文章编号:1001 - 9081(2009)07 - 1767 - 04

主动服务中程序挖掘系统框架的设计与实现

聂 立,钟 珞
(武汉理工大学 计算机科学与技术学院,武汉 430070)
(nieli723@163.com)

摘 要:针对 Web 服务存在的智能化和个性化等问题,引入领域本体和用户兴趣,设计并实现了程序挖掘系统框架。利用本体特征描述构件,提出了一种基于多知识库的构件检索方法和关联构件检索算法,开发了构件检索系统,并以 E-Commerce 领域内构件资源为实例,与关键词和刻面检索机制相比较,验证了该检索方法的有效性。实验结果表明,该方法对于大规模构件库具有较好的查全率和查准率。

关键词:主动服务;程序挖掘;领域本体;本体特征领域模型;用户兴趣模型
中图分类号: TP311;TP393 **文献标志码**:A

Design and implementation of program mining system framework for active service

NIE Li, ZHONG Luo
(*School of Computer Science and Technology, Wuhan University of Technology, Wuhan Hubei* 430070, *China*)

Abstract: To solve the problem of Web service about intelligence and personality, a program mining system framework was designed and implemented through the introduction of domain ontology and user interest. Component was described using ontology and feature, and a component retrieval framework based on multi-repository and an algorithm for retrieving associated components were presented. Then a component retrieval system was given, and an instance with components in E-Commerce field proved validity compared with the retrieval methods based on keywords and facet. The experiments show that the method has higher precision and recall than others for large-scale components library.

Key words: active service; program mining; domain ontology; ontology feature domain model; user interest model

0 引言

程序挖掘是在 Internet 环境下,为实现主动服务提出来的,能解决 Web 服务存在的个性化和集成化等问题。其主要功能是通过分析用户的服务请求,搜索在 Internet 上发布和存在的构件资源,并按一定的规范和标准对它们进行管理,通过构件复用来减少编程工作和实现主动服务。

目前,国内外对程序挖掘技术进行了相关研究。文献[1]给出了程序挖掘的六个过程,研制了一个程序挖掘原型系统 COMP;文献[2]给出了一个程序挖掘实现框架,设置了构件目录服务器;文献[3]设计了程序挖掘代理系统。本文在对程序挖掘系统框架进行深入研究的基础上,针对目前服务中缺少语义信息和个性化等特点,引入领域本体[4]和用户兴趣,设计了程序挖掘系统框架[5],并在基于刻面、关键词检索的基础上,结合领域本体和用户兴趣模型,提出了一种基于多知识库的构件检索策略。文中对基于本体特征的构件描述方法、用户兴趣信息的获取、构件检索框架和关联构件检索算法进行了详细讨论,最后通过实验对该方法有效性进行了评估。

1 程序挖掘系统框架的设计

程序挖掘为用户提供主动服务,它能根据用户的需求,为用户量身定制应用程序。按照用户需求处理的先后顺序,可将程序挖掘过程划分成五个阶段:提交用户需求、用户需求分析与功能提取、构件搜索与获取、构件分析选择与组装、程序验证与执行。

本文将代理技术应用到程序挖掘过程中,在多智能代理运行平台 MARE(Multi-Agent Running Environment)下,引入领域本体库、用户兴趣信息库等多个知识库,设计了基于多知识库的程序挖掘系统框架模型,如图 1 所示。

图 1 将程序挖掘系统分为 5 个模块:用户界面模块、需求分析模块、程序挖掘模块、构件组装模块以及构件库管理模块。

该系统在运行的过程中,首先用户将计算请求经用户接口代理提交到程序挖掘服务器,该服务器启动需求分析代理将用户请求进行需求分析和功能提取,生成需求描述 XML 文档,从文档中提取关键词表,利用构件搜索代理在程序挖掘知识库中查找相同或相似的解决方案,如果有则直接从库中获得程序的功能模块分解方案,以此为依据去搜索和获取相关的构件,并根据构件的实际搜索和组装结果修正解决方案。如果没有,则利用构件搜索代理在构件索引库、领域本体知识库、用户兴趣模型之间自主移动,寻找满足要求的构件,然后通过程序挖掘算法分析构件之间的依赖和调用关系,找出能够实现特定计算功能的构件组合,并使用构件组装代理将构件组装成满足用户要求的可执行程序,再通过用户接口代理提交给用户。最后再把该构件组装方案添加到程序挖掘知识库中,不断丰富程序挖掘知识库的内容。

