

## Remote Control Library – Librairie Remote Control

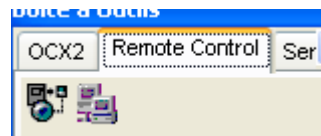
### Installation

File, Fichier : REMOTEVPU.EXE



### After installation Après installation

In the toolbox :  
Dans la boîte à outils :



### Files Added, Files Ajoutés : C:\API32\Modules\

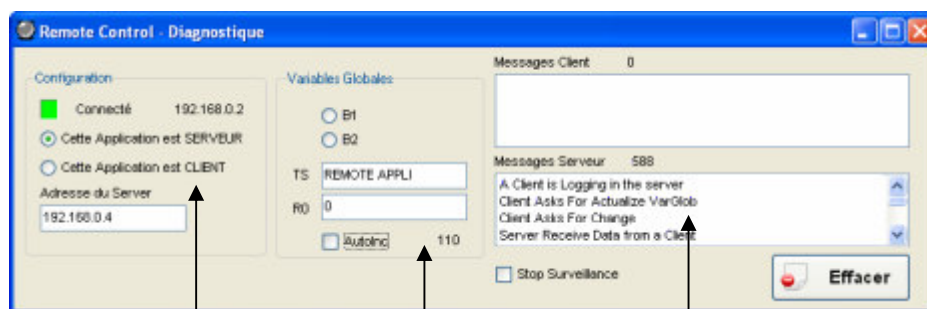
**Remoteserv.Pas, RemoteServ.Vpu** : Tache Serveur et client : Client/Server task  
**RemoteControl.cmp, RemoteControl.Vpu** : Visual I/O component, Composant Visual I/O  
Allow to transform your application in a client or a server.  
Permet de transformer votre appli en client ou en serveur.

**IpadrLocal.Cmp, IpadrLocal.Vpu** : Visual I/O component, Composant Visual I/O.  
To display the IP Address of your PC,  
Pour visualiser l'adresse ip de votre PC

**Winsock.pas, Winsock.vpu** : Windows library interface .  
Librairie Windows permettant de gérer les sockets.

### Application - Test

in <INSTALLDIR>\REMOTECONTROL



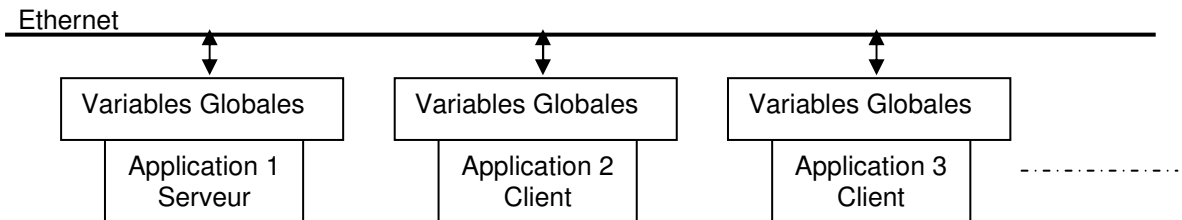
Selection : Client ou serveur    Variables Globales    Messages Client/Server

## Communication par réseau entre plusieurs applications Visual I/O

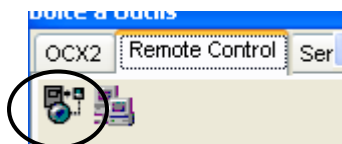
### Principe

Les applications doivent inclure la même déclaration de variables globales. Lorsque une des applications modifie une des ses variables globales, la nouvelle valeur de cette variable est transmise aussitôt sur le réseau à toutes les autres applications connectées.

