

⑤ 293-296, 300

一个基于 Client/Server 结构的 CASE 对象模型

朱 军

TP 311.51

(中国科学院软件研究所, 100080, 北京; 24岁, 男, 博士生)

摘 要 提出了一个基于客户机/服务器结构的 CASE 参考模型。使用这个模型, 集成化的 CASE 环境将随着它所支持的企业、项目和人员的不同表现出适应性、灵活性和动态性。使用对象建模技术, CASE 模型在提供了可重用构件的同时, 还提供了可重用的体系结构。

关键词 计算机辅助软件工程; 客户机/服务器结构; 重用; 面向对象

分类号 TP31

CASE

早期的 CASE 侧重于对软件开发过程中的各种活动孤立地进行辅助, 如需求分析工具、设计工具、编码和调试工具、测试工具等。由于大量的工具是独立研制的, 导致彼此之间联系松散, 甚至互不兼容, 因而如何构造能够支持软件生命全周期, 并适合不同层次软件技术人员使用, 支持包括技术和管理等活动的集成化 CASE 环境, 成为目前 CASE 研究的热点。

目前看来, CASE 的发展仍未彻底解决软件的工程化问题。其主要原因是由于尚未发现一种完全有效并在实际应用中广泛接受的软件工程方法学。结构化方法从理论上讲是完整有效的, 但真正在实际系统开发中的可操作性却不尽人意。从操作角度讲, 方法学的不完善是制约 CASE 生存和发展的关键因素。现行的各种软件开发“方法”是否能适用于整个软件生命周期尚未定论, 因而, 作为“技术”和“工具”的 CASE 环境目前不仅需要支持已有的软件开发方法, 更需要为将来的软件开发方法嵌入 CASE 环境提供一个易于扩充的接口。如何在集成性和灵活性(即易扩充性)之间取得平衡, 建立一个开放式的体系结构, 以支持多种方法的自动化, 正是本文所要解决的问题。

从发展趋势看, 90年代开放式环境下的 CASE 应具有以下特点: ①支持所有的角色和阶段; ②多种方法的自动化; ③可以集成“第三方”工具; ④所有工具共享一个公共的开放式数据库; ⑤提供了复杂的设施, 用以检查工具之间的数据一致性。

1 Client/Server 体系结构

随着软件系统规模和复杂度的日渐升级, 软件体系结构的好坏对整个软件系统的设计与开发显得至关重要。就体系结构的本意而言, 它是指建立系统时的构造范型、构造风格和构造模式, 软件体系结构对于软件系统的构造所起的是指导性的作用, 它抛开了软件系统的功能细节, 着重于探讨软件系统的总体框架。广义而言, 软件系统结构主要包括: ①软件的成分及系统框架; ②软件成分的选择, 各成分之间的相互作用, 软件成分的进一步复合以及指导软件复合过程的总体模式; ③系统的功能、性能、设计以及从多种方案及选项中进行选择的决策; ④软件体系结构更为关注的是产品及其成分, 而方法论更关心建立产品的过程等等。

一般地, 采用 Client/Server 体系结构应遵循如下准则: ①在 Client 与 Server 之间应有清晰的功能划分, 各自均应有具体的功能角色, 但从应用的观点看它们之间的交互应是无缝的; ②Client 与 Server

可以运行在不同的机器上,但这不是必须的,这样有利于应用的可伸缩性;③一个服务器应能并发地支持多个客户,这意味着服务器对客户来说是一个共享的“资源”;④对服务器的改变不应对客户产生影响,这对系统的可伸缩性和可重用性都非常重要;⑤对一个客户的修改不应影响到其他的客户,这意味着客户之间的独立性。

对于数据类型繁多,数据结构复杂的 CASE 工具来说,建立一个可共享的、公用的、开放式的数据库尤为重要。采用具有 Client/Server 结构的数据库管理系统,数据的表现和数据的存取被分别作为 Client 与 Server 的职责,这不仅使 CASE 工具间的数据集成得以保证,也使得 CASE 环境更易于扩充。

