

# Design and Implementation of Food Order Message System Based on

# COM



## 基于 COM 的订餐消息服务系统 设计与实现

宗平 朱辉 (南京河海大学计算机与信息工程学院 210098)

摘要: 文章介绍了某快餐连锁店电话订餐系统中消息服务软件的设计思想, 实现方法及关键技术, 特别是怎样利用组件技术较好地实现用户需求。

关键词: 消息服务 Socket COM Events

### 1 引言

随着社会的高速进步, 生活节奏的加快, 快餐业也迅速地发展起来, 并且已逐渐为人们所接受。由于城市化规模扩大, 人口增多, 必须有一套高效稳定的系统将订餐信息快速准确地发出。传统的订餐模式是通过人工转接调度, 不仅效率低下, 而且差错率高, 已远远不能满足公司发展需求, 所以快餐企业的信息化建设成了当务之急。

本文介绍常州某快餐订餐服务系统的设计与应用。顾客通过订餐热线, 将订餐的规格、送餐地址告诉接线中心的接线员, 接线员生成一分订单, 通过网络传递到调度中心, 调度中心再根据送餐地址, 将订单分配到各个地区。整个过程是通过网络完成, 其中分公司是通过拨号上网连接到调度中心局域网中。随着公司业务的发展, 分公司通过拨号到公司的局域网有一定的局限性, 维护工作大, 故障频繁, 费用较高, 所以决定对系统进行升级。

### 2 实施方案

Socket 是建立在传输层协议(主要是 TCP 和 UDP)上的一种套接字规范, 最初是由美国加州

Berkley 大学提出, 它定义两台计算机间进行通信的规范(也是一种编程规范), 套接字屏蔽了底层通信软件和具体操作系统的差异, 使得任何两台安装了 TCP 协议软件和实现了套接字规范的计算机之间的通信成为可能。

微软的 Windows Socket 规范(简称 Winsock)对 Berkley 的套接字规范进行了扩展, 利用标准的 Socket 的方法, 可以同任何平台上的 Socket 进行通信; 利用其扩展, 可以更有效地实现在 Windows 平台上计算机间的通信。在 Delphi 中对 Windows Socket 进行了有效的封装, 使得用户可以很方便地编写网络通信程序。这使得使用 Delphi 中的网络通信组件来实现消息服务器与各分公司之间的信息成为首选, 特别是当通信的双方只需要传送简单的消息而不需要进行大量数据的传送或复杂的交互通信时。

由于该企业业务发展较为成熟, 工作人员对原系统较为熟悉, 特别是接线员对界面操作的熟悉程度是工作效率的关键。如果全部升级系统势必影响工作, 且不得不重新对接线员进行长时间的培训。所以考虑在客户端采用 COM 组件实现与服务器的通信和订餐消息发送接收, 而不改变

当前的客户界面。COM 组件是遵循 COM (Component Object Model) 规范编写, 以 Win32 动态链接库(DLL)或可执行文件(EXE)形式发布的可执行二进制代码, 能够满足对组件架构的所有需求, COM 兼容组件提供了一系列的接口, 允许客户通过这些接口来调用相关的对象属性、方法和事件。这可以对信息进行有效地封装和隐藏, 便于将来升级和维护。

### 3 系统设计

#### 3.1 总体结构

每个地区设有一个调度中心, 若干分公司, 通过 Internet 连接消息服务器。调度中心, 即接线中心主要负责消息录入和发送, 分公司负责接收信息并响应发送成功标志, 消息服务器负责所有消息转发和用户管理。系统结构如图 1。

具体工作过程如下:

- (1) 调度中心和各分公司连接到消息服务器。
- (2) 将登录消息体发往消息服务器
- (3) 消息服务器验证身份, 返回登录成功标志
- (4) 消息服务器将在线公司列表发给所有在线公司

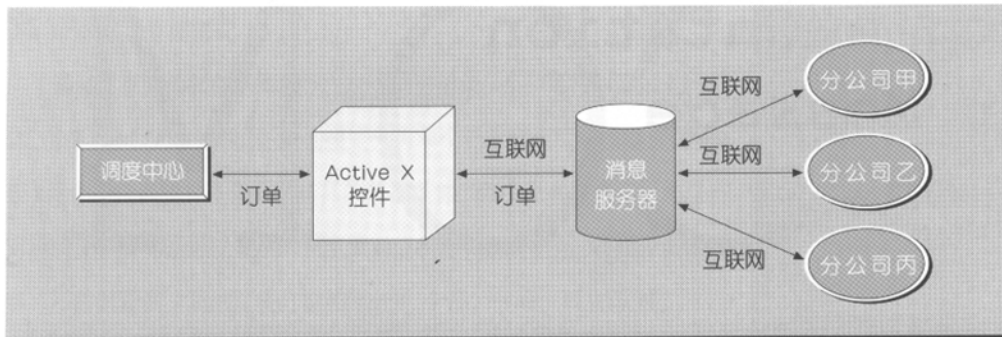


图1 系统框图

- (5) 调度中心将订单发送到消息服务器。
- (6) 消息服务器将订单发送给分公司。
- (7) 分公司收到订单消息后，发送收到消息给消息服务器。
- (8) 消息服务器将收到消息返回给调度中心，如未收到，重发。
- (9) 分公司请求断开连接，断开
- (10) 消息服务器将在线公司列表发给所有在线公司