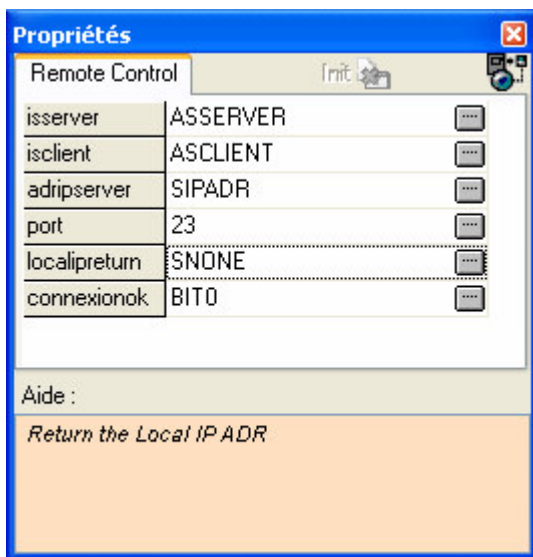
Lors d'une connexion au réseau d'une nouvelle application, celle-ci demande une mise à jour totale des ses variables globales à l'application déclarée comme serveur.



Le Composant RemoteControl dans la boîte à outils



Ce composant permet de transformer votre application en serveur ou client.



**IsServer** : Booléen permettant de définir si l'application est serveur.

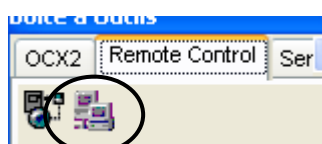
**IsClient** : Booléen permettant de définir si l'application est client.

**AdripsServer** : Chaîne de caractère contenant l'adresse IP du serveur dans le cas où l'application est client.

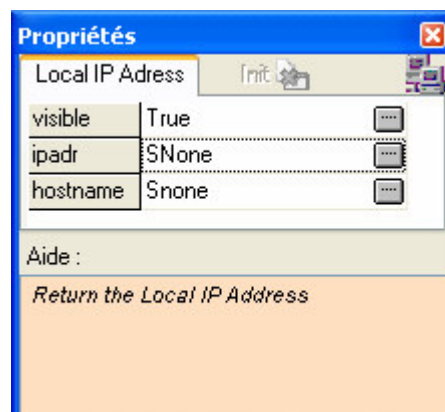
**Port** : Numéro de Port utilisé.

**LocalIpReturn** : Retour dans une chaîne de caractère de l'adresse IP du PC. Si Le pc est serveur cette adresse IP est utilisée par les client pour ce connecter.

**ConnexionOK** : Booléen indiquant si la connexion en tant que serveur ou client est OK.



Ce composant permet d'afficher les informations de votre PC qui est connecté au réseau.



**Visible** : Texte visible ou non.

**Ipadr** : Renvoie l'adresse IP du PC.

**Hostname** : Renvoie le nom du PC.

## La librairie RemoteServ.vpu

Uses Winsock, Sysplc, Linkedlist, Threads;  
Interface

### Var

```
Remote_Server_Port : Integer =23;  
RemotePassWord    : String='Arsoft';  
  
RemoteServerOpen  : Boolean; /** Server is open.  
  
ClientLogged      : Boolean; /** A client is logged  
WatchClient       : String; /** State of Client  
CountRecClient    : Integer; /** Number of communications client  
WatchServer       : String; /** State of Server  
CountRecServer    : Integer; /** Number of communications server  
ViewDebug         : Boolean=true; /** Debug string returns
```

### Cette Librairie permet

**Remote Server Port** : Fixe le Numéro de port (équivalent à Port dans le composant Remote control)

**RemotePassWord** : Pour ce connecter au serveur le client envoie ce mot de passé si il est équivalent à celui du serveur la communication est autorisée.

**RemoteServerOpen** : Dans le cas ou l'application est serveur, ce booléen passe à true si l'application est en réception de commande d'un éventuel client.

