

构件复用的 Web 化支撑模型研究

蔡智明, 刘宗田

(合肥工业大学计算机系, 合肥 230009)

摘 要: 本文基于一种以网络、getn 及领域为基础的 Web 化构件复用支撑模型, 简述了其采用的框架模板表示方法及对构件化软件的分层规范: 构造了领域框架、构件的分布式描述类库, 提出以多 Agent 的手段实现分布式的构件搜索、管理, 描述了各 Agent 的主要逻辑及相互协同关系. 原型实现中使用了 ASP、Java、JDBC、SQL 等工具, 并在实际开发中进行了应用试验.

关键词: Web; 软构件; Agent; CASE

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2000) 03-0142-03

A Model to Support the Component Reuse on the Web

CAI Zhi-ming, LIU Zong-tian

(Department of Computer and Information, Hefei University of Technology, Hefei 230009, China)

Abstract: Based on a model to support the software reuse founded on Agent, network and domain, this paper outlines the frame representation of the model and the specifications of the software built by components on different levels. The distributed class-library is constructed for description of domain frame and domain components. Agents are used in component searching and management. The logic of Agents and relations between them are given. A prototype system of the model is implemented by ASP, Java, JDBC, SQL etc. and tested in some software development.

Key words: Web; software component; Agent; CASE

1 引言

目前, 软件构件化技术, 诸如 COM、CORBA、OpenDoc 等, 由于开放性、产业化、系统化特别是软件重用的需要而发展迅速^[6,8]; 开放与集成相统一、多元化资源相互补等成为软件发展的重要特点^[14]; Internet 则为软件资源的共享提供了全球化的通讯、交互媒体, 使大范围、大规模、分布式的软件复用成为可能; 基于 Agent 的软件工程 (ABSE) 和基于知识的软件工程 (KBSE) 则为软件提供了更深层次的构造与互操作手段^[5].

基于以上背景, 我们提出了一个 Web 化的构件复用支撑模型 AND-CASE^[9], 其核心思想是: 基于 Internet/ Intranet 为软件复用提供分布式共享平台, 利用 Web、Java 等有关技术为这种复用提供可操作的技术支持; 应用 Agent 机制解决其中的分布式搜索与管理这一关键问题; 同时以领域层次构架为基础, 对领域对象进行对象建模与层次结构组织, 建立分布式构件对象描述库体系^[2].

为从概念表示层上支持 AND-CASE 的构造与设计, 首先解决领域框架 (DF), 及各领域构件 (DC) 的表示描述及其关系结构的组织这一基础问题, AND-CASE 将 AI 的知识表示^[7]中的框架方法应用于 DF、DC 的特征表示及层次关系构架中, 采用与领域层次的同构框架, 在各 AND-CASE 的 Web 站点上构

造相应的分布式构件描述库 DCB, 考虑到领域范围及构件数量的庞大, 各 DCB 中并不保存各个构件本身, 而仅保存描述 DF、DC 特征属性的对象类及其相互关联, DCB 中各领域、构件间的继承、交叉、细化等关系将各 DF 对象及 DC 对象连接为一个分布有向网络图 DCBG, DF 是 DCBG 的非终结点, 可进一步延伸; DC 则为 DCBG 的终结点. 在终结点 DC 对象中则保存构件实际存放位置的 URL 地址.

针对 DCBG 体系的构架特点 DF、DC 被视为两块知识模板, DF 是描述领域特征属性及其相互关系的知识模板; DC 是描述领域构件的特征属性及与 DF 的关系的特性知识模板, 并通过相互关联的框架网络体系来表达 DCBG 的层次结构, 一个网络结点为一个框架, 每一个框架表示一个 DF、DC 对象的特性知识模块; 每一框架中又定义若干个槽 (slot), 每个槽表示某一特定的知识特性; 通过定义一些特殊的槽与槽值来实现框架间的关联及分布式构造; 在 AND-CASE 服务器上, 则依照这种方法表示的框架网络同构构造分布式的 DCB (详情另见参考文献 [10, 11]).

2 构件化软件的分层规范

为研究构件化软件分层次、多粒度的构造特性, 可分别从体系结构、构件组合、程序配置、构件特征几个层次方面来探讨构件化软件的特征规范^[18, 19].