2 CASE 模型的 Client/Server 结构

基于上述思想,我们对 NIST/ECMA 建立的集成 CASE 参考模型稍作改动,得到如下适用于开放式环境的 CASE 模型结构:

按照上述 Client/Server 体系结构的划分准则,我们将图 1 所示的 CASE 模型划分成如图 2 所示的 CASE 环境体系结构。由图 2 可以看出 CASE 软件成分的划分如下:

(1)Server 端主要由采用 Client/Server 计算技术的数据库组成,这是一个可共享的,公共的开放式数据库系统,其功能是为 CASE 环境建立一个统一的数据仓库,提供优化的存取手段,并设置必要的管理方法。

(2)Client 端主要由支持各类方法的分析,设计工具及项目管理工具组成,它们通过 Server 端提供的查询语言接口与 Server 相联系,其主要功能是按照特定方法的语义去解释端用户的动作。在 Client 端用户界面与应用间可利用回调技术提供的应用接口进一步分离。

采用 Client/Server 结构,我们可以得益于以下几个方面:①适合建立一个开放的系统环境,我们可以在不同的软、硬件平台上建造适合于自己的 CASE 工具,并可通过分布环境建立工作组(包括项目管理人员,分析人员,设计人员等)协作环境;②可以减少软件的维护费用,由于 Client 和 Server 是相对独立的,因此,在软件系统的开发及维护过程中,可独立进行,一方的修改将不必导致另一方的修改;③有利于 CASE 工具的开发,集中软件方法中的语义处理,而不必过多考虑 Server 端存储数据的内在表示;④可以充分利用工作站的性能,提高系统资源的利用,把更多的事务运行在成本较低的平台,则可大幅度降低计算成本;⑤一般可以通过对系统的共享资源,如服务器机和网络等卸载的办法来提高系统的可伸缩性。

3 CASE 模型的对象设计

采用对象建模技术,我们将 CASE 模型中的软件成分抽象成不同的对象,在此基础上,利用继承和聚合等技术,建立可复用的构件,更进一步通过各抽象基类中定义的抽象操作预定义 CASE 工具中的一些消息传递序列,从而建立一个通用的 CASE 框架,提高它的复用程度。下面我们将 CASE 模型在体系结构划分的基础上对象化,并建立 CASE 的对象模型图。

从图 3 建立的对象模型中我们可以看出,整个 CASE 环境由下列几大对象类组成:①CaseUI 类:这是 CASE 环境中的用户界面类,它实现了 CASE 模型中用户界面的表示集成。每个 CaseUI 对象都由多个 UIO(用户界面对象)聚合而成,其中每个 UIO 都是一个独立可复用的界面元素;②ProjectManager 对象:这是 CASE 环境中用以完成项目管理及进程控制服务管理对象,它主要由各种实现自动管理的工具聚合而成;③Method 类:这是 CASE 环境中具体实现的各种开发方法的抽象基类,预定义了各种

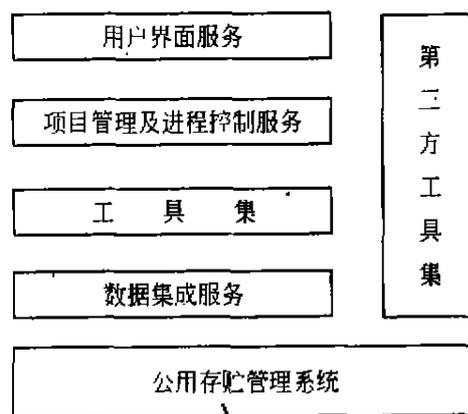


图 1 开放式环境下 CASE 模型

Fig. 1 A CASE Model under Open Environment

开发方法所具备的一般开发步骤；④OODBMS 对象：这是 CASE 环境中一个负责控制、存取数据的面向对象数据库管理系统，它采用了 Client/Server 体系结构。

