

基于进程组构件模型的软件体系结构设计方法

褚 华 陈 平

(西安电子科技大学软件工程研究所 西安710071)

摘 要 软件工程的有效性在于进行体系化或系统化的设计,软件体系结构的好坏对系统的成败起着举足轻重的作用。本文针对并发系统,结合面向对象、构件以及软件复用的思想,提出了一种基于进程组构件模型的软件体系结构设计方法。最后介绍了采用该方法设计的 iCALL[®] 系统。

关键词 软件体系结构,构件,面向对象,进程组构件模型

A Method of Designing Software Architecture Based on Process Group Component Model

ZHU Hua CHEN Ping

(Institute of Software Engineering, Xidian University, Xi'an 710071)

Abstract The efficiency of software engineering is on the architectural design. The quality of a software architecture plays an important role on the success of the whole system. This paper addresses a method of designing the software architecture based on process group component model, which integrates technologies of object-oriented, components and software reuse. Finally, a case study—iCALL system is given to show the application of this method.

Keywords Software architecture, Process group component model, Object-oriented, Components

1 引言

随着软件规模的增大和复杂度的提高,如何有效地进行软件系统开发这一问题变得十分突出。越来越多的研究事实表明,软件工程的有效性在于进行体系化或系统化的软件设计^[5]。系统软件体系结构的好坏对于最终系统的成败起着举足轻重的作用,所以对软件体系结构设计方法的研究变得十分重要。

虽然软件体系结构已经在软件工程领域中有着广泛的应用,但迄今为止还没有一个被大家所公认的定义。Marry Shaw 认为软件体系结构是软件设计过程中的一个层次,这一层次超越计算过程中的算法设计和数据结构。软件体系结构处理算法与数据结构之上关于整体系统功能设计和描述方面的一些问题,如全局组织和全局控制结构、关于通讯、同步与数据存储的协议,设计构件功能定义,物理分布与合成,设计方案选择、评估与实现等^[1]。

基于构件的软件复用作为一种提高软件生产率和软件质量的有效途径,是近几年来软件工程界研究的重点之一,被认为是继面向对象方法之后的一个新的技术热潮。近年来,在中间件技术的基础上,结合软件复用思想和面向对象方法,基于构件的软件开发(component based software development, CBSD)技术受到了高度重视。

对于并发系统,合理的进程结构是设计软件体系结构时需要考虑的首要问题。本文基于 CBSD 模型,提出了一种基于进程组构件模型的软件体系结构设计方法,应用这种方法设计并实现了 iCALL[®] 客户服务中心平台系统的软件体系结构。

2 进程组构件模型

2.1 基本概念

对象与领袖对象。对象是面向对象语言实现的、某个类的实例。领袖对象(Leader Object)是承担对成员对象(Member Objects)进行协调与管理的职能的对象。

构件是指系统中较为独立的功能实体。构件模型是面向构件的软件开发方法的核心,是构件的本质特征及构件间关系的抽象描述,它将构件组装成所关心的构件类型、构件形态和表示方法加以标准化,使关心和使用构件的外部环境(如使用构件构造出的应用系统、构件组装辅助工具和构件复用者等)能够在一致的概念模型下观察和使用构件^[3]。

在本文提出的设计方法中,将构件定义为由若干负担不同职能的对象组成、以既定的协议与外界进行交互并可进行组装的模块,并根据构件职能的不同,将其划分为如下几类:

- 基础构件:是一个模块,由一个领袖对象和多个(实现了相同或不同的职能细节的)成员对象构成。在不需要区分的场合,基础构件简称为构件。

- 领袖基础构件:是承担对成员基础构件(Member Components)进行协调与管理的职能的基础构件。

- 进程构件:是一个独立的进程,由一个领袖基础构件和多个(实现了相同或不同职能的)成员基础构件组合而成。

- 领袖进程构件:是承担对请求的监听、向成员进程构件进行请求分派的职能的进程构件。

2.2 进程组构件模型

采用 UML(Unified Modeling Language)记法体系来表示上文定义的构件,其组织结构如图1所示。

褚 华 博士生,主要研究领域为逆向工程,程序理解,面向对象技术和软件体系结构。陈 平 教授,博士生导师,主要研究领域为电子信息系统软件开发理论与技术,面向对象技术,软件工程,逆向工程。