收稿日期: 1998-10-26; 修订日期: 1999-09-20

基金项目: 教育部博士点基金资助课题; 南京大学软件新技术国家重点实验室基金资助课题; 原机械部发展基金资助课题

在体系结构层上,可采用上下文无关语法,将一个软件体系结构^[4]看作由许多个 NAPE(nattaching-point-entity) 实体互相连接的从结构,每个实体又对应着一个或若干个构件;在构件组合层(CompComponent)上,定义 Component CompUses 是组合中的构件集,Connector ConnUses 是组合连接子集,连接子表示构件角色与涉及者之间的连接,Bind 是外部连接和内部连接的绑定,则:

Compcomponent ::= { CompUses| ConnUses| Bind| Connector } +

在程序配置语法中,定义 Domain 为领域描述,Interface 为构件接口,Config 为构件接口、连接子、角色配置描述,则:

Program ::= < Programname > ; < Domain > ; < Interfaces > ; < Components > ; < Connectors > ; < Config >

在基本的构件特征描述中,依照 DF/DC 的知识表示方法,并经实际应用提炼,定义 DC 特征槽:

<DC> ::= <DCName> { ; <DC-slot> } * [< HTML 详述文档 > | < 详述文档 hyperlink >]

<DC-slot> ::= < Category > | < Super > | < Sub > | < URL > | < Keywords > | < AccessCount > | < Visited Time > | < Outline > |

其中,Category 为构件封装类型,Super、Sub 为父子类链接,URL 为实际构件存放网址,Keywords 为描述关键字,AccessCount 为被访问次数,Visited Time 为最近访问时间,Outline 为构件功能概述,详述文档为网页形式的构件详细功能描述,也可作为超链链接以减少 DCB 本身的信息量。

3 AND-CASE 的多 Agent 体构造

Agent 机制被引入到 DCB 的分布式管理与检索中,在每个 AND-CASE 站点服务器上,均有四个具有 Agent 特性的软件体 SA、MA、GA、LA;各服务器上均有机、功能相同的同类 Agent,形成既相对独立又相互关联的四个 Agent 体系 SAs、MAs、GAs 和 LAs,以此形成一 AND-CASE Agent 的立体型社会体系,如图 1 所示。Agent 的自主性表现为 GAs、LAs、SAs、MAs 以相互关联的多线程方式各自根据内部的状态及外部事件自主决定自己的行为;其独立性表现为 GA、LA、SA、MA 各为范畴明确,可被独立引用的逻辑、过程与数据实体;其能动性表现为任务(如搜索、管理事务)运作从一个服务器往另一个或几个服务器上的迁移;其协同性则表现为同一 Agent 体系内及不同 Agent 之间的消息通讯、协作工作及知识交互。

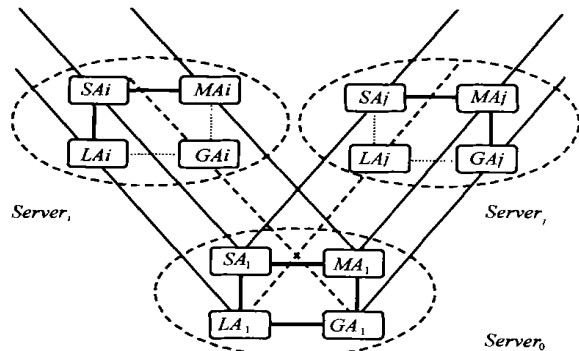


图 1 AND-CASE Agent 体系

GA 为 AND-CASE Server 上 DCB 的同步 HTML 描述文档构造体,GA 按照 DCB 的当前结构和内容动态自动生成相应 DF 和 DC 的描述网页。用户可通过浏览器浏览网页,观察、评价、获取相应 DF 或 DC 描述;也可通过浏览器向 DCB 提交自己的构件。管理体 MA 一旦发生对 DCB 的操作事件,即与 GA 通讯,GA 即刻重新生成相应 DF 与 DC 的描述网页,以保持页面与 DCB 结构、内容的同步一致。

搜索知识学习体 LA 可以对用户浏览搜索的有效知识(例如某领域的某搜索特征要求,如关键词、构件类型,与某个 URL 地址之间,通过浏览建立的联结),进行学习和记忆。搜索体 SA 在搜索中首先与 LA 通讯,如 LA“记忆”中已有相同特征的搜索事件,则可利用记忆的知识准确定位或缩小 DF 范围或直接推理获得匹配的 DC。这样,可避免相同目标的类似反复多层复杂浏览,提高搜索的效率和准确性。LA 对学习到的搜索知识拟人化地自主进行积累、分类、筛选和进化,对积累的搜索知识按照“新频度”—— N 度分级为多种“记忆态”动态记忆、适时进化。

