

# 基于功能重用性的候选服务选定方法

昌广才, 蒋 溢, 熊安萍

(重庆邮电大学 计算机科学与技术学院, 重庆 400065)

**摘 要:** 基于 SOA 的服务建模在企业中得到了广泛运用, 针对 SOMA 服务建模过程中基于用例间包含与扩展关系对可重用候选服务进行选定时存在功能重用性用例确定不准确的问题, 给出了加入用例泛化关系的可重用候选服务选定方法, 完善了可重用性候选服务的选定。实践表明, 改进后的候选服务选定方法不仅使简化了可重用性候选服务选定过程, 还提高了可重用性候选服务选定的准确率。

**关键词:** 面向服务架构 (SOA); 服务建模; 候选服务; 可重用性; 用例关系

**中图法分类号:** TP311.5 **文献标识号:** A **文章编号:** 1000-7024 (2012) 12-4571-05

## Selecting candidate service based on functional reusability

CHANG Guang-cai, JIANG Yi, XIONG An-ping

(School of Computer Science and Technology, Chongqing University of Posts and Telecommunications,  
Chongqing 400065, China)

**Abstract:** SOA-based service modeling is widely used in the enterprise applications. During SOMA service is modeling, according to the identification inaccuracy of function reusability between use case when selecting reusable candidate services, based on the expansion and containment relationship, the selection process of reusable candidate services are provided, which join generalization relationship between the use case, and the selection of reusability candidate services is improved. Practice shows that the improved candidate services selection methods not only simplify reusable candidate selection process, but also improve the selection accuracy of reusability candidate services.

**Key words:** SOA; service modeling; candidate services; reusability; case relationship

## 0 引 言

当前软件技术的快速发展, 面向服务体系架构 (service-oriented architecture, SOA) 成为继中间件技术之后应用程序架构设计的新的道路。在以服务为核心的 IT 架构中, 业务模型将不再需要对应到某个应用程序上, 而是将每个相对独立的业务需求作为“服务”来实现<sup>[1]</sup>。

通过服务建模将业务需求转换成基于服务的解决方案中, 服务建模是实现 SOA 对业务随需应变灵活性支持的关键因素。在建模过程中选定合理的服务对于系统的开发和系统的集成起着重要的作用, 目前在服务建模中有许多关于候选服务选定的指导方法。

针对于 SOMA 服务建模中候选服务选定原则多样 (如: 可组装、可重用、业务对齐等)、选定方法使用难度

大、过程复杂等一系列问题, 本文通过对 SOMA 服务建模中可重用候选服务包含的用例间关系进行分析, 结合面向对象模型和功能点估算法对可重用候选服务进行选定, 使得对重用性候选服务选定过程更加简洁和实用, 并能准确的对候选服务进行选定。

## 1 SOMA 及可重用候选服务选定

SOMA 建模方法是由 IBM 提出的一种面向服务建模方法, 该方法把面向服务分析与设计分成以下三个部分: 发现服务 (候选服务)、规约服务、实现服务<sup>[2]</sup>。该建模方法是将前期需求分析业务组件作为建模的输入, 运用该方法建立候选服务列表, 然后对选定的服务进行规范化描述、最后通过具体的编程技术实现服务, 其建模过程如图 1 所示。

收稿日期: 2012-02-17; 修订日期: 2012-04-20

基金项目: 重庆市信息产业发展专项基金项目 (200811004)

作者简介: 昌广才 (1987-), 男, 重庆人, 硕士研究生, 研究方向为面向服务计算、软件工程等; 蒋溢 (1969-), 男, 湖北安陆人, 高级工程师, CCF 会员, 研究方向为面向服务计算、数据挖掘、软件工程等; 熊安萍 (1970-), 女, 四川泸县人, 副教授, CCF 会员, 研究方向为高性能计算、面向服务计算、软件体系结构等。E-mail: 471072700@qq.com

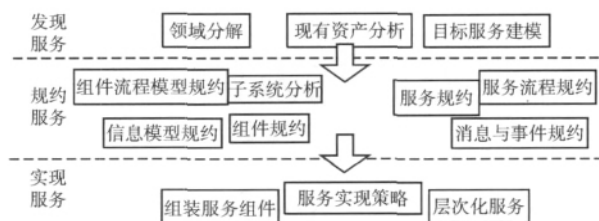


图1 SOMA 建模过程

### 1.1 建模过程

(1) 发现服务：发现服务的主要任务是确定在一定企业范围、业务流程范围、功能区域等可能成为候选服务列表。其包括领域分解、现有资产分析、目标服务建模<sup>[3]</sup>。