### 3.2 协议规范

在网络通信中，协议是通信的核心。为了让传送和接收双方能够顺利沟通，必须制定一套双方都能理解的有固定格式、报头和报尾的自定义协议。由于在本系统中并没有复杂的消息传送，可通过指定特殊字符来区分消息格式。具体消息格式(定单消息体)如下：

消息类型	目的地区	目的公司	密码	定单	结束符
------	------	------	----	----	-----

为了便于理解协议内容，各消息体的消息项之间均采用分隔符分离，通过消息类型和消息结束符来区分有效消息和无效消息体。考虑到安全因素，将用户密码加密后嵌入消息体，在接收方需经解密后才能识别。

## 4 体系结构实现

### 4.1 客户端组件

客户端控件主要实现登录连接服务器，发送接收订餐消息，接收当前在线状态消息，接收登录/退网许可消息。组件设计说明见表1。

虽然在几乎所有Windows程序开发工具中，普通消息事件触发和处理都非常简单。但是对COM组件却可能引起很多麻烦，甚至很多COM

程序员都避免使用事件。由于消息服务的即时性，组件方法调用都由事件触发，所以必须把组件事件考虑在内。

COM中的事件又称自动化事件，是作为接口的一部分，这里的接口常被称为事件接口或输出接口(outgoing interface)，因为它不像别的接口那样由服务器(source)实现，而是由服务器的客户端(sink)来实现。接口的方法从服务器被外调到了客户端。每个事件接口都有相应的IID(接口标志符)以唯一的标识它们。

所幸的是在Delphi中提供了一些对自动化事件的支持功能。通过以下步骤即可完成事件的定义和触发。

- (1) 在type library定义事件
- (2) 更新类型库后，在\*impl实现文件中设置事件源
- (3) 实现事件响应 见图2

### 4.2 消息服务器

消息服务器要求24小时不间断运行，主要功能负责验证用户名和密码的正确性，转发订餐消息，存储订餐消息，写入日志文件或数据库；向分公司发送系统信息。见图3

#### 4.2.1 数据结构设计

建立在线用户信息RECORD类型结构，通过其中sock域唯一标识一个用户，并与Socket连接建立一一对应的映射关系，因为在基于TCP面向连接的服务中进行数据交换前，必须与通信方建立一条路径。这样既保证了通信双方之间存在路由，又保证了通信双方都是活动的，是彼此响应的。但缺点是在通信双方每建立一个通信信道需要很多开支。为了避免无用的连接占用过多资

源以及频繁的建立和释放Socket连接造成的网络堵塞。这种将在线用户信息与socket一一对应是非常必要的，同时也方便消息服务器对客户的管理和控制。

在Delphi中用Tsocket对象代表一个Windows socket句柄，在Winsock单元中定义如下：

```
type
  u_int = Integer;
  Tsocket = u_int;
```

从中看出，Tsocket其实就是一个整型值，可以在Windows socket API中使用来标识一个socket handle。这样当由于系统原因出现无效的Socket连接，并不马上释放内存，而是当有新的用户申请建立连接时，将这个句柄直接分配给它，从而解决了快速连续登录或恶意的攻击造成的问题。

#### 4.2.2 处理流程：

##### (1) 用户登录系统处理过程：

- ① 客户端分公司向消息服务器发送登录消息体。
- ② 消息体处理消息，取出用户名和密码，如果不存在用户或用户名和密码不符，则断开连接，将登录失败信息写入登录日志文件。
- ③ 如果用户名和密码验证成功，发送登录成功消息给登录用户，发送在线公司列表消息体给所有在线分公司。将登录成功信息写入登录日志文件。

##### (2) 订餐系统处理过程：

- ① 调度中心向消息服务器发送订餐消息体。
- ② 订餐消息体中的分公司信息为要送餐的分公司信息，消息服务器收到订餐消息体后取出分公司信息和用户名，将消息体存入消息体队列中，然后将消息体转发给要发送的分公司。
- ③ 分公司收到转发后的订餐消息后，发送收到消息体给消息服务器。
- ④ 消息服务器收到确认消息后，根据消息ID将消息从消息队列中删除，写订餐消息写入订餐数据库，发送确认消息给调度中心。

表 1

接口类型	接口名称	参数	说明
属性	HostIP	无	服务器地址
	SockPort	无	服务器端口
方法	Connect	地区名, 分公司, 密码	连接登录
	Disconnect	地区名, 分公司, 密码	退网
	SendMsg	地区名, 分公司, 订单	发送消息
事件	OnConnect	登录日期, 上/下午	登录成功事件
	OnDisconn()	无	退网成功事件
	OnMsgread	地区名, 分公司, 接收消息内容	消息接收事件
	OnUser	地区名, 在线分公司列表	登录状态事件
	OnExitUser	地区名, 分公司	退网状态事件