构件的描述和检索是程序挖掘系统的核心组成部分,下

收稿日期:2008 - 12 - 10;**修回日期**:2009 - 02 - 21。 **基金项目**:教育部高校行动计划基金资助项目(2004XD-03)。
作者简介:聂立(1983 -),女,湖北宜昌人,硕士研究生,CCF 会员,主要研究方向:Web 技术、主动服务、面向服务的计算; 钟珞(1957 -),男,湖南长沙人,教授,博士生导师,CCF 高级会员,主要研究方向:智能技术、软件工程、科学计算可视化。

面分别对这两部分进行详细介绍。

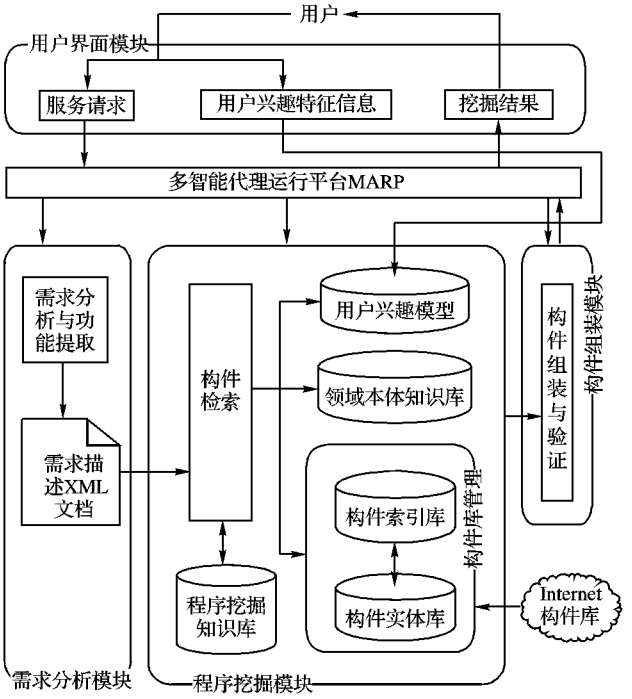

图1 程序挖掘系统框架模型

2 基于本体特征的构件描述

领域具有内聚性和稳定性等特点,因此,软件复用在特定领域中进行容易成功[6]。一个好的描述领域的领域模型是领域内构件复用的基础。目前主流的领域工程模型都是基于特征的领域模型,难以形式化且缺乏语义信息,不利于机器自动处理。由于本体提供了明确的术语定义,领域内概念之间关系的表示易于被机器理解。

2.1 本体特征领域模型的定义

领域本体是对特定领域内概念与概念之间关系的精确描述,一般可用五元组表示: $O = \{C, R, H^c, rel, A^0\}$。其中 C 表示概念的集合;R 表示关系的集合;H^c 表示概念层次;rel 表示概念间的关系,A^0 表示本体公理。服务属性和接口输入输出等都是用领域本体来描述[6]。

在特征模型中,主要包含表示功能的动作(简称动作)和作为动作目标的操作对象(简称对象)两个本体特征,动作作用于对象。对象之间的关系可以按照一般的本体建模方法确立,动作的本体特征不是独立存在的,动作本体特征之间包含着聚集、泛化两种结构关系和一些业务关联。

2.2 基于本体特征的构件描述模型

构件描述是构件检索的基础,特征从用户观点刻画目标系统,其含义用户容易理解,而本体是机器可以自动处理的,因此本节中给出了基于本体特征的构件描述模型 CDM-OF (Component Description Model based on Ontology Feature)。

模型 CDM-OF 的形式化描述如下所示:

CDM-OF = { Fun-P, Non-Fun-P, QoS }

其中 Fun-P, Non-Fun-P, QoS 分别表示构件的功能属性、非功能属性和质量属性。

功能属性: Fun-P = { AD, CF, f_0 },AD 表示构件实际应用的具体领域;CF 是构件在程序挖掘和软件开发过程中所提供的功能集,每个功能可形式化表示为 CF = {operation, {(a, b) | a ∈ facet(operation), b ∈ fat(a)}, object, {(a, b) | a ∈ facet(object), b ∈ fat(a)}},其中 operation 是本体特征领域模型中的动作名, object 是对象名, facet(operation)是动作的刻面集, facet(object)是对象的刻面集, fat(a)表示刻面 a 的术语空间。f_0 是 CF 中动作特征的最小父特征的动作名。