(2) 规约服务：规约服务的任务是对服务的各属性、服务间的依赖关系进行规范的描述。

(3) 实现服务：服务实现的实质是对服务的实现进行决策，将服务分配到相应的服务组件中，然后逐个分析服务实现方式并进行技术可行性的验证<sup>[4]</sup>。

### 1.2 可重用候选服务选定分析

候选服务指通过建模过程中发现服务所得到的结果，是系统所需的服务集合。但是在实际的情况下并不是所有的候选服务都能成为系统所需要的服务，只有合理的服务才能有助于系统的开发，所以必须通过一定的选定方法对候选服务进行选定。SOMA 建模过程中服务选定的基本原则通常有可重用性、可组装性、业务对齐等，其中可重用性是候选服务选定最重要的原则之一，通过对可重用候选服务的选定可以提高服务建模的效率，减少系统开发中对某重用候选服务功能的开发。

目前在服务建模中对于可重用候选服务选定，文献<sup>[5]</sup>从服务建模的基础理论方法中去讨论候选服务的选定，深入分析了候选服务选定的基本原则和指导思想，但是具体给出选定方法及过程；文献<sup>[6]</sup>通过利用基于语义对候选服务进行选定，该方法通过对候选服务进行形式化语言描述，分析候选服务之间语义关系进行选定，在实际服务建模过程中该方法加大了选定候选服务的工作量和复杂度；文献<sup>[7]</sup>基于面向对象模型对重用性候选服务进行选定，该方法从用户角度去对重用性候选服务选定，结合面向对象思想使得选定过程简洁易操作，但是该方法没用深入使用面向对象模型去选定重用性候选服务，选定结果不够完整。还有文献<sup>[8]</sup>基于服务粒度对候选服务进行选定，服务粒度大小直接影响着服务的重用性，通过对候选服务的粒度度量确定候选服务是否可以被重用并进行选定。该方法适用于选定细粒度的候选服务，对于具有粗粒度的复杂候选服务选定效率不是很高。

## 2 基于功能重用性候选服务选定方法

在服务建模中，一个候选服务即是一个功能系统，它

包含完成某项业务的单个或多个功能。在候选服务可重用性选定基本原则下，通过完整分析功能系统中用例间的包含关系、扩展关系、泛化关系，合理选取可重用性功能用例，并运用 COSMIC-FPP 功能点规模度量方法分别计算出功能系统总的功能规模数及可重用部分功能规模数，再根据可重用规模数所占比值对候选服务进行选定。

### 2.1 用例间关系

功能系统中包含着许多完成业务的功能用例，功能用例由基本用例、次用例、扩展用例等组成。用例间关系如图2所示，包括有包含关系(include)、扩展关系(extend)、泛化关系(Generalization)。

(1) 包含关系(include)：用例模型中1个用例可以包含其他用例具有的基本功能和行为，并将包含的用例功能和行为作为本用例的部分内容，称这种关系为包含关系。由基本用例管理这种包含关系，并且被包含的用例的事件流是否被插入到基本用例的事件流中也是由基本用例控制，基本用例与包含用例所执行的结果具有一定的依赖关系，但是双方都不能访问和修改对方的属性。

(2) 扩展关系(extend)：在用例模型中1个用例可以被定义为另一个用例的增量扩展，这称作扩展关系，扩展关系相当于是将新的行为插入到现有的用例中的方法中。基本用例提供了一组扩展点，可以在新增的扩展点中添加新功能和行为，而扩展用例提供了一部分插入功能和行为片段，这些功能和行为片段能够被插入到基本用例的扩展点上。

(3) 泛化关系(generalization)：在用例模型中1个子用例与1个父用例相似，但是在功能和行为中表现出比父用例更特别的行为特征；子用例将继承父用例的所有结构、行为属性和关系等。子用例可以使用父用例的一段功能和行为，同时它也可以重载它。