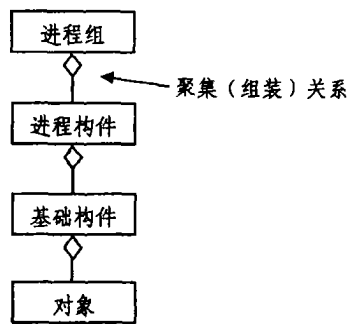


图1 构件组织结构

进程组是一组密切相关的进程,由一个领袖进程构件和多个(实现了相同职能的)成员进程构件组合而成。将每个进程组组织成一个构件,就构成了进程组构件,其概念结构如图2所示。

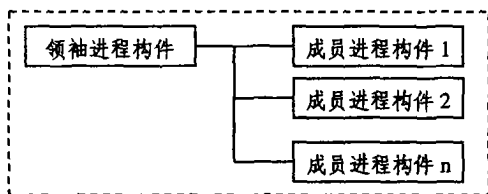


图2 进程组构件

领袖进程构件向成员进程构件进行请求分派时,有如下两种模式:

- 路径分派(Path Dispatching):在特定的进程组构件中,领袖进程构件监听到请求到达的事件后,将承载请求信息的路径分派给成员进程构件,后者既接收请求也提供服务。

- 数据分派(Data Dispatching):在特定的进程组构件中,领袖进程构件接收请求信息后,将请求信息分派给成员进程构件,后者只提供服务。

进程组构件之间采用构件交互协议进行通信。构件交互协议是构件之间既定的通信协议,并对通信方式(如,基于TCP的通信/基于消息的通信/报文通信/文件传送;直接通信/转发器通信;等等)及其实现细节(如格式转换、路由判定)进行了封装。构件交互协议支持进程组构件之间以TCP和消息池的方式进行通信,前者用于机间通信,后者用于机内通信。

本文的设计方法基于如图3所示的软件层次结构,不同层次中的特定职能由对象和不同种类的构件(基础构件、进程构件和进程组构件)组合完成。

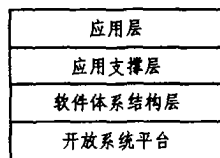


图3 软件层次结构

对象:被用来实现应用层、应用支撑层或软件体系结构层中特定职能的特定细节。

基础构件:被用来实现应用支撑层或软件体系结构层中的一种特定职能。其基本特征是:

- 可以作为流程图结点,按照既定的流程逻辑构成应用支撑流程图或应用流程图^[4]。

- 以一致的方式组织应用层对象,来实现应用层的一种或一组职能。

- 在其内部的对象之间,采用面向对象语言中的常规消息传递方式进行通信。

- 在基础构件之间,按照构件交互协议,通过具有构件交互职能的特定基础构件进行通信。

进程构件:被用来实现涉及应用层、应用支撑层和软件体系结构层的一组特定服务职能。其基本特征是:

- 不同的进程构件以并发方式运行;
- 进程构件可以独立运行,也可以从属于特定的进程组、与进程组内的其他进程构件协同运行;

- 在进程构件之间,按照构件交互协议,通过具有构件交互职能的特定基础构件进行通信。

进程组构件:被用来对一组进程构件进行组织和协同,主要用来提高并发处理能力和系统可靠性。其基本特征是:

- 同类进程组构件的数量是可配置的;
- 每个进程组构件中的成员进程构件数量都是独立可配置的;
- 不同的进程组构件以并发方式运行。

3 实例分析

iCALL[®]系统是我们自主开发的一个客户服务中心平台产品,该系统软件采用层次化结构,明确划分了系统各种成分的职能,统一了系统中各种成分之间的接口。iCALL[®]系统的系统组成如图4所示。图中的虚线表示话路通信,实线表示数据通信;深色矩形表示iCALL系统本身的部件,其它表示与iCALL系统相关的部件或客户服务中心的客户。