非功能属性: Non-Fun-P 可形式化表示为: Non-Fun-P = {(a, b) | a ∈ facet (Non-Fun-P), b ∈ fat(a)},其中 facet (Non-Fun-P)是构件非功能属性的刻面集, fat(a)表示刻面 a 的术语空间。

质量属性: QoS = {(a, r, b) | a ∈ attributes, b ∈ range(a), r ∈ { >, <, =, <=, >= }},其中 attributes 表示构件的质量属性,例如 attributes = {响应时间,费用,可靠性},range(a)表示质量属性 a 的取值范围,r 表示属性和取值之间的关系,例如响应时间 < 10 s。

在构件检索的过程中,使用构件描述模型 CDM-OF,既便于用户理解又可以被机器自动处理,同时考虑了描述构件自身和构件之间的关联信息。

3 用户兴趣模型的创建

目前,提问式的构件检索方法存在一些需要改进的地方,例如对于不同用户给出同样的检索请求,得到的检索结果都是一样的,不能体现个性化特点;用户兴趣的状态信息没有保留下来,用户每次查询是相互独立的。本文提出的基于用户兴趣的构件检索方法,通过获取用户的背景信息,如知识水平、专业方向、职业和兴趣等,来提高构件检索的查准率,实现构件检索的个性化特点。

用户兴趣模型的建立需要在构件检索系统中扩展一些功能模块来实现[7]。系统需要用户先行注册,然后通过跟踪用户行为,学习、记忆用户兴趣,通过描述用户的兴趣来建立个性化用户模型。最后根据兴趣特征信息对用户检索的构件集合进行过滤、整理,把用户感兴趣的构件提供给用户。

用户兴趣的获取,目前主要有两种方法:一种是通过静态人机交互模式获取用户的信息需求。系统会提供给用户一个表达信息需求的定制表单,用户在表单中填写自己的个性化需求信息。这种方法可以准确地获得用户兴趣信息,但是主动性差。另一类是通过书签挖掘或用户行为监控等方式挖掘用户的访问记录,获取用户的需求、兴趣和爱好等。这种方法的主动性强。

4 基于多知识库的构件检索方法

目前,针对不同的描述方法,涌现出了很多构件检索策略。例如基于正文、基于词法描述符、基于构件行为采样的检索策略[8]等。在实际应用系统中,应用较为成功的是基于枚举、属性值、关键词和刻面[9]的检索方法,其检索条件简单,容易实现,但由于缺少语义信息和个性化等特点,使其检索的有效性还有待进一步提高。本节将领域本体和用户兴趣结合起来,根据其语义性和个性化特点,设计了一种基于多知识库的检索策略。

4.1 基于多知识库的构件检索框架

基于多知识库的构件检索框架如图2所示,首先进行基于刻面的检索,得出检索的结果集合 $C1$,在 $C1$ 的基础上进行基于关键词的检索得到结果 $C2$,再对 $C2$ 进行基于领域本体的检索,得出检索的结果集合 $C3$,将集合 $C3$ 与用户兴趣模型进行相似度计算,并根据相关度排序,得出满足用户需求的结果 C,最后将 C 返回给用户。

将基于领域本体的构件检索结果与用户兴趣模型进行相似度计算时,通常从检索结果 $C3$ 和用户兴趣模型中分别提取构件集合向量 $\boldsymbol{C} = (c_1, c_2, \cdots, c_n)$ 和用户特征向量 $\boldsymbol{P} = (p_1, p_2, \cdots, p_n)$,再利用式(1)进行相似度计算:

$$sim(\boldsymbol{C},\boldsymbol{P}) = \cos\theta = \frac{\sum_{i=1}^{n} c_i \times p_i}{\sqrt{\sum_{i=1}^{n} c_i^2 \sum_{i=1}^{n} p_i^2}} \quad (1)$$