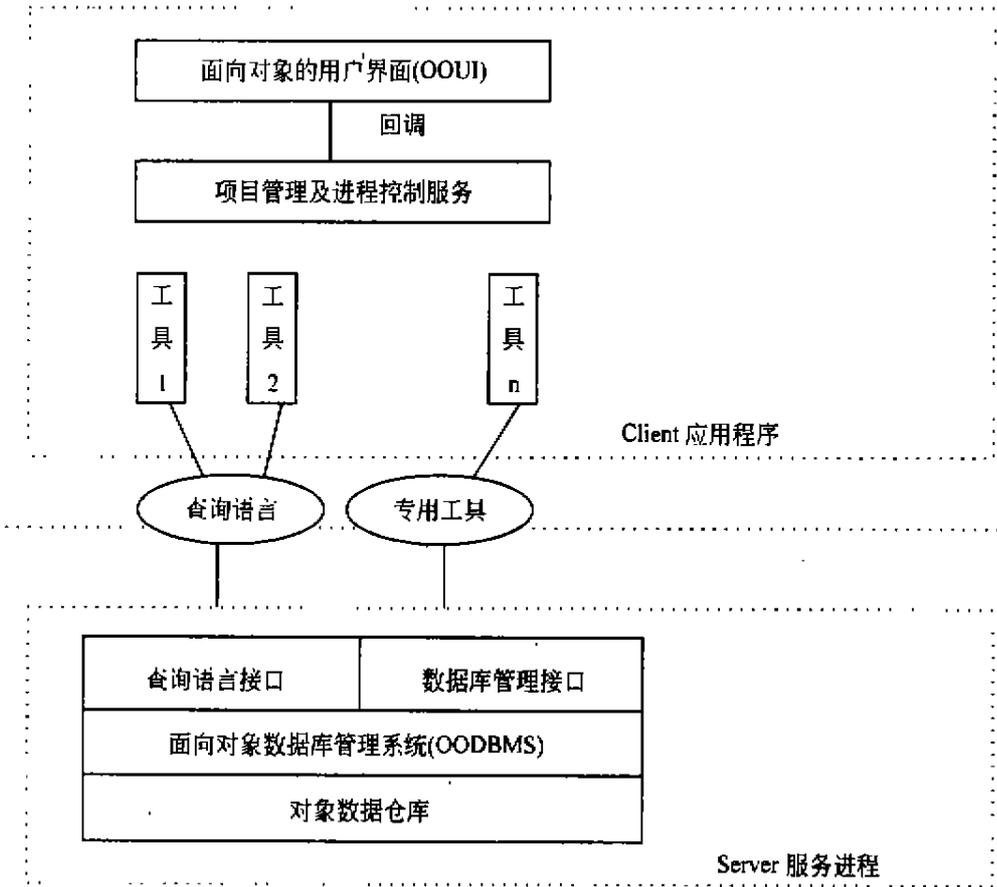


图 2 CASE 体系结构参考图

Fig. 2 The Reference of CASE Architecture

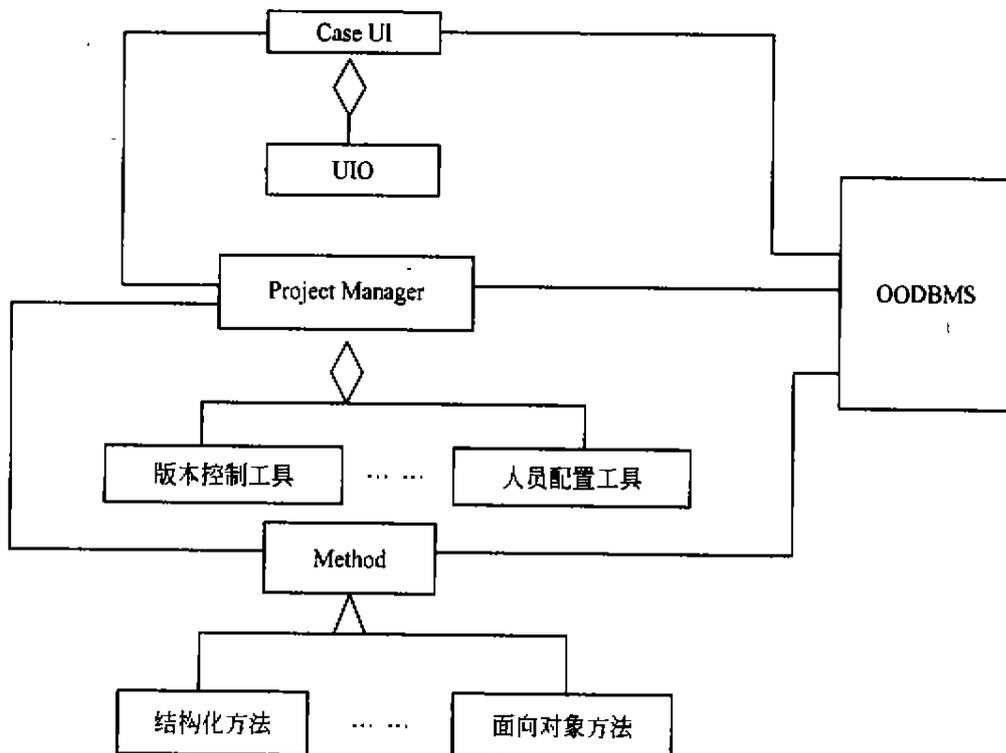


图 3 CASE 对象模型图

Fig. 3 Object Model of the CASE Tool

CaseUI 类、Method 类和 ProjectManager 类之间的联系是多对一的,即由一个 ProjectManager 对象管理多个 CaseUI 对象和多个 Method 对象,同时这三个类于 OODBMS 类之间的关系是多对一的,即大家共享同一个数据库。

4 评价

CASE 正处于一个迅速发展的阶段,集成 CASE 环境的设计和选择主要是为了便于适合于不同方法、不同生命周期的各种 CASE 工具更容易结合起来。为了适应明天 CASE 的发展需要,CASE 环境必须在其集成度和灵活性之间保持某种平衡,过度集成化往往意味着开放程度的不够,而将 CASE 建立在 Client/Server 体系结构上,既使得多种 CASE 工具共用服务器上开放数据库而易于集成,又使得各部门可根据自己的需要在客户端灵活地建立自己的 CASE 环境,这就是所谓的开放式 CASE 环境。

下面我们从技术框架上去分析一下图 2 所示的 CASE 体系结构的特点。

