

## 16 Utilisation des composants sockets

Les composants **TServerSocket** et **TClientSocket** ne sont pas présent par défaut. Il doivent être installés

Selectionner Composant > Installer des Packages puis

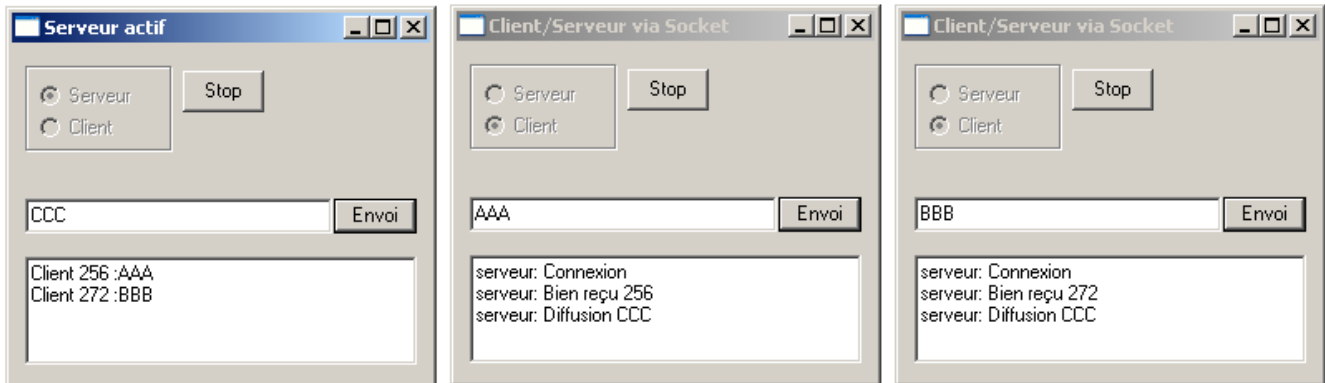
Ajouter. Le package se trouve dans le repertoire **bin** de l'installation de Delphi. C'est un fichier dont le nom commence par **dclsockets** et qui a pour extension **bpl**.

Adresse IP et port doivent être renseignés.


Attention! Autoriser si besoin l'usage du port dans le **FireWall**

Communication simple par TCP/IP

Dans cet exemple nous démarrons trois instances polyvalentes d'un même programme :



Un de ces programmes va avoir le rôle de serveur. Les autres le rôle de client.

<pre> .../... uses   System.Win.ScktComp;  type   TMode = (Indefini, Client, Serveur);    TForm1 = class(TForm) .../...     ServerSocket1: TServerSocket;     ClientSocket1: TClientSocket; .../...   end; var   modeCS: TMode = Indefini;  procedure TForm1.ActiveSocket(mode1: TMode); begin   modeCS := mode1;   case modeCS of     Serveur:       begin         if not ServerSocket1.Active then           ServerSocket1.Active := True;         if ServerSocket1.Active then           form1.caption := 'Serveur actif';         end;       Client:         begin           ClientSocket1.Active := True;           if ClientSocket1.Active then             form1.caption := 'Client Connecté';           end;         end;       end;   end; end; </pre>	<p>L'unité est automatiquement ajoutée lors de l'insertion des deux composants</p>  <pre> object ServerSocket1: TServerSocket   Port = 5725   ServerType = stNonBlocking   OnClientConnect = ServerSocket1ClientConnect   OnClientRead = ServerSocket1ClientRead end  object ClientSocket1: TClientSocket   Address = '127.0.0.1'   ClientType = ctNonBlocking   Port = 5725   OnDisconnect = ClientSocket1Disconnect   OnRead = ClientSocket1Read   OnError = ClientSocket1Error end </pre> <p>Le type <b>Tmode</b> définit l'état de l'application Variable globale <b>modeCS</b> initialisée à <b>indéfini</b></p> <p>Activation du module par <b>ActiveSocket(Serveur)</b> ou <b>ActiveSocket(Client)</b>;</p>
---	--

<pre> procedure TForm1.DesactiveSocket; begin   case modeCS of     Serveur:       if ServerSocket1.Active then         ServerSocket1.Active := False;     Client:       ClientSocket1.Active := False;   end; end; </pre>	<p>Désactivation du socket actif (client ou serveur)</p>
---	--

<pre> ... / ...   for i1 := 0 to ServerSocket1.Socket.ActiveConnections - 1 do     ServerSocket1.Socket.Connections[i1].SendText(stMessage); ... / ...    ClientSocket1.Socket.SendText(stMessage) ... / ... </pre>	<p>Envoi côté serveur d'un message à tous les clients <b>(broadcast)</b></p> <p>Envoi côté client d'un message au serveur</p>
---	---

<pre> procedure TForm1.ServerSocket1ClientConnect(Sender: TObject;   Socket: TCustomWinSocket); begin   Socket.SendText('Connexion'); end;  procedure TForm1.ServerSocket1ClientRead(Sender: TObject;   Socket: TCustomWinSocket); begin   Memo1.text := Memo1.text + 'Client ' + IntToStr(Socket.SocketHandle) +     ':' + Socket.ReceiveText + #13#10;   Socket.SendText('Bien reçu ' + IntToStr(Socket.SocketHandle)); end; </pre>	<pre> object ServerSocket1: TServerSocket   OnClientConnect =   ServerSocket1ClientConnect   OnClientRead = ServerSocket1ClientRead end </pre> <p>Une anomalie de connexion lorsque le serveur n'est pas actif génère une erreur 10061</p>
<pre> procedure TForm1.ClientSocket1Disconnect(Sender: TObject;   Socket: TCustomWinSocket); begin   Socket.SendText('Client Socket Disconnect'); end;  procedure TForm1.ClientSocket1Error(Sender: TObject; Socket:   TCustomWinSocket;   ErrorEvent: TErrorEvent; var ErrorCode: Integer); begin   ErrorCode := 0;   ClientSocket1.Active := False;   Memo1.text := Memo1.text + 'Pb connexion' + #13#10; end;  procedure TForm1.ClientSocket1Read(Sender: TObject; Socket:   TCustomWinSocket); begin   Memo1.text := Memo1.text + 'serveur: ' + Socket.ReceiveText + #13#10; end; </pre>	<pre> object ClientSocket1: TClientSocket   OnDisconnect = ClientSocket1Disconnect   OnRead = ClientSocket1Read   OnError = ClientSocket1Error end </pre>