

# J2EE 平台下的 PLM 构件库系统的设计与实现

曹曼曼, 曹宝香

(曲阜师范大学 计算机科学学院, 山东 曲阜 273165)

**摘要:**软件构件库作为软件复用的一项重要基础设施,得到了产业界和学术界越来越多的重视。文中结合 PLM 项目背景,提出了一种在 J2EE 平台下的 PLM 构件库系统(PLMCLS)的设计和实现方案。给出了 PLMCLS 的整体框架,并用 UML 用例图说明该构件库管理系统包括的功能。PLMCLS 把 PLM 构件整体上分为九个层次的构件,对各个层次上构件进行刻画描述并提供接口描述信息,在检索和匹配机制采用语义检索和语法匹配的模式,保证了构件的重用率并为快速构建 PLM 系统提供了有效支持。

**关键词:**构件;J2EE;刻画分类;构件检索;构件库;构件匹配

**中图分类号:** TP311.5

**文献标识码:** A

**文章编号:** 1673-629X(2008)05-0248-05

## Design and Realization of PLM Component Library System Based on J2EE Platform

CAO Man-man, CAO Bao-xiang

(College of Computer Science, Qufu Normal University, Qufu 273165, China)

**Abstract:** As an important infrastructure, the component library has gained more attention from software engineering researchers. Based on the PLM project, gives a framework about a PLM component library system, and describes the function of PLMCLS by UML - case. Then introduce a classified construction that the components are classified to nine levels based on the hierarchical classification schemes, and the method of faceted semantic describe and interface syntax describe. In the end, it writes up the method of how to retrieve component by the faceted tree and match component by the interface definition. It is helpful to develop the PLM system quickly and raise the efficiency of component reuse.

**Key words:** component; J2EE; faceted classification; component retrieval; component library; component matching

### 0 引言

产品全生命周期管理(PLM)是指管理产品从需求、规划、设计、生产、经销、运行、使用、维修保养,直到回收再用处置的全生命周期中的信息与过程。PLM 以产品的整个生命周期过程为主线,在产品数据管理(PDM)、协同产品定义管理(cPDM)的基础上,融合了协同产品工具,并有效集成了 CAD/CAM、ERP/SCM/CRM 等应用系统,从而成为支持企业运行的统一的集成平台。

传统的软件开发模式开发的 PLM 系统存在通用性差、柔性低、开发效率低等一系列的问题。随着对软件复用研究的不断深入,基于构件的开发(CBSD)得到

了产业界和学术界的越来越多的重视,目前为止,Sun 公司的 EJB、微软的 COM 以及 OMG 的 CORBA 作为 CBSD 三大主流技术都提出了自己的构件复用模型<sup>[1]</sup>。为了提高 PLM 系统的通用性和开发效率,将 PLM 系统中用到的各功能模块按不同的粒度开发成构件,但如果生产出来的构件只是分散、孤立地存在于某个地方,构件信息无法被人获知、评测,这样的构件是毫无复用价值的。构件需要一个可以被潜在的用户检索、评估、提取的支撑环境。构件库作为一种支持软件复用的基础设施,可以提供对构件描述、分类、存储和检索等功能。目前国内外已有相应的产品,如美国军方和政府资助的 CARDS、SSET、DSRS 等构件库系统,国内的北大青鸟构件库系统和上海构件库系统等<sup>[2]</sup>。它们在对构件检索匹配上大多是完全基于语法的。

文中提出的 PLMCLS 是面向项目和市场的构件库,从 PLM 项目的需求背景和其内涵定义出发,对项

收稿日期:2007-08-26

基金项目:国家自然科学基金资助项目(60072014);山东省自然科学基金资助项目(Y2003G01)

作者简介:曹曼曼(1984-),女,硕士研究生,研究方向为构件技术,软件工程;曹宝香,硕士生导师,研究方向为软件工程。

目中已有的和开发的构件按照功能层次分类,同时分别从语义和语法上对其描述,并作为构件检索的依据。

## 1 基于J2EE的PLM构件库系统框架

根据构件库系统必须具有的基本功能,PLM构件库系统借鉴国内外先进的构件库系统并根据自身的需求,从构件的开发制作、组装装配、提取分类以及检索匹配等方面进行了流程化的设计,将整个PLMCLS分为三大部分:构件开发环境、构件组装使用环境和构件管理环境。图1给出了PLMCLS的整体框架结构。