当两向量相同时,相似度为 1;当向量不存在相同项时,相似度为 0,两向量夹角 θ 的余弦值越大,说明搜索的构件越接近用户需求。

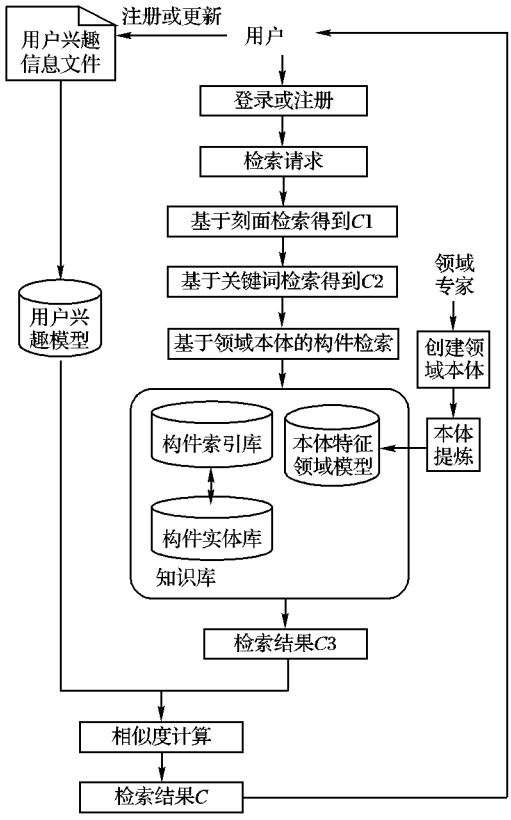

图 2　基于多知识库的构件检索框架

4.2　构件检索过程及算法描述

基于领域本体的检索机制是通过本体特征领域模型与检索词语义词典以功能描述为参照进行检索,用户提出所要检索构件的功能,系统根据其功能描述检索出相关的构件。

本节重点给出基于领域本体的构件检索详细过程。在检索过程中,需要将用户的检索请求转化为用 2.2 节中的构件描述模型 CDM-OF 描述的虚拟构件。即 CDM-OF = { Fun-P,Non-Fun-P,QoS }。

设用户检索请求中各动作名为 $\{a_1,\cdots,a_n\}$,其中动作特征的最小父特征为 f_0,检索时相似度阈值设为 λ,构件检索过程分为以下 3 个步骤:

步骤 1　在构件索引库中查找满足功能 f_0 的构件,放入候选构件集中;如果 f_0 存在二级索引节点,逐一考查,设当前二级索引节点为 f_{0i},如果:

$sim(\{a_1,\cdots,a_n\},f_{0i}) \geqslant \lambda$

则满足功能 f_{0i} 的构件放入候选构件集中,其中 $sim(x,y) = \sum_{a\in x}\sum_{b\in y} sim(a,b)$。

$$sim(a,b) = \begin{cases} 1, & \text{当 } a = b \text{ 或 } b \text{ 是 } a \text{ 的祖先节点} \\ 1/i, & a \text{ 是 } b \text{ 的祖先节点且 } a \text{ 有 } i \text{ 个孩子节点} \\ 0, & \text{其他} \end{cases}$$

步骤 2　在本体特征领域模型中依次考查最小父特征 f_0 的祖先特征和孩子特征,并按步骤 1 同样的方法查找候选构件。

步骤 3　匹配用户检索请求和构件描述。分别计算检索请求和构件描述在功能属性、非功能属性和质量属性三方面的相似性,然后加权相加。

采用 2.2 节中给出的构件描述模型 CDM-OF,基于该模型和本体特征领域模型里的动作和对象之间的关联,用户能够容易地检索到需求的构件。

以下算法采用广度优先的方法,描述了系统如何帮助用户在构件库中检索与当前构件相关联的构件。算法具体描述如下:

输入:本体特征领域模型;检索的构件;

node, nextNode;　　//node, nextNode 为动作节点

F[i] = 0, G[j] = 0;

//i, j = 1,2, …, n, 其中 n 是领域模型中动作数, F[i] = 0 表示领域模型中动作 i 不在候选构件集合中, 否则 F[i] = 1; G[j] = 0 表示待检索的构件功能集中动作 j 没有被处理, 否则 G[j] = 1

输出:满足用户需求的相关构件集合

1) 如果领域模型中的动作 i 包含在待检索的构件动作集中, F[i] = 1;

2) 如果检索请求中动作 p 满足 F[p] = 0, 则 p 为当前节点, 即 node = p;

3) 如果 node 为空, 算法结束, 否则继续;

4) 如果 G[node] = 0, 则处理节点 node, 设 G[node] = 1;

(1) 如果 node 有关联动作 q, 设 q 为下一节点, 即 nextNode = q, 转(2), 否则 nextNode 为空;

(2) 如果 nextNode 不为空, 且 F[nextNode] = 0, 则将 nextNode 以及其对象 m 返回给用户, 用户根据领域模型确定动作;

(3) 将关联动作 q 的所有动作 x 放入到等待队列中, 设 F[x] = 1;

(4) 如果等待队列不为空, 设 node 为队列头节点, 转 4), 否则 node 为空, 算法结束.