4.1 集成性

通常,集成环境表现在数据集成、控制集成和表示集成这 3 个方面。而图 1 中所示的几种服务则从技术上保证了这 3 个方面的实现。

4.1.1 数据集成,这是公用存贮管理系统及数据集成服务所支持的。共享设计信息是工具集成的关键。本文中我们选择了一种基于信息仓库的传输。这种方式较适于紧耦合的一致性环境。一个数据仓库可以提供许多基本服务,包括对象/实体和连接/联系的存贮和管理,版本和配置控制,命名服务,安全性检查和传输控制等。采用面向对象的数据库作为信息仓库可提供如下的数据集成服务:①元模式服务,用以定义、控制和维持一个元素模式;②查询服务,支持数据仓库中元模式和元数据的检索;③视图服务,允许开发者定义数据仓库中一个对象和操作的子集并在总体环境与子环境之间维护其一致性;④内部数据交换服务,用以支持仓库数据与其他类型(如文件)数据间的双向传输。

4.1.2 控制集成,这是由项目管理及进程控制服务所支持的。CASE 环境下的各工具必须能够互相发送消息,在适当条件下激活其他工具,以达到共享功能。控制集成机制包括明确的消息传送、时间触发和事件触发。本文所示的 CASE 模型中,项目管理及进程控制服务可以明确各工具触发的序列和策略,使得端用户避免去了解不同工具的细节,而是把注意力放在软件开发方法的抽象概念层上,进一步的控制集成将有利于软件开发自动化程度的提高,并有可能支持软件开发的整个生命周期及项目级管理。