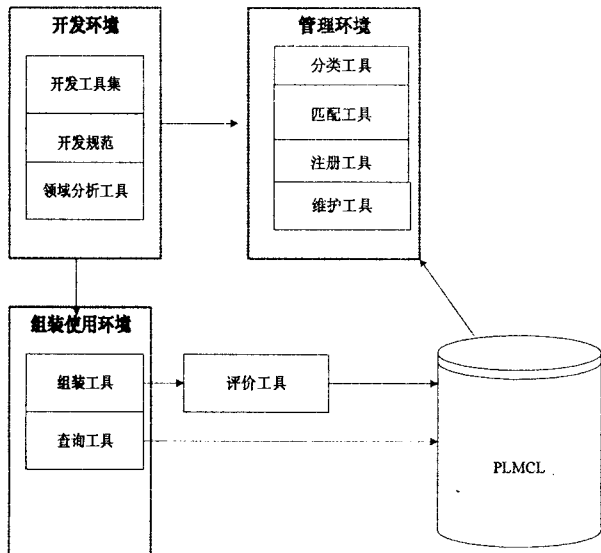


图1 PLMCLS的整体框架结构

开发环境是构件开发者利用一系列开发工具如UML、JDK、Jbuilder等,根据开发规范,完成生产制作符合项目要求的构件,并将其分门别类地存入构件库中等一系列工作;组装使用环境是构件使用者利用构件库系统提供的查询工具从构件库中检索合适的构件或作适应性修改或重新开发,然后进行组装并经过评价工具评价合格后入库的过程;管理环境是构件库系统提供关于构件的分类、描述、检索和匹配的一个支撑环境;评价工具是表示与检索子系统的重要外围工具,对候选构件进行评价,辅助用户选择最符合需求的构件。上述3个环境均基于J2EE技术开发,符合J2EE的关于企业应用系统的开发规范,并按照J2EE标准的分层体系结构搭建,开发的构件也都基于J2EE构件模型开发,符合Java语言语法的规范。所以PLMCLS是基于J2EE平台开发的。限于篇幅,就不再介绍有关J2EE方面的知识,可参考文献[3]。其中构件库管理环境的搭建是整个构件库系统的核心,它的4个主要环节:分类、描述、检索和匹配,关系到构件的查找率和查全率,影响构件的组装和复用等环节。下面就构件库管理系统作详细介绍。

## 2 PLM构件库管理系统

PLM构件库管理系统(PLMCLMS)应该可以有效地管理和组织大量的可复用构件,并提供相应的工具支持软件开发者在开发工作中方便地查找和理解构件,图2用UML-case图给出了PLMCLMS应该覆盖的功能。