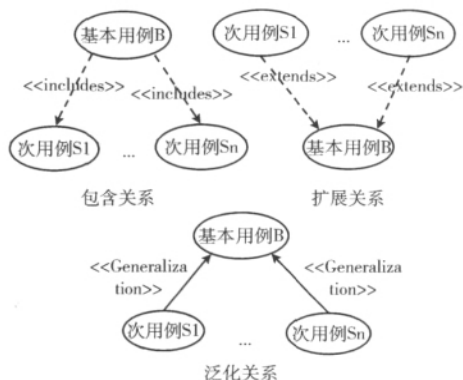


图2 用例间的关系

### 2.2 可重用性用例确定

传统的功能重用性确定主要是通过对功能系统中的基本用例、次用例、扩展用例等之间的包含与扩展关系来该

用例是否为功能系统的重用部分, 该方法是从包含关系与扩展关系两方面进行确定可重用用例, 却没从其他的方面进行讨论, 从而影响可重用性用例的准确选定。

传统方法中, 在功能系统中如果多个基本用例都包含该某一个次用例, 说明多个用例都共用了该次用例, 即该次用例的基本功能在功能系统中被重用; 如果一个次用例扩展两个或多个基本用例与扩展用例, 那么该次用例个功能在系统功能中被重用; 如果一个次用例被一个基本用例包含, 并且它又扩展了一个或用例, 说明该次用例的基本功能在基本用例与扩展用例中都存在, 即次用例的功能在功能系统中重用。传统包含与扩展关系用例重用如图 3 所示。

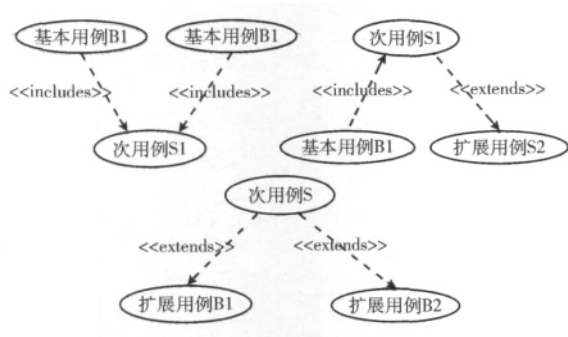


图 3 传统包含与扩展关系用例重用

通过分析, 我们可以发现传统方法只是围绕包含与扩展用例进行功能用例可重用确定, 而在深入分析功能系统中用例间关系时, 发现在泛化关系中同样存在功能用例重用的情况。在泛化关系中, 1 个用例继承了另外 1 个用例的基本功能、行为和结构等, 并可以在继承的基础上扩展新的功能和行为, 那么如果在功能系统中如果某一次用例继承了多个基本用例的某一功能行为等, 那么继承的部分在多个基本用例间是可以被重用的, 即该用例继承的功能在功能系统是被重用。与此同时, 当一基本用例包含某一次用例, 而该次用例继承了另一基本用例, 那么该次用例也被重用等, 加入泛化关系的用例重用如图 4 所示。

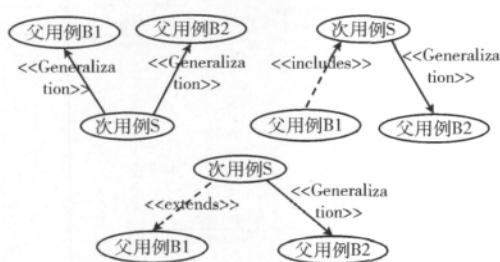


图 4 加入泛化关系用例重用

通过对上述对用例间几种关系的分析和研究, 在可重用用例确定过程中加入泛化关系, 以便更加准确和完整地

确定可重用用例。这里归纳了几个规则来确定可重用用例:

- (1) 当 1 个次用例被 2 个或者 2 个以上的基本用例包含;
- (2) 当 1 个次用例扩展 2 个以上的扩展用例或基本用例;
- (3) 当 1 个次用例泛化 2 个以上的父用例;
- (4) 当 1 个次用例被 1 个基本用例包含并且它扩展 1 个用例或继承了父用例。

