类对业务对象进行建模,就不能在不重新编码的情况下对业务对象的结构进行调整。业务对象是相对僵化的,可操控的柔点较少。如果采用柔性业务对象模型对业务进行建模,业务对象的属性、行为和业务对象之间的关系均可以成为柔点,设属性处的柔点为 F_a ,行为处的柔点为 F_b ,关系处的柔点为 F_r 。由柔度的计算公式 $K_i = S_{\max}(i) / (1 + F_{\min}(i))$,参考柔力柔距的参数设定表,随着业务实体的增多,软件的柔量 $A = \sum_{i=0}^N K_i$ 会成倍提高。由此可见,柔性业务对象模型非常适合于柔性软件的开发。

3.3 柔性对象业务模型的实现流程

柔性业务对象根据其适用范围的不同分为领域通用柔性业务对象、领域专用柔性业务对象两类。领域通用柔性业务对象来源于特定领域的共性需求,领域专用柔性业务对象是特定领域的个性化需要。根据面向对象的领域分析方法^[9],柔性业务对象模型的实现(图 5)分为以下几个步骤:



1) 领域需求定义 通过系统分析人员与特定领域的领域专家的反复交流和对现有系统需求定义的分析,明确该领域的业务需求并为每一种业务能力初步定义相应的术语。通过运用领域术语进行表达和交流,相关人员加深对领域知识的理解,确定领域的范围和边界,同时收集领域的相关信息。

2) 领域共性分析 在前一阶段对领域理解的基础上,对特定领域的业务进行抽象和精化,确定领域的共同需求。在识别共同需求的过程中,应充分参照标准规范,以提高领域工程产品在未来新系统开发中被复用的可能,同时应完善和调整领域术语字典以精确反映领域知识。

3) 领域变性分析 识别具有变化性的需求,确定可变需求的类型(可选的还是多选一的)。在识别变化性需求时,不仅要反映当前领域可能的变化,还应尽可能地预测未来领域可能要求的变化。

4) 建立面向对象的分析模型 在现有的业务对象模型基础上,利用共性变性分析的结果,对业务对象模型进行补充和修改,采用类图、用例图等形式表达特定领域的业务控制模型、功能模型、业务过程模型和业务组织模型。

5) 设计实现柔性业务对象模型 根据前几个阶段的分析结果,参考柔性业务对象模型的形式化定义,确定业务模型中

的柔点、柔性层次、业务对象的视图、业务模型的适配操作,设计实现系统的业务模型。

6) 复审 由系统分析人员和领域专家对实现的业务对象模型进行分析和评估,如果没有通过评估,则迭代以上步骤。

4 应用实例

ERP 项目风险管理系统用来对 ERP 实施过程中牵涉的各种潜在的不确定因素进行识别并制定相应的防范和处理策略。项目中形式多变的不确定性因素会导致新类型风险的出现,同时特定类型风险的属性和应对策略也可能会有所变化。

一般的风险管理系统仅能提供数量有限的风险类型,特定风险类型的相关属性和应对策略也是预先确定的,用户无法在系统部署以后对其作出改变。一旦需要增加新的风险类型或对特定的风险类型进行调整,只能依赖开发人员通过编程来适应需求的变化。

因此,ERP 项目风险管理系统必须解决以下几个关键性问题:a) 允许增加新的风险类型以满足各种可能出现的风险因素;b) 允许对特定风险类型的属性进行调整以满足特定风险可能发生的变化;c) 允许对特定类型风险的应对策略进行定制以适应不同的情况带来的差异。

为了解决以上问题,可以采用柔性业务对象模型对业务功能进行建模(图 6),高级用户可以使用软件操控平台提供的图形界面面对项目风险的属性、风险的应对策略进行动态配置,同时可以创建新的风险类型来应对风险不确定性带来的需求变化。