5　实验结果与分析

为了验证上述构件检索方法的有效性,我们在 Windows Server 2003 下用 Microsoft. Net 实现了一个构件检索系统,该系统包括了 4.2 节中的关联构件检索算法。

通过实验,将常用的基于关键词检索方法(简称 R1)、基于刻面的检索方法(简称 R2)和基于领域本体和用户兴趣的检索方法(简称 R3)的有效性进行了比较。检索方法的有效性从查全率、查准率和效率(构件检索执行时间)三个方面来衡量。下面通过几个测试用例对三种检索方法进行评估。

测试用例采用电子商务(E-Commerce)领域的多个构件作为实验对象。使用 20、50、100、150、200、300、400、500 八个构件数量级,相同检索条件下测试检索方法 R1、R2 和 R3 的有效性。

实验结果表明,对于不同的构件数量级,R1 的平均查全率和查准率分别是 32%,29%。R2 的平均查全率和查准率分别是 62%,51%。R3 的平均查全率和查准率分别是 78%、84%。可见本文提出的检索方法 R3 取得了较为理想的结果,在以上两个指标中都优于检索方法 R1 和 R2。图 3 分别是三种检索方法在查全率和查准率上的比较。

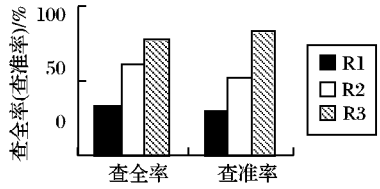

图 3　三种构件检索方法查全率和查准率的比较

三种构件检索方法 R1、R2、R3,在相同检索条件下,对于不同的构件数量级(单位:个)的执行时间详细数据如表 1 所示。

通过分析表1和图4,可以看出,随着构件数量的增加,构件检索方法R3在检索效率上优于方法R1和R2,这种优势在构件库规模大时更明显,更适应于大规模构件库的检索。

表1 三种构件检索方法的执行时间 ms

构件数量	R1执行时间	R2执行时间	R3执行时间
20	139	118	252
50	171	147	284
100	192	162	293
150	263	230	308
200	385	350	352
300	820	692	423
400	1135	1064	497
500	1728	1638	564

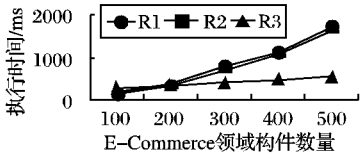

图4 三种构件检索方法检索时间的比较

6 结语

程序挖掘是主动服务的实现机制,本文以主动服务和程序挖掘概念为基础,通过建立程序挖掘知识库、构件索引库、构件实体库、领域本体知识库和用户兴趣模型,设计了基于多智能代理运行平台的程序挖掘系统框架,利用多智能代理完成对用户需求的分析、功能提取,构件的搜索获取、组装以及验证,从而实现按需服务。为了实现程序挖掘,本文将刻面、关键字、领域本体和用户兴趣模型相结合,设计并实现了基于多知识库的构件检索方法。首先给出结合本体和特征领域模型的本体特征领域模型的定义,然后在本体特征领域模型的基础上给出构件描述模型CDM-OF,该模型既便于用户理解又可以被机器自动处理,同时建立了用户兴趣模型,在此基础上,设计了基于多知识库的构件检索框架,并给出一个检索关联构件的检索算法。

最后,通过实验,以E-commerce领域内构件资源为例,对该检索方法的有效性进行了分析,并与常用的基于关键词和刻面的检索方法相比较,验证了该检索方法在一定程度上能够提高某领域内构件检索的查全率、查准率和效率,尤其适用于大规模构件库的检索。

参考文献:

[1] 张尧学,方存好. 主动服务——概念、结构与实现[M]. 北京:科学出版社,2005:43-61.

[2] 窦郁宏. 程序挖掘中构件描述和检索的研究[D]. 长沙:中南大学,2002.