根据以上规则对可重用性功能用例进行确定, 为后期功能用例规模的计算和可重用度量提供保障。

### 2.3 可重用性度量过程

通过对用例间的关系, 确定出可重用功能用例后, 使用功能点估算法对系统功能总的用例规模数和可重用功能用例规模数进行计算<sup>[9]</sup>, 根据重用功能用例规模数所占总规模数的比例来确定该功能系统是否重用, 即对该可重用候选服务进行选定。

#### 2.3.1 功能规模数计算

根据功能系统中用例的数据移动, 本文使用 COSMIC-FFP 方法是来计算功能规模数, 该计算方法中一个数据的移动是一组数据组的传输, 并记为 1  $Cf_{su}$ 。该方法中数据移动包括 4 类: 输入 (entry)、输出 (exit)、读 (read)、写 (write)<sup>[10]</sup>, 而在 UML 中用例时序模型描述对象之间的协作关系, 每个用例至少需要一个时序图来对应, 其中对消息的交互有 4 种类型: Signal、Service、Query、Connect, 通过将两者建立相应的映射规则, 就可以通过 COSMIC-FFP 方法对用例功能点规模数进行计算, 映射规则如表 1 所示。

表 1 用例模型与 COSMIC-FFP 模型对应关系

COSMIC-FFP 基本元素	用例和时序模型基本元素
功能过程	用例 (基本用例和次用例)
数据组	时序图中的实体类和用例图中的参与者
数据属性	service 类型消息的常量和变量
输入 (entry)	signal 消息, 是具有输入值的消息
输出 (exit)	signal 消息, 是具有输出值的消息
写入 (write)	service 的创建、销毁; connect 更新构造型消息
读取 (read)	query 消息, 消息发出响应请求

#### 2.3.2 相关计算公式

(1) 每个用例的数据传输规模

$$Size(UC_i) = \sum_{j=1}^n Size(DataMove)_j \quad (1)$$

式中:  $UC_i$  —— 用例  $i$ ,  $DataMove$  —— 该用例  $i$  中数据的移动量, 该量包括输入、输出、读、写等。

## (2) 重用次用例规模总数

$$RFSize = \sum_{i=1}^n Size(SecondUC_i) \quad (2)$$

式中:  $RFSize$  —— 被重用次用例规模总数,  $SecondUC_i$  —— 被重用次用例  $i$ 。

(3) 一个基本用例功能规模总数 (用例本身的数据传输规模和所包含次用例规模的总和)

$$Size_{all}(BaseUC_i) = \sum_{i=1}^n Size(SecondUC_i) + Size(UC_i) \quad (3)$$

式中:  $Size_{all}(BaseUC_i)$  —— 基本用例  $i$  的规模总数,  $BaseUC_i$  —— 基本用例  $i$ 。

## (4) 总的功能规模数 (所有基本用例规模数总和)

$$AFSize = \sum_{i=1}^n Size_{all}(BaseUC_i) \quad (4)$$

式中:  $AFSize$  —— 所有基本用例规模总数。

## (5) 执行度量

$$\text{功能重用性百分比} = \frac{RFSize}{AFSize} \times 100\% \quad (5)$$

## 2.4 候选服务选定过程

候选服务的选定, 可以根据候选服务包含的功能用例间的关系, 确定出具体可重用功能性用例, 并使用功能点规模估算法计算出各用例的规模数, 然后根据可重用功能用例规模所占总功能用例规模的百分比对候选服务进行重用性度量。其过程可具体为以下步骤:

