

在VisualBasic中使用Winsock控件 PDF转换可能丢失图片或格式  
，建议阅读原文

[https://www.100test.com/kao\\_ti2020/275/2021\\_2022\\_\\_E5\\_9C\\_A8VisualB\\_c67\\_275904.htm](https://www.100test.com/kao_ti2020/275/2021_2022__E5_9C_A8VisualB_c67_275904.htm) WinSock控件能够通过UDP协议（用户数据报协议）或TCP协议（数据传输协议）连接到远程的机器并进行数据交换。这两种协议都能用来创建客户端和服务端应用程序。就像定时器控件一样，WinSock控件运行时没有一个可视的界面。可能的用途 创建客户端应用程序，它能在信息到达中央服务器之前把用户的信息收集起来。创建服务端应用程序，它能作为来自多个用户的数据一个集中处理点。创建“聊天”程序。协议的选择 当我们使用WinSock控件时，首先要确定的是使用TCP还是UDP协议。它们之间主要的区别在于连接状态：TCP协议控件是一个基于连接的协议，就像电话机一样，用户必须在通话之前建立连接；UDP是一个无连接的协议，两台计算机之间的事务处理就像传纸条一样：一台计算机向另一台计算机发送消息，但是它们之间并没有一个明确的连接路径。另外，发送的单个信息量的大小取决于网络。通常，你要创建的应用程序的类别就决定了你要选择的协议。以下是几个能够帮助你选择合适的协议的问题：当发送或接收数据时，该应用程序需要从服务端或客户端获得认证吗？如果要的话，那么TCP协议就正好需要在发送或接受数据前建立明确的连接。要发送的数据量大吗？（就像图片、声音文件之类）一旦建立了连接，TCP协议就会保持连接并保证数据的完整性。但是，这种连接会占用的更多的处理器资源，成本也会更高一些。数据是陆续传输的，还是一次全部传完呢？比如，如果你要创建的应用程序

在某些任务完成时会告知具体的计算机，那么选择UDP协议会更合适一些。UDP协议也更适合于发送少量数据。协议的配置 配置你的应用程序所用到的协议：在设计阶段，单击工具窗口里的协议，选择 `sckTCPProtocol`或`sckUDPProtocol`。你也可以在代码里配置协议，就像下面这样：

`Winsock1.Protocol=sckTCPProtocol` 确定你的计算机名 要连接到远程的计算机，你必须知道它的IP地址或别名。IP地址是一串用句点分隔的3位数字。通常，计算机的别名更容易让人记住。按下面的步骤可以找到你的计算机名: 在“任务栏”里单击“开始” 在“设置”选项里单击“控制面板”；双击“网络”图标；单击“网络标识” 在“计算机名”中显示的就是你的计算机名。一旦你找到你的计算名，它就可以作为远程主机的属性来用了。 TCP连接入门 当用TCP控件创建应用程序的时候，必须首先明确你的程序是作为服务端还是客户端。创建服务端程序就意味着你的程序能够在指定的端口进行“监听”，而客户端则能够提出请求，服务端能够接受请求并实现连接。一旦连接建立起来，客户端和服务端就能够自由地进行通信。创建服务端程序 下面是创建一个简单服务端程序的步骤：创建一个标准EXE工程；把默认窗体(Default form)的名字改为`frmServer`. 把form的标题（caption）改为TCP Server；把Winsock控件拉到窗体中，并命名为`tcpServer`；在窗体中添加2个文本框，分别命名为`txtSendData`和`txtOutput` 在窗体中加入下列代码；`Private Sub Form_Load()` 在窗体中加入下列代码；`Private Sub Form_Load()` ' Set the LocalPort property to an integer. ' Then invoke the Listen method. `tcpServer.LocalPort = 1001` `tcpServer.Listen` `frmClient.Show` ' Show the client form. `End Sub`

```

Private Sub tcpServer_ConnectionRequest _ (ByVal requestID As
Long) ' Check if the control ' s State is closed. If not, ' close the
connection before accepting the new ' connection. If
tcpServer.State = sckClosed Then _ tcpServer.Close ' Accept the
request with the requestID ' parameter. tcpServer.Accept
requestID End Sub Private Sub txtSendData_Change() ' The
TextBox control named txtSendData ' contains the data to be sent.
Whenever the user ' types into the textbox, the string is sent '
using the SendData method. tcpServer.SendData txtSendData.Text
End Sub Private Sub tcpServer_DataArrival _ (ByVal bytesTotal As
Long) ' Declare a variable for the incoming data. ' Invoke the
GetData method and set the Text ' property of a TextBox named
txtOutput to ' the data. Dim strData As String tcpServer.GetData
strData txtOutput.Text = strData End Sub

```

上面就是创建一个简单的服务端应用程序的过程。然而，要完成整个过程，你还得创建一个客户端程序。创建TCP客户端程序 在工程中添加一个新的窗体(form)，并命名为frmClient；将窗体的标题(caption)改为TCP Client；添加一个Winsock控件到窗体中，命名为tcpClient；添加2个文本框控件到frmClient窗体，分别命名为txtSend和txtOutput；添加一个按钮控件(CommandButton)到窗体，命名为cmdConnect；将按钮控件标题(caption)改为Connect；在窗体中添加下面代码：注：确保将远程主机属性(RemoteHost property)改为你的计算机别名。

```

Private Sub Form_Load() ' The name of the Winsock
control is tcpClient. ' Note: to specify a remote host, you can use
' either the IP address (ex: "121.111.1.1") or ' the computer ' s

```

```
"friendly" name, as shown here. tcpClient.RemoteHost =  
"RemoteComputerName" tcpClient.RemotePort = 1001 End Sub  
Private Sub cmdConnect_Click() ' Invoke the Connect method to  
initiate a ' connection. tcpClient.Connect End Sub  
Private Sub  
txtSendData_Change() tcpClient.SendData txtSend.Text End Sub  
Private Sub tcpClient_DataArrival _ (ByVal bytesTotal As Long)  
Dim strData As String tcpClient.GetData strData txtOutput.Text =  
strData End Sub
```

以上代码就能创建一个简单的客/服应用程序。要试着让两者建立连接，可以运行工程，单击Connect。在任意一个txtSendData文本框中输入文本，同样的文本信息就会出现在另一个窗体的txtOutput文本框中出现。

100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)