[3] XU KE-GANG, ZHANG YAO-XUE, WEI ZI-ZHONG. PMMAP: A mobile agents platform for program mining[C]// Computer Networks and Mobile Computing. Los Alamitos, CA: IEEE Computer Society, 2001:347-355.

[4] MAARTEN B, BART D W, WOUTER J, *et al*. Ontology-based discovery of data-driven services [C]// SOSE'06: Service-Oriented System Engineering. Shanghai: IEEE Computer Society, 2006:175-178.

[5] ZHONG L, NIE L. Design of program mining system framework for active service[C]// 2008 International Conference on Computer Science and Software Engineering. Wuhan: IEEE Computer Society, 2008:436-439.

[6] 陈刚,陆汝钤,金芝. 基于领域知识重用的虚拟领域本体构造[J]. 软件学报,2003,14(3):350-355.

[7] 张彦,张永奎. 基于层次概念的用户兴趣模型研究[J]. 计算机工程与设计,2008,29(1):181-183.

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

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

(上接第1766页)

```
myCode + = " Event. observe( " + getForId( context) + "," + getEvent
(context) + ", send" + curCompS + "Req, false); ";
```

这些动态代码准备好后,即可交由AjaxAction中的encodeBegin方法呈现到客户端。

3.4 组件的使用

至此,该Ajax使能组件即可投入使用。使用方式与普通的JSF定制组件类似:事先在页面顶端做好相关tld声明,然后,利用该组件对应标签在页面适当位置引用组件,并做好相关属性设置,最后在托管Bean中编写好事件处理代码。

下面以一个具体的实例来说明该组件的使用。在原始页面中,有一个Id为userName的输入组件,其值与托管Bean中的enterName属性绑定在一起,另外还有两个Id分别为info和sysmsg的标签组件,这两个标签组件的值都用el表达式设置为托管Bean的valiadUserMsg属性。

在页面中添加以下代码来为输入组件添加Ajax事件支持:

```
< ajax: action source = "userName" targetset = "info, sysmsg"
event = "blur" actionListener = "#{ userBean. validUser} "/ >
```

然后,在托管Bean的validUser方法中设置事件处理代码:

```
public void validUser( ActionEvent event) {
  this. validUserMsg = this. enterName; }
```

最后,该输入组件就具备相应的Ajax事件处理能力了。

4 结语

本文利用定制请求处理生命周期的思想,为开发支持Ajax的JSF组件提供了一个支撑框架,使得开发出来的组件可以同时支持普通Faces请求和Ajax请求。由于组件采用JSF组件封装机制封装,使用起来非常方便,同时有效避免了单独使用Ajax技术时常见的诸多麻烦和不便。实际应用表明,使用该框架开发Ajax组件,开发强度低,开发效率高,可以充分融合JSF和Ajax技术,使它们优势互补,十分适合Web用户界面的快速构建。

参考文献:

[1] 左学明,张力. 一种新的基于JSF技术的Web用户界面开发方法[J]. 计算机应用,2005,25(1):215-217.

[2] 刘高原,刘觉夫,张国平. 结合Ajax和JSF技术开发Web应用[J]. 微计算机信息,2007,23(12):252-253.

[3] 吴鹏,尹思良,张学杰. 一种基于Ajax的JSF应用改进[J]. 云南大学学报:自然科学版,2007,29(s2):244-248.

[4] SCHALK C, BURNS E. JavaServer Faces完全参考手册[M]. 张猛,译. 北京:人民邮电出版社,2007.

[5] 邝文清,郭跟成. 基于JSF框架Web应用开发的研究[J]. 计算机应用研究,2007,24(12):272-275.

[6] 王峰,江勤绕,俞欢军. 基于JSF框架的信息管理系统的设计与实现[J]. 计算机工程与设计,2007,28(21): 5211-5224.

[7] DUDOEY B, LEHR J, WILLIS B, *et al*. Mastering JavaServer™ Faces[M]. Indianapolis: John Wiley and Sons, 2004.

[8] JACOBI J, FALLOWS J R. Pro JSF and Ajax: Building Rich Internet Components[M]. New York: Springer-Verlag, 2006.

[9] 施伟伟,张蓓. 征服Ajax——Dojo、Prototype、script. aculo. us框架解析与实例[M]. 北京:人民邮电出版社,2007.