(1) 分析候选服务包含的功能用例, 使用 UML 建立相应的用例关系模型, 根据用例间关系确定出该候选服务中包含的可重用性功能用例。

(2) 将步骤 1 中候选服务包含的功能用例映射成用例的时序模型, 计算出时序模型中数据的移动量。

(3) 通过 3.3.1 中时序模型与功能点规模估算法的元素对应原则, 以及 3.3.2 中计算公式, 计算出各用例的功能点规模数。

(4) 最后, 通过 3.3.2 中的计算公式, 计算出可重用功能用例规模百分比, 对候选服务进行可重用进行度量, 得出结论。

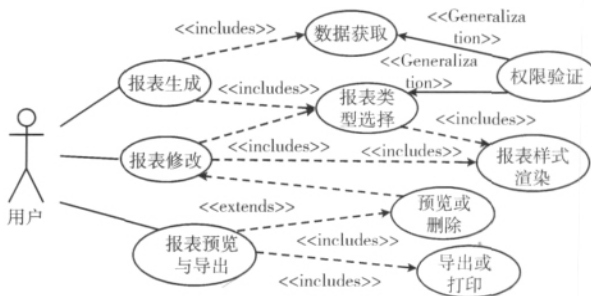
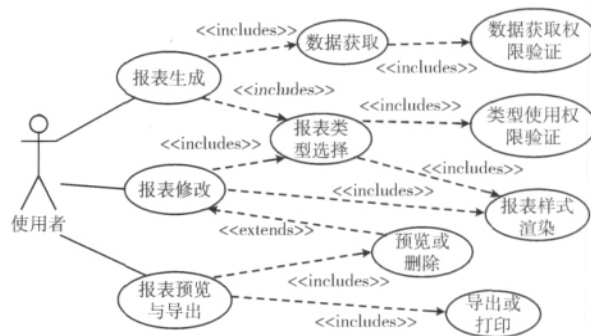
## 3 应用实例

本节结合通用报表生成候选服务来说明从功能重用性角度选定候选服务的过程。根据该候选服务的基本功能, 分析该候选服务所有的功能用例, 并建立相应的用例模型, 确定可重用功能用例。然后使用功能点估算法对用例的功能规模数进行计算, 求出所需规模数进行度量。

## 3.1 建立用例模型

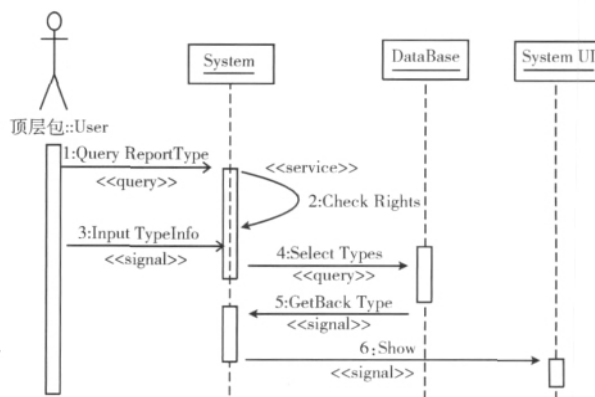
分析候选服务的基本功能, 确定该候选服务包含的功能用例以及用例间的关系, 确定可重用功能用例。对于系

统的功能描述和功能用例模型这里不在详细介绍, 报表生成与管理候选服务的主要功能包括: 报表生成、报表修改、报表类型选择、报表样式渲染、报表权限管理、报表预览与导出等。未考虑泛化关系与考虑后, 该用例图分别如图 5 与 6 所示。



## 3.2 用例对应的时序模型

通过面向对象的需求模型得到用例模型, 根据该模型分别找出用例对应的时序图, 识别出相关的数据或者数据组的传输。本节以考虑泛化关系后的报表类型选择用例对应的时序图为例, 并给出该用例时序图相关的数据移动。如图 7 所示。



在该时序图中, 用户首先向系统发出报表类型选取的请求, 其次系统对用户请求进行权限验证 (不同权限对类型选择的权限不同), 然后验证通过后用户提交报表类型参数, 系统向数据库发出请求提取报表样式, 并返回给用户, 最后展现在系统界面上。

根据前面提到的映射规则, 步骤 1 中使用者信息输入是一个具有输入值的消息 signal, 所以作为 1 个 entry。其他可依照规则进行计算, 这里不再详细描述。最后可以识别出 1 个 entry、2 个 exit、1 个 write、2 个 read。

### 3.3 度量计算

在前面叙述的 COMISC-FFP 方法中, 每个数据的传输作为 1  $Cfsu$ , 根据用例的数据传输的计算公式 (1) 可以计算出类型报表选择用例规模数:  $Size(Forms) = 1 Cfsu + 2 Cfsu + 1 Cfsu + 2 Cfsu = 6 Cfsu$ 。通过以上步骤和相关计算公式把主次用例的相关数据计算出来, 未考虑泛化关系与考虑后用例数据如表 2 与表 3 所示。

表 2 未考虑泛化关系规模 (单位:  $Cfsu$ )