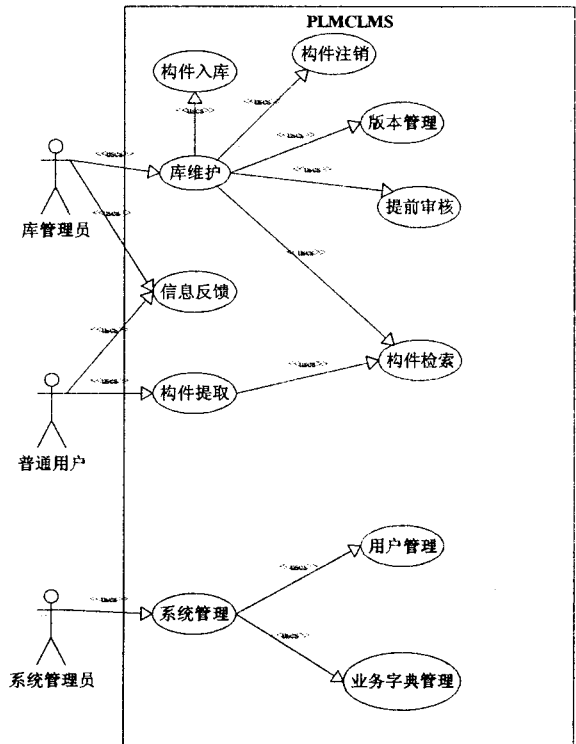


图2 PLM构件库管理系统用例图

(1)构件入库:进行构件的入库信息登记,包括构件的属性信息、构件源码、构件使用手册、构件本身等。这些信息需要一定的构件分类方法支持。

(2)版本管理:构件本身是不断演变升级的,版本管理提供构件历史信息维的功能,包括版本号管理、构件状态管理。

(3)安全审计:构件库管理系统有不同的用户角色:系统管理员、构件管理员、普通用户。用户只可以使用被授权访问的功能,系统记录用户的行为。

(4)构件检索:用户从查询工具处输入所需构件的需求,在构件库中匹配合适的构件,根据描述的方法决定。

(5)构件获取:构件的需求申请、审判、提取的管理。

(6)构件注销:处理过期、报废的构件。

(7)日志管理:用户使用构件库的业务操作日志。

(8)分析统计:构件被检索、提取、使用的相关的分析统计报表,以更方便提供给用户更好的理解构件。

由此可见构件库管理系统研究的重点是构件的分

类和检索,即研究构件的分类策略、组织模式、检索手段和构件相似性分析。

### 3 PLM 构件的分类和描述

#### 3.1 PLM 构件的分类

目前有许多构件分类方法,从构件的表示出发可分为人工智能方法、超文本方法和信息科学方法三类。在实际应用中,构件库基本是采用信息科学的方法对构件进行分类,主要使用枚举、刻面、属性值、关键词和正文检索方法,其中刻面分类方法能够表达丰富的构件信息,尤其引人注目<sup>[4]</sup>;层次分类模式(Hierarchical classification schemes)是另外一种常见的分类方法,它用“自然的”或“逻辑的”方法把每个主题划分为多个分区,最大可能地把相似主题的条目项聚类到一起。文中采用层次分类模式将 PLM 构件在功能逻辑上分为 9 个层次等级的构件,粒度随着层次等级的增高而增大,高层次的构件是由所有它的低层次的构件组装而成,然后对这九级构件再分别进行刻面和接口描述并存储,对于级别低的构件,可以直接存储其实体至数据库;级别高的构件可以存储其存放路径,而用文件系统存放其实体。表 1 是对 PLM 构件进行九级分类的说明。

表 1 九级构件的划分

构件层次	划分标准
一级构件	HTML 组件、页面组态功能构件
二级构件	表单级构件
三级构件	过程级构件
四级构件	用况级构件
五级构件	业务级构件
六级构件	业务域级构件
七级构件	应用域级构件
八级构件	系统级构件
九级构件	系统群集构件

一级构件主要是一些像标签、文本框、按钮等 HTML 的组件和日历控件以及数据库操作的一些运算级的构件,一般直接用面向对象的语言编写就可以;二级构件又称表单级构件,主要是通过构件组装工具将一级构件组装成具有一定逻辑功能的构件,如查询表单、更新表单等;三级构件是可以完成某一具体的操作过程的构件;四级构件是以 UML 为组装工具,完成一组用况的构件,以使需求分析更加充分;五级构件完成一个业务流;六级构件是完成一类业务的构件;七级构件是应用领域的构件,如 PLM 系统由 SCM/ERP/PDM 等不同应用领域的构件组成;八级构件是一个系统级构件,如 PLM 构件系统;九级构件是不同系统级