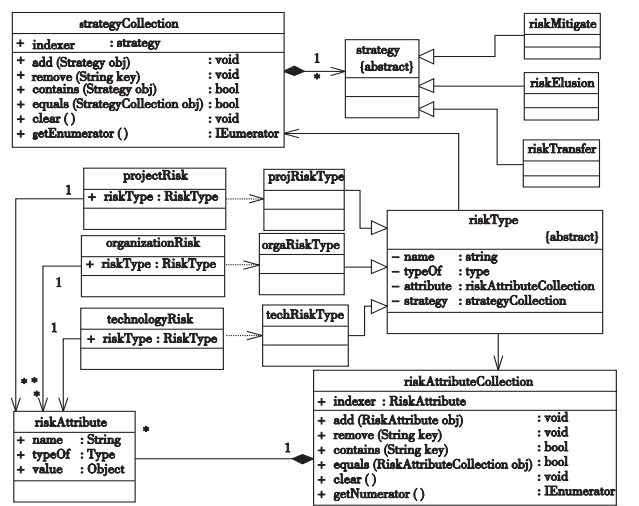


图6 ERP项目风险管理系统功能模型

5 结束语

企业外部环境和内在机制的剧烈变更要求软件系统必须应对各种形式的变化,因此如何提高柔性,尤其是面向高级用户和开发级用户的柔性一直都是信息系统面临的一个主要问题。在对信息系统柔性的分析和总结适应性对象模型思想的基础上,提出了采用柔性业务对象模型对信息系统进行建模,将业务对象模型的信息存储在元数据中,通过操控平台改变业务对象模型能够比较容易地应对各种需求的变化,为企业复杂信息系统的设计提供了一条新的思路。目前,上述思想已成功运用到笔者开发的 ERP 项目风险管理系统中,并取得了良好的效果。进一步的研究方向有基于柔性业务对象模型系统的一致性问题、支撑工具及版本管理等一系列问题。(下转第 916 页)

5 剖面描述形式化方法

对一个剖面描述方案,本文将其中的剖面、子剖面分别映射为树中对应的父节点、子节点,对采用某个剖面描述方案的构件,可以将其相应的剖面描述术语映射为对应的叶子节点。例如:Prieto-Diaz 最早提出来的剖面描述方案为两个主剖面,即功能和环境,且每个主剖面分别有三个子剖面(作用、对象、媒介)和(应用领域、系统类型、客户类型)。在该剖面描述方案下描述的某个构件可以用如图 3 所示的剖面树来表示。对于构件的查询也可以相应地表示为一棵查询树,即将查询中出现的剖面名、子剖面名转换为相应层次的父节点和子节点,将查询的剖面术语值转换为叶节点,并用一个虚拟的根节点将它们组合为一棵查询树。

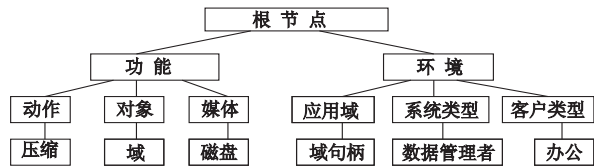


图3 一个构件描述的剖面树

6 结束语

从文献[11,12]中可以推出构件到后面的匹配必先找到该子树,再在该子树中找到属性匹配。笔者提出一种构件匹配的解决方案,对树查找到该子树的算法优化会更好。用决策树来匹配决策子树,将不同于剖面检索。本文提出一个树型的形式化方案,用子节点来构成父节点,例子分析如下:

```
Window; = me|98|2000|xp
Linux; = redhat|bluepoint
Serveropsystem; = window|linux
Serveraddress; = ip|port
Server; = Serveraddress|Serveropsystem
```

构件方面的工作有文献[11~14]提出的在 XML 构件上查找,而本文提出一种基于机器学习的解决方案。当然,笔者的工作目的不一样,但就 XML 构件上查找是一样的。笔者用实例学习和分析学习结合的机制来解读构件,认为这是一种新的方法,有可能解决这类问题。笔者将文献[11~14]提出的检索称为静态检索;本文提出的称为动态检索,在机器学习中称为假设空间搜索。本文很少提到策略,是因为策略在 XML 文件中。

参考文献:

[1] IVAR J. Software reuse: architecture, process and organization for business success[M]. Reading: Addison-Wesley Publishing Company, 1997:4-15.

[2] MILI H, MILI A. Reuse based software engineering[M]. New York: Wiley, 2002:444-459.

[3] SCHANTZ R, LOYALL J, ATIGHETCHI M, et al. Packaging quality of service control behaviors for reuse[C]//Proc of the 5th IEEE International Symposium on Object-oriented Real-time Distributed Computing(ISORC). Washington DC: IEEE Computer Society, 2002: 375.

