

企业网中电子邮件服务的功能扩展应用

Application of Function Extended about Corporation Network Email serve

摘要: 本文针对目前企业网中的电子邮件服务提出了一种功能扩展的应用技术, 根据邮件接收方是否在线采用不同的邮件发送方式, 以减轻服务器负担, 提高网络应用效率。

关键词: 电子邮件 局域网 邮件传递

1 引言

目前在企业网上电子邮件的收发一般都是通过电子邮件服务器来完成的, 客户端仅用来浏览邮件或写邮件, 这样的系统如果一旦邮件服务器出现故障, 将导致整个企业邮件系统陷入瘫痪; 并且所有在线用户都必须通过邮件服务器收发邮件, 这在一定程度上也将影响到网络系统的执行效率。

为此, 本文提出一种功能经过扩展增强的电子邮件服务集成应用技术, 对普通的电子邮件服务进行优化, 以减轻邮件服务器的负担, 提高系统的工作效率。本系统使用 Delphi5.x 设计实现。

2 系统工作原理及主要功能

2.1 系统工作流程

系统工作流程如图 1 所示

2.2 系统主要功能及实现

2.2.1 显示网络主机列表

在网络用户进入邮件系统时都要经过用户登录窗口, 用户需输入相关用户名、口令、电子邮件地址、服务器 IP 地址等。可以使用 Delphi 提供的 AssignFile 和 Writeln 方法创建一个文本文件, 并将用户的信息保存到该文本文件中。当打开邮件发送窗口时可利用 AssignFile 和 ReadLn 方法从该文本文件中读出保存的用户信息, 然后将所有登陆系统的用户名显示在邮件

发送窗口的网络主机列表框中。

2.2.2 判断接收方是否在线

由于接收方是否在线决定邮件消息是直接发送还是通过邮件服务器发送, 因此在消息发送前需要判断接收方在线否, 可以让客户端根据用户所指定的网络服务器 IP 地址提取最新的网络主机映射表, 根据此表向局域网中的每台主机发出特定的查询信号, 再通过每台计算机是否有响应来判断该计算机是否在线, 对于在线计算机在发送邮件消息窗口的网络主机列表中显示时计算机名后加上“//Online”标志, 使发送方对所有在线计算机能一目了然。

在实际设计中, 可以将计算机的发送设置成不能发送空字符, 这样就可以用一个空字符作为查询信号, 其它主机的响应信号用两个空字符表示, 从而解决了判断其它主机是否在线的问题。具体代码如下:

```
for I:=0 to Form2.ListBox1.Items.Count-1 do
Begin
    NMMsg1.FormName:=Form2.NMMsg1.
    LocalIP;
    NMMsg1.Host:=HostList2.String [i*3+1];
    Try
        NMMsg1.PostIt(' '); // 发送空字符作为查询信号
        // 若发送成功, 则将 ListBox1 的相应列表后加上 //Online 字符串标记
        Form2.ListBox1.Items.Strings [i]:
        =Form2.ListBox1.Items.Strings [i] + ' //Online';
        NMMsg1.Disconnect;
    Except // 发送不成功, 则捕获错误信息
```

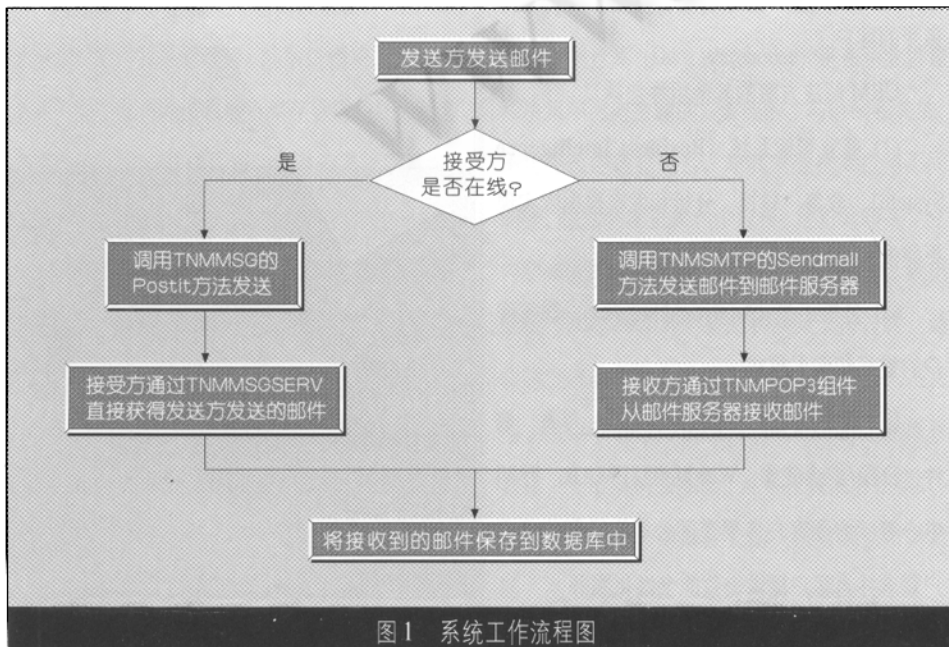


图 1 系统工作流程图

并断开连接

```

on E:Exception do
    begin
        NMMsg1.Disconnect;
    end;
end;
end;
...
{根据列表显示来判断接受者是否在线}
If StrPos(pchar(ListBox1.Items [ListBox1.
ItemIndex]),Pchar(' //Online '))< >nil then
    {接收者在线}
else
    {接收者不在线}
endif