构件的集成的通用的 MIS、EIS 等。

#### 3.2 PLM 构件的描述

为了方便组装和检索,给出其刻面和接口描述信息,刻面描述是一种语义描述,接口描述是语法描述,具有唯一性。在项目开发中,根据 PLM 的应用需求,给出了 5 个刻面值:使用环境(application environment)、功能(function)、应用领域(application area)、形态(type)、表示方法(representation)。使用环境是使用该构件所需要的硬软件环境;功能是该构件在原有或可能的软件系统能提供的功能集;应用领域是构件在原有或可能被使用到的应用领域的名称;形态是指该构件以上述九级构件中的哪种形态被使用;表示方法是用来描述该构件内容的语言形式或媒体。每个刻面都有相应的标准的刻面值,这些刻面值还具有一个达到一定阈值的语义同义词表。每个构件的刻面信息用 XML 来描述,例如一个查询客户资料的构件,它的刻面描述信息可以表示为:

```
<component>
  <application environment>
    <database>oracle</database>
    <operation system>windows</operation system>
    .....
  </application environment>
  <function>select information</function>
  <application area>CRM</application>
  <type>Two</type>
  <representation>IDL</representation>
</component>
```

文中刻面描述中的表示方法是用接口描述,接口描述是对如何使用构件的语法描述,大多数构件既对外提供服务,又需要外界为它提供某些服务,这种依赖关系也是构件语境的一种体现。文中划分的九级构件中,除了一级构件不用接口描述外,都应给出接口信息,因为一级构件封装粒度小,可以将其代码存储,使用时将其转为 JSP 标签。PLM 构件的接口都由一个或多个接口界面组成,接口界面又由构件对外提供及需求的服务组成,下面给出构件服务和接口界面的定义。

定义 1: 构件服务记为 CService, 是一个和接口中其他功能没有关联的功能;或者是一个由若干功能组成的功能集,这个功能集满足如下条件:

- (1) 同属于构件对外提供的功能,或者同属于构件对外需求的功能;
- (2) 这些功能按设定的交互协议关联执行;
- (3) 对外提供一个单一的接口。

与 plug and socket 类型接口中定义的服务不同,这

里是把 provides 和 requires 中的功能分属为不同的构件服务<sup>[5]</sup>。因为在检索时,对 provides 和 requires 的要求不同,在复用时,对 requires 的要求更加严格,provides 的冗余不会影响用户的工作,而 requires 的冗余会迫使用户再次检索新构件或自己去实现这些功能。

文中的接口描述采用 BNF 形式描述如下:

```
<CService_declaration> ::= CService_ <N> ([<para_type_list>]) return([<para_type_list>]) [invariant] [precondition] [postcondition]
```

```
<para_type_list> ::= <para_type> , <sum> { ; <para_type> , <sum> } ;
```

CService\_ <N> 是构件服务的序号; <para\_type\_list> 是构件服务参数列表,标明了参数的类型和每种类型的数量(sum),如 int, 2; float 1; invariant 为不变量,定义在构件服务执行中和执行后必须有效的状态; precondition 为前驱条件,定义在构件服务执行前必须满足的条件; postcondition 为后继条件,定义构件服务执行后哪些条件有效。

定义 2: 接口界面记为 CInterface, 是对构件的一种使用方法的语法描述,由构件对外提供的服务和对外所需的服务组成。

构件接口描述形式如下:

```
<Component interface description> ::= Module <Component_name> { CInterface CInterface_ <N> <CInterface_declaration> } End CInterface CInterface_ <N> }
```

其中: <Component\_name> 是构件的名字,由开发者指定, <CInterface\_declaration> ::= provides: { <CService\_declaration> }

requires: { <CService\_declaration> }

provides 部分可以有一个或多个, requires 部分可以有 0 个或多个。

这种接口描述方法剔除了传统的构件接口描述中的语义的描述,同时给出对外需求和对外提供的服务,具有唯一性,有利于提高构件的查全率和检索效率。

## 4 PLM 构件的检索与匹配

PLM 构件的检索和匹配分为两个步骤:

(1) 检索过程, 寻找满足要求的构件。

(2) 匹配过程, 检查这些构件的接口是否与待组装的构件的接口或系统接口相匹配。

### 4.1 PLM 构件的检索

构件的分类和描述方法决定着构件的检索机制, 高效的检索使用户能够快速定位到所需的构件, 从而满足用户重用行为的需要<sup>[6]</sup>。文中提供两种检索方式, 一种是将九级构件按照层次高低组织成一个树, 并