LA 知识库中的知识按搜索特征组织存放,每条知识记录 K 的 N 度计算公式是:

$$N_k = \frac{AT_k}{CT - LT_k + 1}$$

其中, $\frac{1}{CT - LT_k + 1}$ 为调节因子系数, AT_k 为 K 的被访问次数, CT 为当前时间, LT_k 为 K 的上次访问时间。

为使“新鲜”的知识能被优先引用,并保持 LA 知识库的进化,以淘汰“老化”的知识,避免知识库过于庞大而影响检索效率,特设置了一组权值系数 $B_0 \dots B_n$,当知识 K 的 N 度大于权值 B_i 时,LA 将 K 的索引相应保存在“活动态记忆体索引 I_i ”中,搜索体 SA 在搜索中,依 $I_n, I_{n-1}, \dots, I_1, I_0$ 的次序引用索引,因而 N 度较高(即较“新鲜”)的 K ,将较为优先、快速的被应用。LA 自主对各 K 的 N 度刷新,对某些逐渐“陈旧”的 K ,其 N 度将不断降低,当某个 K_j 的 N 度 $< B_0$ 时, K_j 即被进化出“记忆索引”和知识库而被 LA“遗忘”。

搜索体 SA 接收构件搜索的请求委托事件,并通过各服务器上 SA 之间的协同实现分布式有效搜索。在协同方式上我们提出了“并发拍卖”、“传递拍卖”及“指派传递”三种方法;关于三个算法的详细逻辑和性能比较见参考文献[12、13]。各 SA 在执行搜索时又与 LA 保持协同,首先依据 LA 学习的搜索知识,对在 LA 知识库中记忆的满足搜索特征要求的 DC,将直接从记忆体索引中回忆、匹配;若记忆体中满足特征要求的 DC 有多个,则 N 度较高的将被优先匹配;若记忆体搜索失败,则按领域分类, N 度分级,由大到小,由新到旧逐层推理搜索。

DF、DC 对象的增加、修改、删除等操作由管理体 MA 提供方法。MA 的操作方法只有具有 AND-CASE 站点类库管理权限的管理程序和管理员方可操作,一般用户只能通过 GA 生成的同步网页在 DCBG 中浏览、查寻。MA 还采用类似 LA 管理 K 的办法自主对保存在 DCB 中的各 DF、DC 对象类依 N 度计算规则计算各类的 N 度,并在服务器空闲时依 N 度动态调整各类在库中的位置, N 度较大的,将安排在优先被浏览和搜索的位置上。采用类似内存页面淘汰的 LFU 算法,最近较久较少使用的类将被淘汰出 DCB,以保持进化与新陈代谢,控制 DCB

的总 N 度及总体规模。

4 设计与实现

我们在由 Windows98/NT 环境构成的一个 Intranet 平台下,采用 Web 化的技术对 AND-CASE 进行了原型实现.设计实现方案如图 2 所示.

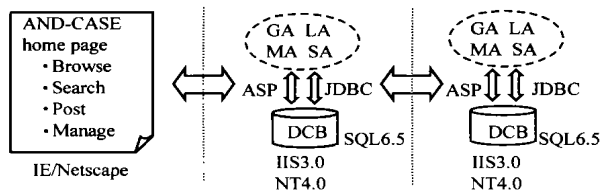


图2 设计实现方案

NT4.0 下的 Web 服务器采用 IIS3.0,其可以通过 HTTP 向 Intranet/Internet 发布信息.DCB 的设计采用 SQL Server6.5;GA、MA 采用 ASP(Active Server Pages)建立与 DCB 之间的接口,用 ASP 服务端 Scripting(脚本)语言实现,ASP 技术十分便于动态发布与管理 DCB 中的构件描述信息.LA、SA 使用 Java 语言实现,通过 JDBC 访问 DCB,SA、LA 的复杂逻辑与交互操作的实现十分需要 Java 语言的跨平台、灵活性、网络交互能力等特性;每个 Agent 均有三类主要线程:(1)监听线程:监听外部事件;(2)服务线程:提供各种操作服务;(3)状态线程:监控内部状态.每个服务器上设计了一个公用电子信箱实现同一服务器上各 Agent 之间的通信;跨服务器的通信则采用了流套接字(Stream Socket)机制,通过 TCP 协议提供的一种可靠的端对端的通信通道实现网络上 Agent 之间的交互.