用例	是否主用例	数据传输规模	包含次用例规模	用例总规模	次用例是否重用
报表生成	是	4	24	28	
数据获取		4	3	7	否
报表类型选择		6	11	17	是
数据获取权限验证		3	-	3	否
类型使用权限验证		3	-	3	否
报表修改	是	4	17	21	
报表样式渲染		8	-	8	是
报表预览与导出	是	3	8	11	
预览或删除		5	-	5	是
导出或打印		3	-	3	否

表 3 考虑泛化关系规模 (单位:  $Cfsu$ )

用例	是否主用例	数据传输规模	包含次用例规模	用例总规模	次用例是否重用
报表生成	是	4	18	22	
数据获取		4	-	4	否
报表类型选择		6	8	14	是
权限验证		5	-	5	是
报表修改	是	4	14	18	
报表样式渲染		8	-	8	是
报表预览与导出	是	3	8	11	
预览或删除		5	-	5	是
打印或导出		3	-	3	否

由上表数据与公式可以计算出: 表 2 中该候选服务总的功能规模数为:  $AFSize = 28 + 21 + 11 = 60 Cfsu$ , 可重用功能规模数为:  $RFSize = 17 + 8 + 5 = 30 Cfsu$ , 可重用百分比为:  $30/60 = 50\%$ ; 表 3 中总的功能规模数为:  $AFSize = 22 + 18 + 11 = 51 Cfsu$ , 可重用功能规模数为:  $RFSize = 14 + 5 + 8 + 5 = 32 Cfsu$ , 可重用百分比为:  $32/51 = 63\%$ 。通过分析比较, 发现表 2 中数据获取数据权限验证与报表类型使用权限验证是不可重用的, 其用例重用百分比为 50%, 而表 3 中考虑了用例间泛化关系, 使得权限验证实现重用, 且重用百分比为 63%。

通过上述度量结果发现, 加入泛化关系功能点估算法与面向对象模型使服务建模重用性功能候选服务选定更加准确和完整, 在系统不同功能模块中可以重复的使用该报表模块, 所以在 SOMA 服务建模中可以将系统验证候选服务选定服务, 为建模过程的服务设计与组合提供保障。

## 4 结束语

在 SOMA 建模过程中如何完善、准确、高效的将候选服务选定为服务是建模过程中重要的组成部分, 因此本文结合 COSMIC-FFP 方法和面向对象方法中 UML 用例和时序模型从功能重用性角度去讨论了可重用性候选服务的选定。其中分析了需求模型中功能重用可能存在的用例间关系, 结合 UML 模型的时序图模型进行具体的度量, 最后给出了一个通用服务的确定候选服务过程。实践表明, 加入用例间泛化关系后使得 SOMA 服务建模中可重用候选服务选定更加完善、准确及合理。

## 参考文献:

- [1] MAO Xincheng. SOA methods principle and practice [M]. Beijing: Publishing House of Electronics Industry, 2007: 12-15 (in Chinese). [毛新生. SOA 原理方法实践 [M]. 北京: 电子工业出版社, 2007: 12-15.]
- [2] WANG Linlin, ZHANG Chun, LIU Xin, et al. Survey of SOA modeling in whole lifecycle [J]. Application Research of Computers, 2011, 28 (1): 37-41 (in Chinese). [汪林林, 张春, 刘歆, 等. SOA 全生命周期建模方法综述 [J]. 计算机应用研究, 2011, 28 (1): 37-41.]
- [3] CHEN Liang. SOA-based logistics information system service modeling study [D]. Chengdu: Southwest Jiaotong University, 2009 (in Chinese). [陈亮. 基于 SOA 的物流信息系统服务建模研究 [D]. 成都: 西南交通大学, 2009.]
- [4] JIN Dingyong. A SOA-supporting software analysis and design method [J]. Practice and Application, 2010 (1): 41-45 (in Chinese). [金定勇. 一种支持 SOA 的软件分析和设计方法 [J]. 实践与应用, 2010 (1): 41-45.]