生成一个索引目录, 在查找时可以根据其所处的功能层次, 点击其内部包含的构件名称, 就会显示该构件的刻画描述信息。另一种是直接按照刻面值检索, 将构件的刻画描述映射为一棵无序标签树, 即构件中出现的刻画名和子刻画名转化为相应层次的父节点和子节点, 构件的刻画描述术语值为叶子节点, 然后用一个虚拟根节点将它们组合为一个构件树, 如图 3 所示。第一种检索方法适合于明确构件的层次级别的情况, 第二种是本 PLM 构件库主要的检索方式。这样构件的查询可以转化为一棵树查询与构件库中构件描述树之间的匹配问题, 查询结果即为满足查询条件的构件描述树。针对构件刻画描述树的查询, 文献[6]提出了构件树路径包含匹配模型, 把基于刻画描述的构件树用组成构件的路径字符串连接表示, 用子字符串代表某刻画属性路径, 即将路径匹配转化为字符串的匹配, 应用高效字符串查询方法来实现构件查询的目的, 并给出了查询算法。文中借鉴这种刻画检索机制, 实现构件的检索, 得到一个或多个构件描述树。图 4 给出路径包含匹配模型。其中 Q 是查询树, T 是构件库中某一构件描述树。

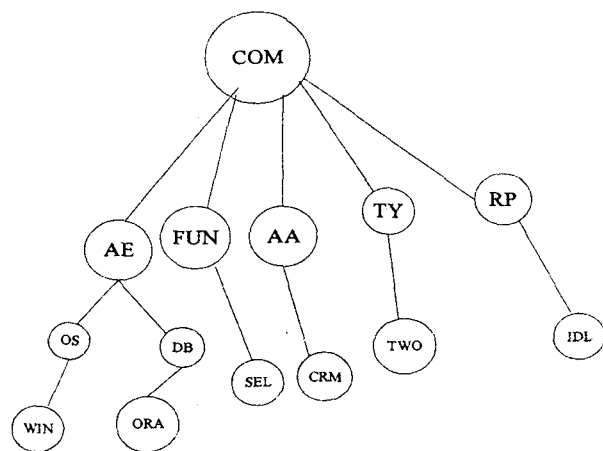


图3 构件的刻画描述树

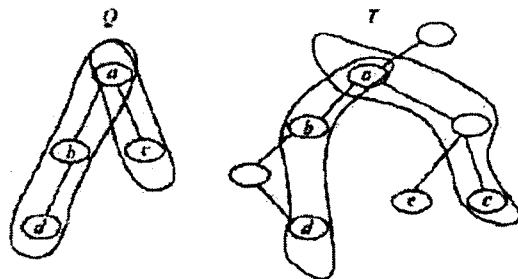


图4 构件树的路径包含匹配模型

### 4.2 PLM 构件的匹配

无论用上述两种检索方法中哪种得到的构件都不能保证是最合适的, 为了进一步缩小查询范围, 还应该

再进行构件匹配。针对提出的构件接口描述,判断构件是否匹配应该考虑两方面的因素。第一,匹配构件与被匹配构件是否具有一致的接口界面;第二,对应的接口界面提供的 provides 服务和 requires 服务是否一致<sup>[7]</sup>。

从语法的角度上看,由于构件服务的名字和构件接口的名字是可以变化的,不能体现接口的语法本质,而构件服务的参数服务列表和返回值类型是不可替换的,代表着接口语法的本质特征,因此参数类型和返回值列表是语法中与检索相关的信息。

对于接口描述中构件服务所给出的是构件服务的序号,  $\langle Cservice \rangle \rightarrow \langle N \rangle$ , 即只采用 Cservice\_1, Cservice\_2... 的形式标明构件服务的序号,这样可以简化构件服务的检索,消除不同名方法的影响,提高查全率。这里要规范一下 Cservice 的序号,所有序号的排列都按照如下规则确定:

(1) 没有参数的构件服务,序号较小。

(2) 比较各个构件服务  $\langle para\_list\_type \rangle$  中的第一个参数的类型,按构件库规定的顺序,参数类型在前的,序号小,规定的顺序依次是 char, string, short, int, long, float, boolean。参数类型相同的,参数数量少的序号小。