4.1.3 表示集成,这是由用户界面服务所支持的。在本文所示 CASE 体系结构中,面向对象的用户界面所提供的界面服务使 CASE 用户可以对不同 CASE 用户使用同样的方式交互,这样新的工具将更易于使用。基于窗口的工具通常有四种不同层次上的表示集成:窗口系统,窗口管理器,用户界面开发工具和风格指南。通过面向对象方法,我们将界面的对话控制,表示元素及应用接口定义在同一个 UIO 类中,不仅在用户界面开发工具上有效地体现了表示集成,而且更利于软件的重用。

4.2 开放性

本文提出的 CASE 体系结构的开放性是体现在可共享的、公共的、开放式数据库系统上的。由于表示外部信息的内部数据管理仓库是开放的,因此,第三方的 CASE 工具或系统可以加入现有工具集,并可进行数据交换,也有利于多方协同工作。另外,这里的开放性是建立在数据集成之上的开放,因而是可行的,无限制的开放将导致系统的复杂度过高而难以实现。

4.3 复用性

为了提高软件开发的质量,缩短软件开发的周期,软件成分及软件结构的复用率愈来愈被人们所重视。本文提出的 CASE 环境体系结构是在确定了 CASE 含义之后,针对 CASE 的现状 & 未来的发展趋势提出的,在设计中利用面向对象方法预定义了各对象之间的联系,建立了一个通用的框架。通过对这个框架的复用,我们可以减少重复劳动,使产品的质量得以保证并使产品易于维护,易于测试。

(下转第 300 页)

A Study on the Gas Sensing Properties of Thick Film SnO₂ Materials

Hou Yuqing

(Department of Electronics Science, Northwest University, 710069, Xi'an)

Abstract The Special SnO₂ materials were prepared in chemical method. By doping Pb, these SnO₂ mixed with other materials made the mixture for thick film. The SnO₂ thick film gas sensing parts were made in Screen-printing method. The properties of the gas sensitive film were studied. The results show that they have a high sensitivity for alcohol-gas.

Key words tin-oxide; thick-film; gas sensing part; alcohol-gas

(上接第 296 页)

5 应用

在用 C++ 建立了这个 CASE 对象模型的框架结构后,我们先后在其支持下完成了面向对象的分析与设计工具 OmtTool 和基于 Petri 网的半形式化面向对象开发工具 CpnTool,并分别通过鉴定和验收。与会专家一致认为,该项技术达到 90 年代国际先进水平。进一步,我们将完成在框架支持下的多种 CASE 工具并逐步建立开放式 CASE 环境。

参 考 文 献

- 1 黄日日,杨美清.对计算机辅助软件工程的若干看法.计算机科学,1995,22(2):38~42
- 2 车敦仁,周立柱,杨亚奇.Client/Server 计算与现代 DBMS 的体系结构.计算机科学,1995,22(5):26~33
- 3 Gane C. Computer-Aided Software Engineering. New Jersey, Prentice-Hall, 1990. 1~105
- 4 Berson A. Client/Server Architecture. New York, McGraw-Hill, 1992. 1~99
- 5 Kroha P. Objects and Databases. New York, McGraw-Hill, 1993. 1~33
- 6 Zand M. A Survey of Current Object-Oriented Databases. The Data Base for Advances in Information System, 1995, 26(1): 8~15
- 7 Chen M. A Framework for Intergrated Case. IEEE Software, 1992, 41(3): 35~42

责任编辑 曹大刚

A Case Model Based on Client/Server Architecture

Zhu Jun

(Institute of Software, Chinese Academic of Scinces, 100080, Beijing)

Abstract CASE technology has made significant advances recently, but its potential is limited by integration difficulties. There is as yet no coherent strategy for building and using integrated CASE. In order to solve this problem, a reference model is proposed on the basis of the Client/Server architecture. Using the model, an integrated CASE environment will be as adaptable, flexible and dynamic as required by the enterprise, projects and people it supports. Using object model technology, the CASE reference model supplies not only the reusable components, but also the reusable architecture.

Key words CASE; Client/Server architecture; reuse; object-oriented