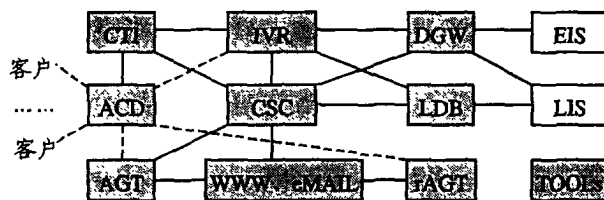


图4 系统组成

图5是基于进程组构件模型的软件体系结构图。图中白色矩形表示领袖进程构件,深色矩形表示服务进程构件;深色联线表示TCP通信,浅色联线表示消息池通信。

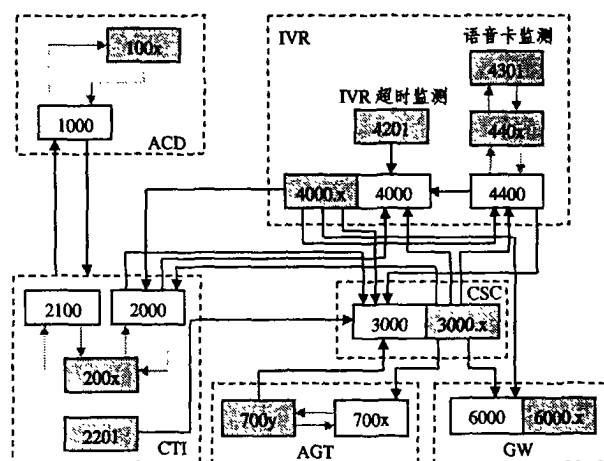


图5 软件体系结构图

(下转第137页)

接、重定向、提交和构建的关系;如果这些页面连接成环路,则需要人工干预,删除形成循环的较少参数的连接。集成测试的次序可以通过如下几个步骤得到:首先,根据测试模型,建立一个图,图中的节点代表页面和页面中的组件,图中的连接表示各节点的关系,每个服务端页面和所构造的客户端页面表示为一个单一的节点;其次,图中的每个连接可以有权值,权值与传递的参数个数成正比,若将页面装入框架则给予较高的权值;然后,通过删除较小权值的连接将该图转换为一个非循环图;最后,计算该非循环图的最逻辑次序,得到集成测试的次序^[5]。集成测试的次序建立后,就可以进行集成测试,每一次集成根据需要可以增加新的测试用例,或者继续使用上一步的测试用例,直到所有有关的页面被集成。

4.2 导航测试

Web 应用软件的导航行为非常复杂,为了确保导航行为是正确的,并且满足了用户的需求,必须进行导航测试^[7]。通常,导航测试要经过建立导航行为图、写出路径表达式、选择测试用例和执行测试用例四个阶段来进行。首先,通过分析测

试模型,可以建立一个导航行为图,该图中的节点表示页面、页面中的表单和框架,图中的每个边用 $e_i (i=1, 2, \dots)$ 来标识,表示节点间的超链接、提交、构建和重定向等关系,另外,还必须表示相关的关联类例如 creatcondition、loadpage-frame,图3是一个导航行为图的例子。接着,开始构造导航行为图的路径表达式,一般来说,路径表达式是图中所有路径的代数表示,表达式中的变量是边标签,表达式中的操作符有三个,分别为(),+和*,其中,()用于表示子表达式组,+用于路径的选择,*用于路径的循环^[8,9],图3的路径表达式就可以表示为 $e1e3e5 + e2e4e6e7(e8 + e9)$,在 serverpage1 处存在一个选择, $e8 = \{\text{serverpage1}, p5\}$, $e9 = \{\text{serverpage1}, p6\}$ 。然后,由于路径表达式直接表示了图中的所有路径,因此可以知道有多少独立路径、有多少选择路径和多少循环路径,并相应产生测试用例的节点序列,评估满足了何种覆盖标准。一旦测试用例从导航行为图的路径表达式中产生,测试工程师就可以手工输入需用表单收集的数据,并借助工具自动执行测试用例。