DCB 中的构件及其描述信息选择了我们实际开发中经常涉及的 MIS 系统的 PowerBuilder 构件作为实例,并已在实际开发中通过 Intranet 实现开发人员间的共享复用,有效提高了开发效率,现已在两个煤矿 MIS 信息管理系统,一个电力 MIS 信息管理系统和一个啤酒企业信息管理系统的开发中应用.进一步准备再扩大领域(如工业控制)与构件类型和数量进行实验与研究.

5 结束语

AND-CASE 模型拓展了一种 Web 化的分布式构件复用工作方式,探索了构件化软件的多层结构规范.利用 KBSE 和 ABSE 的方法手段,特别考虑应用 Web 化的技术及 Java、ASP、JDBC、SQL 等工具实现了原型,提供了一种支持分布式软件复用的软件支撑模型.在研究方法上试图将构件化复用、知识工程、领域工程、Web 化技术有效地结合起来,尤其是尝试着将 Agent 机制引入分布式、Web 化的资源搜索、管理中,以寻求一种解决这一类问题的有活力的方法途径.

感谢秦宗贵、黄伟东、李锐、鲍敢峰、黄自强、孙惠杰、汪巨涛等为本项目所做的大量工作.

参考文献

- [1] William B, Thomas P. An empirical study of representation methods for reusable software components. IEEE Transactions on Software Engineering, 1994, 20(8): 617 ~ 630
- [2] Mili M, Mili F, Mili A. Reusing software: issues and research directions. IEEE Transactions on Software Engineering, 1995, 21(6): 528 ~ 562
- [3] Gacek C. Exploiting domain architectures in software reuse. ACM Software Engineering, 1995, 20(8): 229 ~ 232
- [4] Carlan D. First international workshop on architectures for software system workshop summary. ACM Software Engineering, 1995, 20(3): 84 ~ 89
- [5] Wooldridge M. Intelligent Agents: Theory and Practice. Knowledge Review, 1995, 10(2)
- [6] Adler R M. Emerging standards for component software. IEEE Computer, 1995, 18(3): 68 ~ 77
- [7] 石纯一等. 关于知识表示的讨论. 计算机学报, 1995, 18(6): 212 ~ 223
- [8] 冯玉琳等. 面向对象的组合软件工程研究. 计算机学报, 1996, 19(3): 237 ~ 241
- [9] 蔡智明等. Web 上的组件化柔性 CASE 研究. 计算机科学, 1999, 26(3): 77 ~ 81
- [10] 蔡智明等. 分布式领域组件的框架表示. 微电子学与计算机, 1999, 16(4): 22 ~ 26
- [11] 蔡智明等. The Knowledge Representation and Agent Construction with a System for Component Reuse on the Web. Advances in Computer Science and Technology, ICYCS '99: 273 ~ 275
- [12] 蔡智明等. The Cooperative Search based on Multi-Agent in AND-CASE. ISFST '99: 417 ~ 420
- [13] 蔡智明等. The Distributed Cooperative Work of Multi-Agent on Network. CAID '99: 231 ~ 235
- [14] 蔡智明等. CASEE 的集成与开放. 96 全国软件工程会议论文集: 79 ~ 84
- [15] 蔡智明等. CASE 构件的表示与分层. 合肥工业大学学报, 1996, 19(4): 60 ~ 63
- [16] 鲍敢峰等. 软件复用库的模糊表示和查询方法. 小型微型计算机系统, 1997, 18(4): 27 ~ 31
- [17] 鲍敢峰等. 软件规格说明综述. 软件, 1996, (6): 14 ~ 19
- [18] 鲍敢峰. 基于复用的软件体系结构描述语言的研究. 合肥工业大学硕士论文, 1997
- [19] 孙惠杰. 基于复用的软件体系结构描述及实现环境的研究. 合肥工业大学硕士论文, 1998

蔡智明 副教授, 博士. 主要从事软件工程、分布式软件、软件工具与环境等方面的研究.

刘宗田 研究员, 博导. 主要从事软件工程、反编译等方面的研究.