

多层应用技术建立信息系统

傅效群 (解放军总医院计算机室 100853)

摘要:本文首先介绍信息系统的发展过程,讲述多层技术的优点,并以医院信息系统为例讨论了采用 DCOM 技术,利用 Delphi 3.0 工具建立客户端'0'配置的应用。

关键词:多层应用 DCOM 医院信息系统 ActiveForm

一、信息系统的发展

今天随着计算机硬件、网络技术的发展,信息系统也朝着这更加实用,更加方便的方向发展。信息系统发展可分为三个阶段。

1. 单层的主机/终端方式

单层应用的结构是基于将所有的部件都在一个运行在单一主机上程序中。主机/终端方式的系统技术封闭,所以整个系统有很好的安全性,但用户界面不友好。

2. 二层的 C/S 方式

随着 LAN, PC 机的广泛应用, RDBMS, RAD 技术的成熟,二层的 C/S 应用得到了大量应用。Client 端一般是使用 GUI 的 Windows 或 MAC/OS 操作系统, Servre 端运行 UNIX 或 WINDOWS NT。与单层的主机/终端方式相比,这种开放式的体系结构的耗费大大降低,分布式的计算结构充分利用整个系统的资源,Client 端的 GUI 使用户操作更加方便, RAD 使开发人员能够快速开发各种应用。C/S 系统也存在一些很难解决的问题,首先是用户较多的系统的安装和升级十分困难的,维护工作量大。其次,系统封闭,不同的系统之间交流困难,已有系统难以继续使用,造成重复投资,而且不同的系统用户界面也不统一,培训的工作量也较大。以医院信息系统的开发为例,现在较成熟的应用一般采用 C/S 技术开发的,但在开发和使用过程中也暴露出许多问题,最突出的就是系统维护困难。系统运行的过程中发现的错误要及时更正,这就需要操作人员到现场更新版本。对于应用点较少的系统,如病案编目只在病案室用,更新还比较容易,但对那些应用点非常多而且分散的护士工作站,门诊收费系统更新版本的工作量就很大了。管理模式的改革,会使系统的一些关键的参数,算法和公共模块的改变从而整个系统就要重新编译和发布,工作量那就更大了。据统计,由于采用两层 C/S 结构,每年全世界有约 53%

的软件开发项目以失败告终。

3. 三层的 C/S 结构

因为 C/S 系统的种种不便,信息系统又向着多层应用发展。现在多层应用一般采取三层 C/S 结构的形式。

三层 C/S 应用将数据库应用分成三个逻辑单元,每个单元一般是运行在相互连接独立的计算机上,通过计算机网络互相共享数据和通信,三个单元是:

- Client Application 主要完成数据显示用户界面的功能。

- Application Server 集中管理业务规则和 Client 与 RDBMS 的数据交换 也叫做数据代理。

- RDBMS 提供数据的存储和管理功能。

三层 C/S 应用的优点

(1) 系统维护变得容易,三层 C/S 应用的 Client 端只处理用户界面代码量减小,无须配置 DBMS 的 Client 端程序,甚至可以利用 INTRANET 技术做到 Client 端的'0'配置。而且业务逻辑的变化,只要更新中间层就可以完成,不必对整个系统更新。这是三层 C/S 应用的最主要的优点,尤其是对大型信息系统,使用多层技术可以大大降低系统的维护费用。

(2) 对象可重用,每个三层 C/S 应用,都是由 Client 端和 Server 端的模块组成的。这些模块通过标准的抽象界面,连接在一起,完成一个完整应用。每个模块都是可以被其他应用调用的,可共享、可再用的对象。

(3) 更高的开发效率,擅长开发用户界面的程序员可以专心开发用户界面,较少的关心业务逻辑。业务逻辑分析者可以专心开发业务算法规则,而不必关心数据是如何表现的。选择工具也比较自由,这样可以缩短应用开发周期,提高软件质量。

(4) 分布计算的基础结构,三层 C/S 应用可以更充分的利用系统资源,在大型的联机应用中,数据库面临的客户数量是非常庞大的,使用传统的客户/服务器模式可

能根本无法胜任。例如,可能有上千个客户机在同时运行,需要访问数据库。如果它们的请求都直接传递到数据库服务器上,就必须要有非常强大的硬件支持。通过中间层的缓冲,连接数据库的用户大大减小。