(3) 前一个参数相同时,比较后一个参数。

(4)  $\langle para\_type\_list \rangle$  相同的构件服务,分配连续的序号。

(5) 序号从 1 开始。

对于接口界面的序号,同样要遵循一定的规则:

1) requires 部分中 Cservice 的数目较少的,序号小。

2) requires 部分中 Cservice 的数目相同的分配连续的序号。

3) 序号从 1 开始。

按照上述规则将构件服务和接口界面命名规范后,则可以给出构件  $A(CI_1, CI_2, \dots, CI_N)$  和构件  $B(CI_1, CI_2, \dots, CI_M)$  匹配的必要条件是,对于 A

中的每个  $CI$  (接口界面) 的和 B 中的每个  $CI$  的 provides 部分是包含关系,但是 requires 部分是精确等于。

即  $(A.CI_1 \langle N \rangle . provides \subseteq B.CI_1 \langle N \rangle . provides \vee A.CI_1 \langle N \rangle . provides \subseteq B.CI_1 \langle N \rangle . provides) \wedge A.CI_1 \langle N \rangle . requires = B.CI_1 \langle N \rangle . requires$

## 5 结束语

文中以 PLM 项目为背景,给出 PLM 构件库的整体设计框架,并重点介绍了 PLM 构件库管理系统的分类描述和检索匹配。创新点是从功能逻辑层次上将构件分为九个层次的构件,并对每个构件进行剖面语义描述和接口语法描述,统一接口描述规范,使其具有唯一性,在检索和匹配时也是遵循语义和语法相结合的方式,给出了构件语法匹配的必要条件。在一定程度上提高了查全率、查准率和检索效率,整个构件库系统搭建都是在 J2EE 平台的支持下,为以后快速搭建 PLM 系统提供了有利的保证。

## 参考文献:

- [1] 李玉山,侯德文. 基于 Webservice 的构件的分类与检索[J]. 计算机技术与发展, 2006, 16(3): 69-70.
- [2] 潘颖,赵俊峰. 构件库技术的研究与发展[J]. 计算机科学, 2003, 30(5): 90-93.
- [3] 杨绍方. 深入掌握 J2EE 编程技术[M]. 北京: 科学出版社, 2002.
- [4] 常继传,李克勤. 青鸟系统中可复用构件的表示和查询[J]. 电子学报, 2000, 28(8): 20-23.
- [5] Edwards S H. Toward a model of software subsystems[C]//In Latour L, Philbrick S, Stevens M, eds. proceeding of the fifth Annual Workshop on Software Reuse. Los Angeles, CA, USA: [s. n.], 1992.
- [6] 贾晓辉,陈德华. 基于剖面描述的构件查询匹配模型及算法研究[J]. 计算机研究与发展, 2004, 41(10): 1634-1638.
- [7] Canal C, Fuentes L, Pimentel E, et al. Extending Corba interface with protocol[J]. The Computer Journal, 2001, 44(5): 448-462.

(上接第 247 页)

码进行修改,扩展所需的功能,所以,Struts + Spring + Hibernate 框架开发物流管理系统是一种非常好的解决方案,相信经过实践的检验,Struts + Spring + Hibernate 联合框架开发物流管理系统会有更美好的前景<sup>[5]</sup>。

## 参考文献:

- [1] 肖剑锋,雷升印. 基于 Struts 与 Hibernate 的 MIS 开发[J]. 微计算机信息, 2006(22): 22-24.
- [2] 华涛,都克刚,葛玮. 基于 Hibernate 和 Spring 框架的 Web 应用研究[J]. 计算机技术与发展, 2006, 16(11): 84-86.
- [3] 周彩兰,李素芬,孙琳. Hibernate 在 Spring 中的研究与应用[J]. 计算机技术与发展, 2006, 16(10): 62-64.
- [4] 张琛,吴跃,邱会中. 基于 Struts + Spring + Hibernate 的整合架构及其在电信业中的应用[J]. 计算机应用, 2006(12): 265-266.
- [5] 穆福森,吴观茂. 基于 Struts + Spring + Hibernate Web 应用开发框架技术[J]. 电脑知识与技术, 2006(2): 81-82.