[4] MITCHELL T. Machine learning[M]. 曾华军, 张银奎, 等译. 北京: 机械工业出版社, 2003:165-169.

[5] PODGURSKI A, PIERCE L. Retrieving reusable software by sampling behavior[J]. ACM Trans on Software Engineering and Methodology, 1993, 2(3):286-303.

[6] ZAREMSKI A M. Signature and specification matching[D]. Pittsburgh: School of Computer Science, Carnegie Mellon University, 1996.

[7] MERKL D, TJPOA A M, KAPPEL G. Learning the semantic similarity of reusable software components[C]//FRAKES W B. Proc of the 3rd International Conference on Software Reuse(ICSR'94). Rio de Janeiro: IEEE Computer Society Press, 1994:33-41.

[8] DAMIANI E, FUGINI M G, BETTETTINI C. A hierarchy aware approach to faceted classification of objected-oriented components[J]. ACM Trans on Software Engineering and Methodology, 1999, 8(3):215-262.

[9] HENNINGER S. Supporting the process of satisfying information needs with reusable software libraries: an empirical study[C]//SAMADZADEH M H, MANSOUR K Z. Proc of the 17th International Conference on Software Engineering and Software Reusability. Seattle: ACM Press, 1995:267-270.

[10] 边肇祺, 张学工. 模式识别[M]. 北京: 清华大学出版社, 2002: 136-159.

[11] 王渊峰, 张涌, 任洪敏, 等. 基于剖面描述的构件检索[J]. 软件学报, 2002, 13(8):1546-1551.

[12] 徐如志, 钱乐秋, 程建平, 等. 基于 XML 的软件构件查询匹配算法研究[J]. 软件学报, 2003, 14(7):1195-1202.

[13] 姚全球, 孟丽, 崔杜武. 基于 CBR 和 XML 的软构件检索方法[J]. 计算机应用, 2007, 27(7):1711-1714.

[14] 孙念民, 李万龙, 郑山红, 等. 基于 XML 技术的软件构件表示与检索[J]. 计算机系统应用, 2007, 17(10):94-97.

[15] W3C. Extensible markup language(XML)[EB/OL]. (2008-07-06) [2008-07-13]. <http://www.w3.org/XML/>.

[16] 常继传, 李克勤, 郭立峰, 等. 青岛系统中可复用软件构件的表示与查询[J]. 电子学报, 2000, 28(8):20-24.

[17] 李晓博, 缪淮扣, 刘静. 基于形式规格说明的构件匹配[J]. 计算机应用与软件, 2006, 23(10):10-12.

(上接第 910 页)

参考文献:

[1] 申利民, 高春艳, 李峰. 软件功能柔性定量分析[J]. 小型微型计算机系统, 2008, 29(1):61-65.

[2] 李安波, 黄家柱, 毕硕本, 等. 柔性软件系统研究与应用[J]. 计算机应用研究, 2005, 22(1):140-143.

[3] 李长云, 何频捷, 李玉龙. 软件动态演化技术[M]. 北京: 北京大学出版社, 2007:39-42.

[4] YODER J W, BALAGUER F, JOHNSON R. Architecture and design of adaptive object-models[C]//Proc of the Technology Presentation at the 2001 Conference on Object Oriented Programming Systems, Language and Applications(OOPSLA'01). 2001:50-60.

[5] YODER J W, JOHNSON R E. The adaptive object-model architecture

ral style[C]//Proc of the 17th IFIP World Computer Congress TC2 Stream /3rd IEEE/IFIP Conference on Software Architecture: System Design, Development and Maintenance. 2002:3-27.

[6] 欧金梁, 金茂忠. 利用元数据建模构建可扩展的灵活系统[J]. 计算机科学, 2005, 32(7):214-217.

[7] 姚海琼, 倪桂强. 自适应对象模型体系结构研究[J]. 计算机应用研究, 2004, 21(9):121-124.

[8] WELICKI L, YODER J W, BROCK R W. A pattern language for adaptive object models[C]//Proc of IEEE International Conference on Object-Oriented Programming, Systems, Languages, and Applications 2007(PLoP '07). 2007:787-788.

[9] 王千祥, 吴琼, 李克勤, 等. 一种面向对象的领域工程方法[J]. 软件学报, 2002, 13(10):1977-1984.