**ClientLogged** : L'application si elle est client est connectée au serveur.

**WatchClient** : Texte permettant d'afficher l'état actuel de l'application si elle est client.

**CountRecClient** : Nombre de dialogue de l'application si elle est cliente.

**WatchServer** : Texte permettant d'afficher l'état actuel de l'application si elle est serveur.

**CountRecServer** : Nombre de dialogue de l'application si elle est serveur.

**ViewDebug** : Booléen autorisant ou non le renvoi d'états du serveur ou du client dans les variables WatchServer et WatchClient.

### Fonctions disponibles

**Function** InitRemoteServer : **Boolean**;

**Function** InitRemoteClient (IPA : **String**; CPort : **Integer**) : **Boolean**;

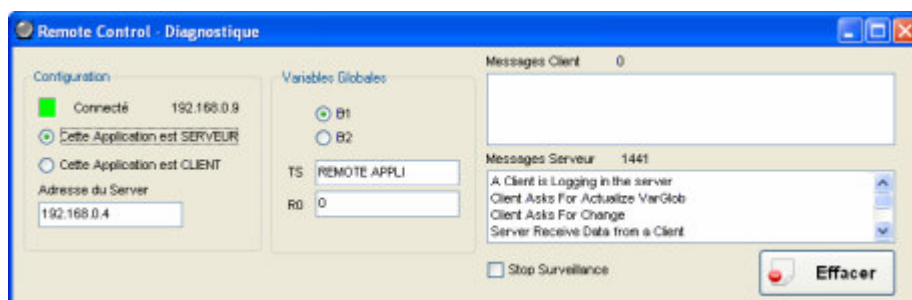
2 fonctions sont aussi disponible pour définir un serveur ou un client dans votre application. Vous pouvez utiliser ces fonctions dans le cas ou vous n'utilisez pas le composant RemoteControl.

**Function** InitRemoteClient (IPA : **String**; CPort : **Integer**) : **Boolean**;

**IPA** : Adresse IP du serveur.

**CPort** : Port de connexion. (si =0 ou omis, Moi Page 3 12/01/2007 le port est 23);

**Programme de démonstration** : CLIENTSERV.EXE permet d'être lancé sur plusieurs PC et de visualiser l'activité, de forcer des variables globales B1 (boolean) , B2(boolean), TS(string), R(Real).

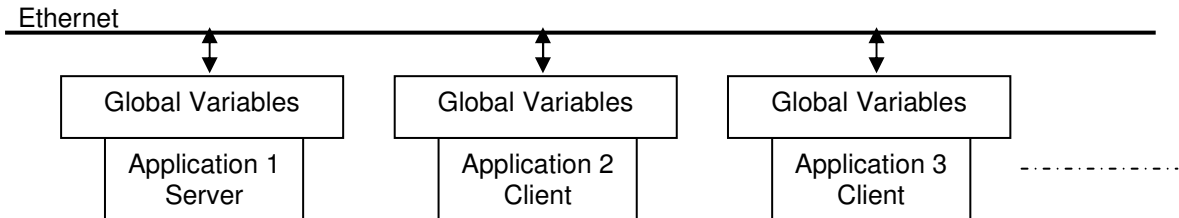


## Communication by Ethernet between several Visual I/O applications

### Principle

The applications must include the same global variables declarations. When one of the applications modifies one of its global variables, the new variable value is sent immediately on the network to all other connected applications.

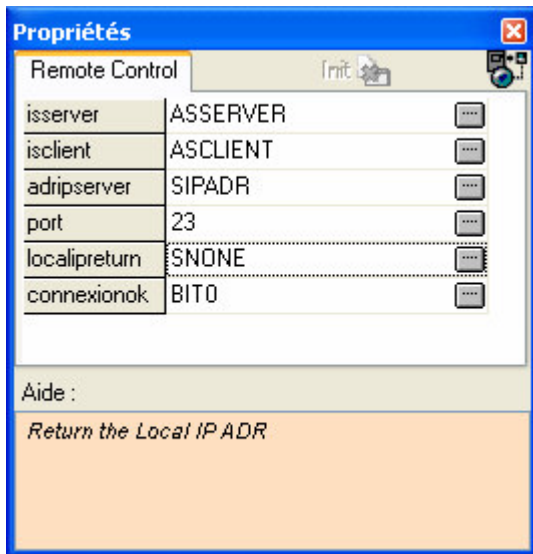
During a connection at the network of a new application, this one asks for a total update of its global variables from the application declared as server.