5. 系统更加安全, Client 端不直接与 DBMS 相连, 通过中间层可实现更加健全的安全机制。

二、DCOM 技术

利用多层应用技术构建新一代医院信息系统就可以克服上面提到的 C/S 系统的弊端。多层系统主要利用分布式对象技术来实现, 业界的对象分布标准也日趋成熟, Microsoft 公司提出的 DCOM, OMG 定义的 CORBA 以及 Sun 公司的 JavaBean, 这些组件标准都可以很容易地实现中间层的功能, 而且放在任何层次的对象都可以很容易地访问其他层次上的对象。CORBA 和 JavaBean 都提供跨平台技术, DCOM 是在 WINTEL 平台开发的, 国内的计算机大多采用 WINTEL 平台, 一般不需要跨平台技术, 采用 DCOM 是比较合适的选择。

1. DCOM 简介

DCOM(Distributed Component Object Model)即分布式组件对象模型, 是微软在 COM 组件基础上增加开放性软件 DEC 远程过程调用协议的基础上开发的。在可视化编程环境下, 编程者可以利用已有的组件(如 VBX, OCX)构建自己的程序, 但这些组件都是运行在本地的, 缺少分布式计算的能力。采用 DCOM 编程技术, 就可以将不同的组件放置到网络的不同位置, 这些组件在网络上协同工作, 共同完成应用。

2. DCOM 的应用

利用 DCOM 开发多层医院信息系统应用, 主要要编写 Client Application 层和 Application Server 层 DCOM 部件。DCOM 支持现有的 TCP/IP, HTTP, IPX/SPX 等网络标准协议, 可以采用多种开发语言如 JAVA、C、PALCAL 等, 这些开放的标准技术使得 DCOM 的开发成本较低, 程序员可以根据自己的爱好选择开发语言。Application Server 层 DCOM 部件主要完成业务逻辑功能也就是对数据库的查询和修改, 如在医院信息系统中的平均住院日的计算, 病人的入出院处理。这些功能用 DCOM 的开发语言和嵌入 SQL 共同完成。Client Application 层的 DCOM 主要要编写用户界面代码, 与 Application Server 层 DCOM 部件交换数据, 完成数据的录入和表现。

网络应用必须考虑到系统的安全, DCOM 把基于认证的 INTERNET 安全机制同基于 NT 的 C2 级的安全

机制集成在一起。在 NT 环境下, 对每个 DCOM 组件都可以象控制文件系统一样利用访问控制列表(access control lists)控制用户或用户组的访问。传统的数据库中由于用户数量大, 为便于管理, 需引入角色管理机制。用户组的概念就象传统的数据库中角色概念一样, 医院中尽管用户数量很大, 但用户所属的角色即用户组非常有限, 如医生、护士、收款员等。通过对用户组进行授权, 可以对该组的所有用户进行授权。

3. DCOM 组件的并行发布

当很多用户同时使用一个 DCOM 组件, 如 Application Server 层的收费项目录入, 科室名称和代码的对照等业务逻辑就会产生系统的瓶颈, 影响系统效率, 这就需用对关键的 DCOM 组件进行并行发布, 即可在多台服务器上运行相同的 DCOM。当客户端调用这些 DCOM 时, 解决调用平衡可采用两种方式: 一种是静态的调用平衡, 即指定某些用户调用指定的 DCOM 服务, 显然, 这种方法对系统资源利用率是低的, 而且要对同一系统的各个客户端做不同的配置; 另外一种方法是动态的调用平衡, 当客户端的 DCOM 要调用 DCOM 服务时, 先与公共的负责调用平衡的 DCOM 通信由它返回所能连接的 DCOM 服务器的地址。这样做使系统的容错性也增强了, 当某个 DCOM 服务发生意外失败, 系统可利用在其他服务器上的相同的 DCOM 来代替。DCOM 有灵活的连接管理技术, 用一计数器来纪录有多少 CLIENT 与 SERVER 相连, 通过智能 ping 技术来确定连接的有效性, 当计数器为零时应用会自动关闭, 需要时 DCOM 会自动启动应用。这些技术可以更有效的利用系统资源。

4. DCOM 部件的调用

DCOM 部件存放在服务器端, 客户机要使用无须下载到本地。使用 DCOM 部件构成的大型程序, 可以把处理相同工作的部分分割出来交给一个专门的软件模块完成, 而其他程序或其他 DCOM 部件只需对其进行调用, 即可获得所需信息。这和传统的模块化程序不同, 一个 DCOM 部件的修改并不需要重新编译其他 DCOM 部件, 有效地提高了整个系统的灵活性。比如在院病人查询的 DCOM 部件, 护士工作站、综合查询等系统都要用, 如果要修改这个部件来改变病人信息的输出格式, 其他系统不需要做任何修改。DCOM 组件支持多界面, 这个界面也就是 DCOM 对其他 DCOM 提供的方法和属性。这就为系统升级提供了便利, 因为在企业级应用中很难对系统进行全面升级, 需要在一定时期内支持新旧两种系统, DCOM 就可以用不同的界面支持不同的系统。