```

2.2.3 邮件消息的发送与接收(详见下节)

2.2.4 立即通知功能

发送方总希望自己的消息能尽快被对方接收,该系统能做到不论对方在做什么,可以立即中断对方的工作,弹出信息并鸣铃提醒。

2.2.5 邮件保存功能

为帮助用户处理收到的邮件,包括保存、删除、阅读旧邮件,可将收到的邮件或消息存入数据表中,以便用户查询、管理。

3 邮件消息的发送与接收

如图1所示,邮件发送时要根据接收方是否在线来选择不同的邮件传送方式。若接收方在线利用 TNMMSG 组件在 TCP/IP 协议的网路中发送简单的 ASCII 文本文件,响应的接收方可直接通过 TNMMSGSERV 组件接受消息,消息邮件不需经过任何邮件服务器的存储及处理。如果接收方不在线,利用 TNMSMTP 将邮件发送到所连的邮件服务器,接收方通过 TNMPOP3 组件从邮件服务器接受邮件,显然,在这种情况下如果邮件服务器发生故障,邮件就不能被正常发送和接收。

3.1 不在线邮件的发送与接收

3.1.1 MOTIS 模型

当接收方不在线时,必须通过邮件服务器采用网络 E-mail 系统来完成邮件的发送和接收。在整个邮件系统的设计中,可参照 OSI 提出的 MOTIS 模型,主要包括以下几个部分:用户界面,主要用于邮件的生成和对邮件进行各种处理,“报文传输代理”指客户和服务器的传输处理,主要负责邮件的传输,“邮局”实际是邮件服务器上的消息库,由邮件服务器上的收件箱和发件箱目录组成,消息库中包含用户 ID、口令、用户优先级、消息文件夹和带有任何附件的邮件消息。邮件在本地系统准备好之后,系统内存存储这条消息,用户发出相应指令后,就传送它。消息一般是通过从定向传送到邮件服务器上的收件箱目录,UA 可以创建、阅读、格式化或删除消息,所以它在邮件服务器上必须有相应的权利,但 UA 只能对一个邮件服务器有存取权,如果有不止一个邮局存在,那么在网路上至少有一个 MTA 用于在两个邮局之间交换邮件。MTA 在消息库中是以存储转发的方式来交付邮件。

目前,在一些成熟的电子邮件服务器软件中对邮件服务器与油箱都有较好的支持。

3.1.2 Delphi5.x 中用于接收和发送电子邮件的组件

(1) TNMMSMTP 组件: TNMMSMTP 向 SMTP 服务器发送邮件,它所连接的邮件服务器由 Host 属性和 Post 属性设置,Connect 方法建立到邮件服务器的连接,而 Disconnect 方法则断开连接,PostMessage 属性包含了要发送的电子邮件的内容,实际的发送任务有 SendMail 方法完成,使用 Verify 方法可以检查邮件服务器上是否存在某个特定的电子邮件用户,使用 ExpandList 方法可以获取电子邮件列表,相应的处理代码可以添加到事件句柄 OnMailListReturn 中,PostMessage 属性是一个在邮件发送与接收中必不可少的属性,它包含邮件信息内容,如发送者

名称、发送者电子邮件地址、接收方应答地址、附件列表、邮件正文内容、邮件主题等信息,Host 属性包含了一次 SMTP 邮件事务中使用的邮件服务器名,可将此属性赋值为用户登陆窗口中输入的服务器地址: NMSMTP.Host:=Form3.Edit4.Text, Post 属性用于指定连接到远程主机的端口号,可使用其缺省值 25,属性 UserID 则指定了登陆到 SMTP 邮件服务器的用户名,可从“用户资料”的“用户名”中获得这一消息,TimeOut 属性设置了等待套接字响应的的时间,超过这个时间值,当前操作将取消,系统同时引发一个异常,邮件发送步骤:当属性设置完成后,可通过 Connect 方法连接到邮件服务器,然后用 SendMail 方法将电子邮件发送到 SMTP 邮件服务器,再用 Clear Parameters 方法来消除属性 PostMessage 的所有字段,最后用 Disconnect 来断开与邮件服务器的连接。