(下转第 4608 页)

## 参考文献:

- [1] LIU Hai, HAO Kegang. Cause analysis method of software defect [J]. Computer Science, 2009, 36 (1): 242-243 (in Chinese). [刘海, 郝克刚. 软件缺陷原因分析方法 [J]. 计算机科学, 2009, 36 (1): 242-243.]
- [2] Taitz J, Genn K, Brooks V, et al. System-wide learning from root cause analysis: A report from the new south wales root cause analysis review committee [J]. Qual Saf Health Care, 2010, 19 (6): 1-5.
- [3] YUAN Donglin. Software process measurement based on orthogonal defect classification [J]. Computer Application and Software, 2007, 24 (3): 65-68 (in Chinese). [袁东林. 基于正交缺陷分类的软件过程测量方法 [J]. 计算机应用与软件, 2007, 24 (3): 65-68.]
- [4] YIN Xiangle, MA Li, GUAN Xin. Research of software defects classification [J]. Computer Engineering and Design, 2008, 29 (19): 4910-4912 (in Chinese). [尹相乐, 马力, 关昕. 软件缺陷分类的研究 [J]. 计算机工程与设计, 2008, 29 (19): 4910-4912.]
- [5] XU Y, MOSTOW J. Using logistic regression to trace multiple sub-skills in a dynamic bayes net [C]. Eindhoven, Netherlands: 4th International Conference on Educational Data Mining, 2011: 6-8.
- [6] JIANG L, ZHANG H, CAI Z. A novel bayes model: Hidden naive bayes [J]. IEEE Trans. Knowledge and Data Engineering, 2009, 21 (10): 1361-1371.
- [7] ZHOU Feng, MA Li. Software defect prediction model based on AODE and resampling [J]. Computer Engineering and Design, 2011, 32 (1): 210-212 (in Chinese). [周丰, 马力. 基于AODE和再抽样的软件缺陷预测模型 [J]. 计算机工程与设计, 2011, 32 (1): 210-212.]
- [8] WANG Q, ZHANG L, Chi M, et al. MTForest: Ensemble decision trees based on multi-task learning [C]. Amsterdam, Netherlands: Proceedings of the 18th European Conference on Artificial Intelligence, 2008: 122-126.
- [9] Subhransu Maji, Alexander C Berg, Jitendra Malik. Classification using intersection kernel support vector machines is efficient [C]. Anchorage, Alaska, USA: IEEE Conference on Computer Vision and Pattern Recognition, 2008: 1-8.
- [10] YAO Quanzhu, TIAN Yuan, WANG Ji, et al. Novel fast training algorithm for SCM based on self-adaptive steps [J]. Application Research of Computers, 2008, 25 (6): 1679-1681 (in Chinese). [姚全珠, 田元, 王季, 等. 基于自适应步长的支持向量机快速训练算法 [J]. 计算机应用研究, 2008, 25 (6): 1679-1681.]
- [11] Van Der Walt C, Etienne Barnard. Data characteristics that determine classifier performance [J]. SAIEE Africa Research Journal, 2007, 98 (3): 87-93.

(上接第 4575 页)

- [5] CHEN Fu, YANG Yang, XIONG Zenggang. A study of fundamental theory on service logic modeling [J]. Journal of Frontiers of Computer Science and Technology, 2008, 23 (4): 439-448.
- [6] CHU Wang, QIAN Depei. Supporting web services reuse by semantic service component and composition pattern [J]. Journal of Southeast University (English Edition), 2006, 22 (3): 311-314.
- [7] CHEN Yanyan, GU Xunmei. New method for software reuse measurement [J]. Computer Engineering, 2010, 36 (24): 68-70 (in Chinese). [陈艳艳, 顾勋梅. 一种软件复用度量的新方法 [J]. 计算机工程, 2010, 36 (24): 68-70.]
- [8] DU Pan, XU Jin. Discussion about the integration of component services with finer granularity in SOA applications [J]. Computer Applications, 2006, 26 (3): 699-702 (in Chinese). [杜攀, 徐进. SOA 体系下细粒度组件服务整合的探讨 [J]. 计算机应用, 2006, 26 (3): 699-702.]
- [9] Abran A. COSMIC method v3. 0 measurement manual [EB/OL]. [2010-04-30]. <http://www.lrgl.uqam.ca/cosmic-ffp>.
- [10] WANG Xinyu, HOU Hong, HAO Kegang. Research and application of COSMIC-FFP methodology [J]. Computer Applications and Software, 2008, 25 (10): 11-13 (in Chinese). [王昕渝, 侯红, 郝克刚. COSMIC-FFP 方法的研究及应用 [J]. 计算机应用与软件, 2008, 25 (10): 11-13.]