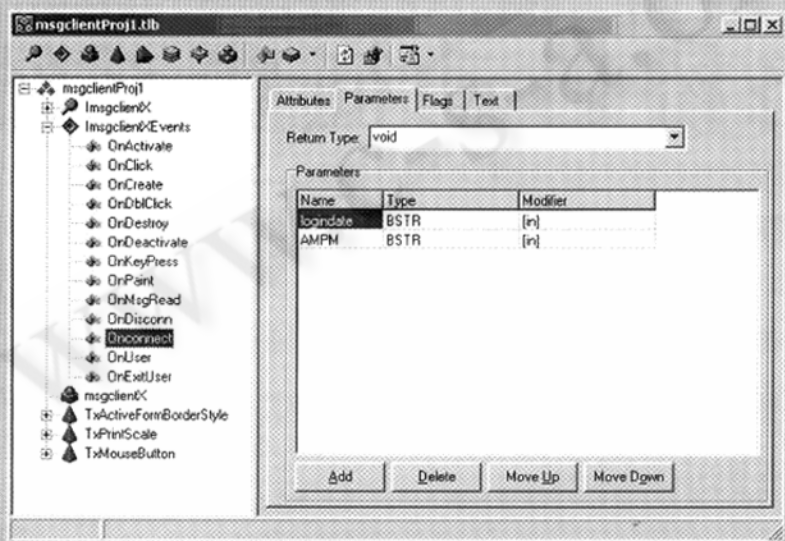


图 2 类型库编辑器

(3) 断网处理过程:

① 收到断网请求后, 从消息队列中查找是否有该公司消息。

② 如果有该公司消息未处理, 则处理后, 发送退网允许消息

③ 发送新的在线公司消息给在线的分公司。

(4) 异常处理过程:

① 本机重复登录: 保持连接, 重名登录则更新在线公司登录时间为当前时间, 如果异名登录, 则同时更新公司名和时间

② 非本机重名登录: 断开同名公司已有连接, 重新建立新连接, 更新在线公司列表

③ 发往消息公司异常退网: 从在线公司数据结构中删除该公司, 更新在线公司列表

### 4.3 软件开发平台

开发工具: 消息服务器软件和客户端 COM 组件采用 Delphi 6.0, 客户端界面使用 Visual Basic 开发, 操作系统为 Windows2000, 数据库平台采用 SQL Server2000。

## 5 结语

该系统现已在北京、上海等地进行投入运行, 从目前使用情况来看, 运行良好。由于采用了组件技术, 对原系统界面和操作没有丝毫更改, 节约了工作人员的培训费用, 也保证了公司的正常运转。同时新系统加强了网络容错处理, 对客户端连续快速登录, 异常退出和同名登录等情况都进行了考虑和处理, 使得系统的稳定性大大增强。 ▀

## 参考文献

- 1 静海, 深入 Delphi6 网络编程, 中国铁道出版社, 2001, 10。
- 2 Steve Teixeira, Xavier Pacheco. Delphi 5 开发人员指南, 2000, 7。
- 3 陈旭, Delphi COM 深入编程, 2000, 10。
- 4 牛新庄, 丁宗红, 用 Delphi 5.0 实现网络通信, <http://www.powerba.com/develop/delphi/>。

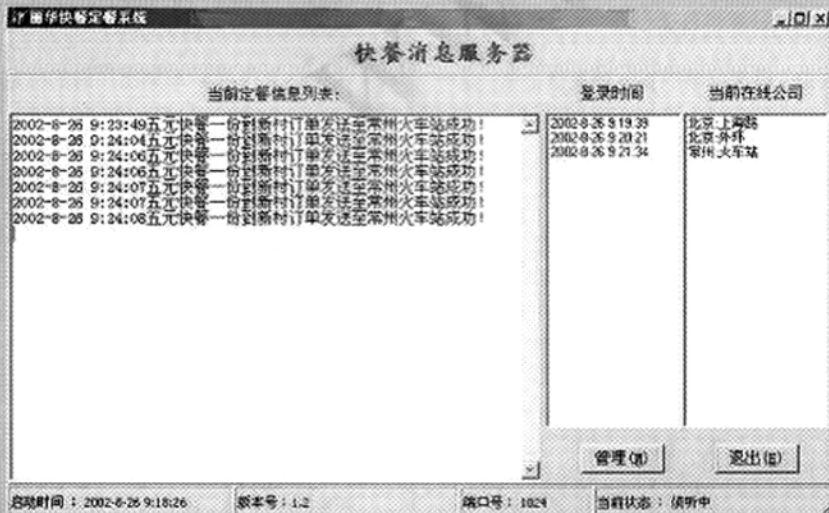


图 3 消息服务器主界面