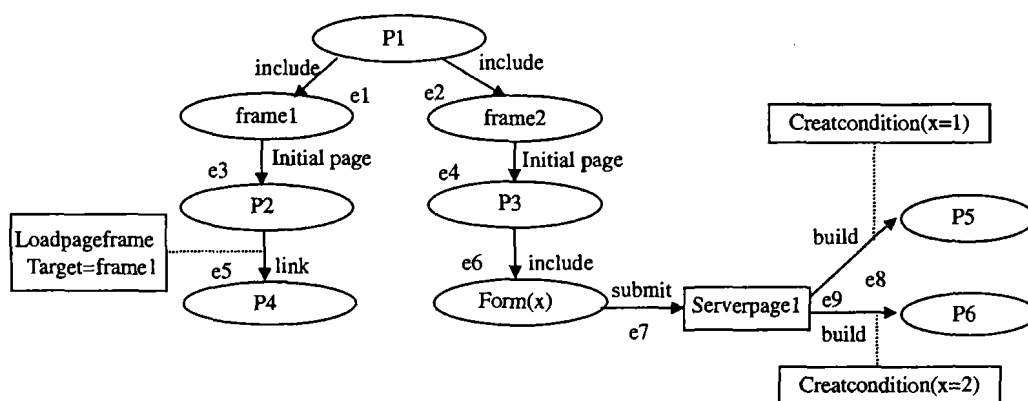


图3

结束语 随着 Web 技术的飞速发展和 Internet 的不断普及,Web 应用软件的需求不断扩大,应用领域也更为广泛,为了保证日益复杂的 Web 应用软件的质量和可靠性,使其具有良好的交互性、安全性和分布性,Web 应用软件测试将会扮演更加重要的角色,引起人们更多的关注。本文首先描述了 Web 应用软件的 RE 分析方法,接着提出了一个 Web 应用软件测试模型,然后详细讨论了 Web 应用软件单元测试、集成测试和导航测试的策略和方法。

参考文献

1 许蕾,许宝文,陈振强. Web 测试综述. 计算机科学,2003,3

- Lucca G A D, Penta M D, Antoniol G. An approach for Reverse Engineering of Web based application. IEEE Computer, Jan. 2001
- Conallen J. Modeling Web Application Architecture With UML. <http://WWW.conallen.com>.
- Conallen J. Building Web Application With UML, Addison Wesley Publishing Company, Reading, MA, 2000
- Lucca G A D, Fasolino A R, Faralli F, Carlini U D. Testing Web application. IEEE Computer, Feb. 2002
- 郑人杰. 计算机软件测试技术. 清华大学出版社, 1999
- Kung D C, Liu C H. An Object-oriented Web test model for testing Web application. IEEE Computer, Jan. 2002
- Ricca F, Tonella P. Analysis and testing of Web application. IEEE Computer. July, 2001
- Tonella P, Ricca F. Dynamic Model Extraction and Statistical Analysis of Web Applications. IEEE Computer, April, 2002

化描述。

参考文献

- Shaw M, et al. Abstractions for Software Architecture and Tools to Support Them. IEEE Transactions On Software Engineering, 1995, 24(4): 314~335
- Hayes-Roth B, et al. A Domain-Specific Software Architecture for Adaptive Intelligent Systems. IEEE Transactions in Software Engineering, 1995, 21(4): 288~301
- 梅宏, 陈锋, 冯耀东, 杨杰. ABC: 基于体系结构、面向构件的软件开发方法. 软件学报, 2003(4)
- 李青山, 陈平, 褚华. 支持柔性机制的元数据模型的研究与应用. 西安电子科技大学学报, 2002(3)
- 于卫. 软件体系结构与软件重用方法的研究. [学位论文]. 西安电子科技大学, 1999
- Taylor R N, Medvidovic N, Anderson K M, et al. A Component-and Message-Based Architectural Style for GUI Software. IEEE Transactions on Software Engineering, 1996, 22(6)

(上接第119页)

系统的每一个组成部分都由一个或多个进程组构件构成,每一个进程组构件被赋予一个全系统内唯一的标识,我们称之为进程组标识。其中进程组构件3000、4000、6000采用路径分派方式;进程组构件2100、4400采用数据分派方式。进程组构件之间通过构件交互协议进行通信。

结束语 软件体系结构和基于构件的软件开发是近年来软件工程界关注的重点。本文介绍了一种基于进程组构件的软件体系结构的设计方法,并简要介绍了应用此方法开发的 iCALL 客户服务中心系统。

进一步的工作包括如何从更高的抽象层次来完善基于进程构件的设计方法,给出各种构件以及构件交互协议的形式