5. 其他具体问题

在用 DCOM 构造系统时,有些问题要具体考虑。Application Server 层的 DCOM 需要很好的总体规划设计,这是系统成功的关键。各个 DCOM 部件所完成业务规则,一定要做细致的分析和研究,既要考虑系统的效率,又要兼顾系统的灵活性。协同工作的 DCOM 组件间要有较宽的通信带宽,这样的系统才能高效稳定的运行。组件的大小对系统也有着不同的影响,较小的 DCOM 使系统有更大的灵活性,但增加了系统通信负担;较大的 DCOM 减少了系统通信负担,但丧失了系统的灵活性。如果不做各个 DCOM 部件所完成业务规则的总体规划,在系统开发时就会做许多重复工作,而且造成同一业务规则实现的不一致,产生错误数据。

三、利用 Delphi 3 创建 DCOM 控件,实现多层应用

DCOM 的这些特点为企业级信息系统的多层应用提供了良好的技术支持,利用 DCOM 技术构造多层应用是比较好的选择。在具体选择工具时可选用 Delphi 3.0、VC++ 5.0、VB5.0。Delphi 3 针对这种新一代的数据库管理系统观念提供了 ActiveForm 技术适应了 Intranet 应用。ActiveForm 是一种 ActiveX Controls,它可以完成 Delphi FORM 的各种功能,如对用户事件的响应,控制窗口中的控件,即 C/S 系统中窗口的功能都可以实现。作为 ActiveX Controls,它也需要 ActiveX Controls 的容器(如 IE, VB)来运行它。这就可使我们将 Client Application 做成 ActiveForm 放在 WEB 服务器上,在客户端用 IE 浏览含 ActiveForm 主页来下载运行 ActiveForm,这样就能真正做到客户端“0”配置,降低系统的维护费用。

Delphi 3 还使用三种以 DCOM 技术为基础的 Broker 和新一代的数据库引擎,来适应这种多层应用需求。

(1)Remote Data Broker: Delphi 3.0 使用 Remote Data Broker 实现了瘦客户机技术,Remote Data Broker 完成与 RDBMS 的通信。一个或多个 Remote Data Broker 组成 DCOM 构件,客户机通过标准的 DCOM 连接手段与 Remote Data Broker 交换数据。Remote Data Broker 智能的传递数据,降低网络负荷,只有被 Client 端修改的数据才传回 DBMS,在向客户端发数据时不仅包括 DBMS 数据而且包括业务规则,使应用在 Client 端完成大部分操作。

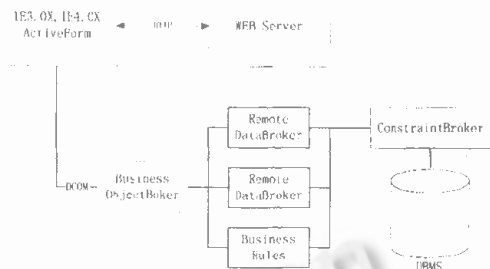
(2)Constraint Broker: Constraint Broker 主要完成数

据校验的工作。在三层 C/S 应用中数据校验工作可在 DBMS, Application Server, Client 端处理。

①Client 端:校验规则在 Client 端执行,可以减少网络流量,不必将数据传到 Server 端才知道数据是否合法。缺点是维护困难,当校验规则变化时,需要对每个 Client 端更新。②DBMS 端:与在 Client 端校验相反大量的未校验的数据传到服务器增加了网络流量,但是维护方便。③Application Server 端:与在 DBMS 端校验相似,只是校验规则书写简单,不针对某个 DBMS。Constraint Broker 很好的解决了上面的问题。Constraint Broker 自动的将校验规则从 Server 端传到 Client 端,即减少了网络流量,又做到了数据的一致性。

(3)Business Object Broker: Business Object Broker 可以智能的平衡中间层的使用,也就是 DCOM 的调用平衡问题。

利用 Delphi 3.0 构建多层应用的结构如下



利用 DCOM 技术和 Delphi 3 工具创建多层应用已十分方便,需要指出的是 WINDOWS 95 环境下不直接支持 DCOM,需要安装 VC, VB 带的 DCOM 支持工具(DCOM95.EXE, DCOM95CFG.EXE), NT4.0 和 WINDOWS98 都支持 DCOM 技术规范。各个含在不同 HTML 页面 ActiveForm 不能象在 C/S 系统下各个窗口可以方便的互相引用,一般需要通过中间层或本地文件来交换数据。

参考文献

- [1] MICROSOFT DCOM Technical Overview [http://www.microsoft.com]
- [2] Building Scalable Application with Windows NT and DCOM [http://www.borland.com/midas/dcom/dcom.html] (来稿时间:1998年6月)