(2) TNMPOP3 组件: TNMPOP3 组件可从 POP3 服务器检索邮件, TNMPOP3 所连接的 POP3 服务器由该组件的 Host 和 Port 属性设置,Connect 方法建立到邮件服务器的连接,而 Disconnect 方法断开与邮件服务器的连接,属性 UserID 和 Password 为有效电子邮件帐号的用户名和密码,使用 GetMail Message 方法可以检查 POP3 邮件服务器上的电子邮件,邮件的内容保存在 Mail Message 属性中, DeleteOnRead 用来指定是否读取邮件后删除它,缺省值为不删除,在设计中为了便于将邮局中的电子邮件依次加到数据库中,该值设为 False, MailCount 包含了服务器上用户油箱中的邮件数目, DeleteMail Message 方法删除由参数 MailNumber 指定的电子邮件,在断开连接前,此方法删除的邮件不会从物理上删除。

3.1.3 已接收邮件的保存

为保存收到的邮件可建立一个数据表,在接收邮件的同时使用一条循环语句将邮件依次加入此数据表,该数据表由两个字段组成,其结

构如表 1 所示。

邮件一旦被保存到数据表中，就可把邮件服务器中的邮件删除，使其不影响下一次的邮件接收，具体代码如下：

```
count:=NMPOP31.MailCount;
if count<>0 then
  for I:=1 to count do
    begin
      Table1.Append;
      NMPOP31.GetMailMessage(i); //从邮件服务器上依次接收邮件
      DbMemo1.Lines.Text:=NMPOP#1.
      MailMessage.Form+'发给你的信'; //显示发送者信息
      DbMemo2.Lines.Assign(NMPOP31.
      MailMessage.Body); //显示邮件内容
      Table1.Post
    end
  end
else
  Edit2.Text:= '没有您的信!';
endif
```

然后将数据表与接收窗口中用于显示发送者信息和邮件内容的 DBMemo1、DBMemo2 控件建立关联，利用窗口提供的数据库记录定位按钮就可向前向后查看数据表中保存的每一条邮件内容。

3.2 在线消息的发送与接收

当接收方在线时，就采用消息的直接发送与接收，不必通过邮件服务器。由于消息的传输不必通过服务器，所以相对于邮件的传输就简单得多。利用 Delphi 5.x 的 TNMMSG 和 TNMMSGSERV 组件实现消息的发送与接收。

TNMMSG 组件使用 TCP/IP 协议发送简单的文本消息。在发送前须先设置 TNMMSG 组件的 Host 属性为远程主机名或主机 IP 地址，Post 属性为通信端口号，可使用缺省值 6711。

表 1 数据表结构

字段名	类型	长度	描述
Fromname	字符型	40	保存邮件发送者信息
Message	备注型	4	保存邮件内容

FormName 属性使收件人知道消息的来源，该属性在运行和设计时都有效。设置完上述三个属性，通过调用 TNMMSG 组件的 Postit 函数来发送消息，如果消息被成功发送，函数返回 OK 字符串，实现代码如下：

```
NMMsg1.Host:=Form1.HostList2.Strings
[ListBox1.ItemIndex*3+1];
NMMsg1.Post=6711;
NMMsg1.FromName:=Form3.Edit1.Text;
NMMsg1.PostIt(Form2.Memo1.Text);
```

TNMMSGSERV 组件接收 TNMMSG 组件通过网络发送的消息，此组件必须与 TNMMSG 组件成对出现在客户机上才能保证正确工作。同样需先设置端口属性。消息成功发送到接收计算机时触发 OnMsg 事件，事件参数 sformsFrom 记录该消息的发送者，Smsg 参数为消息内容。为及时保存收到的消息及便于以后处理、查询，同样可将消息保存到数据表中，先用 Table1.Append 命令在数据库中添加一个空记录，然后用 Table1.Post 命令将所收到的消息添加到数据表中，并按先后顺序排列，这样，在接收消息的对话框中显示的消息实际上是从数据库中提出的，把消息的处理转变为数据库数据的处理，这样可使消息处理功能进一步完善。

```
Procedure TForm1.NMMSGServMSG(Sender:
Tcomponent;const sFrom,sMsg:String);
Begin
  If (Trim(sMsg)<>`)and(Trim(sFrom)<>`)
then
  Begin
```

```
Table1.Append;
DbMemo1.Lines.Text:=sFrom+'发给你的
消息!'; //显示发送者信息
DbMemo2.Text:=Smsg; //显示消息内容
Table1.Post;
If n20.Checked then
  MessageBeep(1000); //铃声提示有消
息传来
If n19.Checked then
  ShowMessage(sFrom+'发给你的消息!
'+#13#10+sMsg);
End;
...
```

4 结束语

网络的运用使得企业各部门之间的行为需要经常协调，因此相互通信频繁。采用本技术设计的电子邮件服务可有效的利用资源，不仅在一定程度上减轻了邮件服务器的负担，还解决了传统的邮件消息传送支持在线传送的局限，当接收方不在线时仍可发送邮件消息到服务器，十分适合在企业网和校园网中使用。 ■

参考文献

- 1 李维, Delphi 5.x 分布式多层应用系统篇, 机械工业出版社, 2000.
- 2 王涛, 多层分布式数据库实战, 清华大学出版社, 2000.
- 3 乔林, 扬志刚, Delphi 5.0 程序设计 Internet 应用实务篇, 中国铁道出版社, 2000.