The RemoteControl component in the toolbox :



This component allows transforming your application in a server or a client.



**IsServer** : Boolean enables the application to become a server.

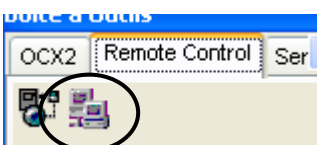
**IsClient** : Boolean enables the application to become a Client.

**Adripserver** : String retrieving the IP address of the server in the case of the application is a client.

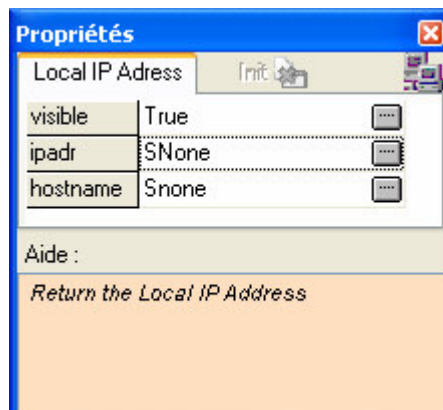
**Port** : Port number used.

**LocalipReturn** : Returns in a string the PC ip address. If the pc is server, the clients to connect it use this address.

**ConnexionOK** : Boolean returning the state of the communication in server or client mode.



This component allows showing the information of your PC connected to the network.



**Visible** : Hide / Show the text.

**Ipadr** : Returns the IP address of the PC.

**Hostname** : Returns the name of the PC

## The RemoteServ.vpu library

**Uses** Winsock, Sysplc, Linkedlist, Threads;  
Interface

### **Var**

```
Remote_Server_Port : Integer =23;  
RemotePassWord    : String='Arsoft';  
  
RemoteServerOpen  : Boolean; /** Server is open.  
  
ClientLogged      : Boolean; /** A client is logged  
WatchClient       : String; /** State of Client  
CountRecClient    : Integer; /** Number of communications client  
WatchServer       : String; /** State of Server  
CountRecServer    : Integer; /** Number of communications server  
ViewDebug         : Boolean=true; /** Debug string returns
```

## This library allows

**Remote Server Port :** Sets the port number (equivalent to Port in the Remote control component)

**RemotePassWord :** To connect to the server, the client sends this password. If it is equivalent to the server's one the communication is authorized.

**RemoteServerOpen :** In the case of the application is server, this Boolean is true if the application is waiting message from a client.

**ClientLogged :** True if the application is a client and is connected to a server

**WatchClient :** Text allowing to show the current state of the application if it is a client.

**CountRecClient :** Number of communications of the application if it is a client.

**WatchServer :** Text allowing to show the current state of the application if it is a server.

**CountRecServer :** Number of communications of the application if it is a server.

**ViewDebug :** Boolean authorizing or not the return of the server or the client states in the variables WatchServer and WatchClient.

## Available functions

**Function** InitRemoteServer : **Boolean**;

**Function** InitRemoteClient (IPA: **String**; CPort : **Integer**) : **Boolean**;

2 Functions are also available for defining a server or a client in your application.

You can use these functions in the case you do not use the RemoteControl component.

**Function** InitRemoteClient (IPA: **String**; CPort : **Integer**) : **Boolean**;

**IPA** : IP Address of the server.

**CPort** : Connection Port. (if =0 or missing, the port is 23)

**Demonstration program:** CLIENTSERV.EXE allows to be launching on several PC and to display the activity. You can also force global variables B1 (Boolean), B2 (Boolean), TS (string), R (Real).

