

Presentation

SpyObjects

Provide remote control functions to your program in different systems (Linux & Windows), different language configurations, VCL (for delphi 7 & 6), CLX (Delphi7 & Kylix 3) and ActiveX (for all COM compatible languages). SpyObjects source files are a cross-platform remote control developer library that link your program to anywhere from any system. Using Pascal, HTML, DHTML, Java & JavaScript. All in one tool. You can watch from anywhere what happens in the P.C application where it's installed with a simple browser.

Features

- **Remote Java desktop.** With the new spyviewer Java applet, you have full desktop access to the remote computer. You can see remote screen, drag & drop, key events, mouse movement in the remote host.
- **Files & directory navigation.** Sort by name, size. Download , Upload , rename , delete, see, edit host files.
- **Remote Object Inspector.** See tree components and properties, change properties, see what property in what object change, of your remote pc/ applicationDump memory. Delete & create objects. Find objects. List dump, collection dump.
- **See remote static screen and click & key events.** You can configure the resolution and quality of screen image
- **On-line demo system.** You execute your application, and your client can see a TV-Window of your computer.
- **Proxy net.** One Gp component can be a bridge between other 2 computer.
- **off-line & E-mail Suport** You can resume a session of your application (save screen & component changes), Then when you connect with granprimo and see the changes and historic log. You can configure it to send to you this informatil via e-mail.
- **Hex Editor** of remote application memory.
- **See active process** . You can kill , see properties.
- **See machine information.** Cpu speed, win version, Local Net Info
- **Alias db access.** Add, change, delete remote alias.
- **Multi-language.** English, Castellano..
- **Extensible.** You can easy add your own capabiltes.
- **Very compact.** You only add 200 K to your application.

Future

- Patch remote program.
- I'll like to add a step by step debugger.

GranPrimo is a beta product.

¿ Why GranPrimo ?

As a difference from other remote-control programs, where you install a server program in the "server side" and you need to have the same program or the client version program to see the screen from another computer, with granprimo you only need to install it, and then you can "take the remote-control" anywhere, because you only need a html browser. Screen, Files, Remote Object inspector, *off-line debug*.

¿ who needs granprimo?

Those who want to see what happens in a remote computer.

- 1- A programmer to control remote application.
- 2- A programmer to show a demo application.
- 3- teacher in a Demo cyber -net lesson
- 4- see if your boss, secretary, ... is working from the office or outside the office.
- 5- maybe you are cyber-showman and you want the people can see what you are doing.
- 6- You want to make a joke to a friend, and click him in the screen without another complex program.

Different packs

Because there is different scenarios , there is different spy-packs configurations.

SpyIndy pack

Indy version of spyobjects library (Both Kylix and Delphi)

TGooSpyIndy is the basic TIdCustomHTTPServer extension that provide primary remote web access to your program. Funny access to your files & program objects in Windows and Linux. You can find, read, edit, send, download files, and remote object inspection Ready for delphi and kylix. (not desktop remote control support)

TspicelIndy provide instant web access to your Delphi & Kylix form application. By instant translate Pascal form component to HTML form. Then you can use the program form both in local and remotely by web browser.

Limitation

There is only supported the normal VCL & CLX controls, (but you can extend your own control support) Isn't multi-state user. (you haven't different session forms).

Example

The example linked is the same example of Borland convert demo application with only one SpiceObject component dropped in form and active property = true.

TspyIndy is a spy extension that provide full remote control, similar than TGranPrimo but in Indy framework (The remote screen control, only work in Windows).

Kprimo pack

Is the Kylix release of spyObjects. At the moment, screen remote control is not enabled , but you have file remote control & Remote object inspector. It uses Indy components as communication library.

The source files are the same in kprimo that in SpyIndy, But there is different *.dcu distribution because {\$linux} directive make different compilations.

The component in this pack (TgooSpyIndy and TspiceIndy).

GranPrimo pack

Is a stand-alone Delphi component for remote control and debug your normal delphi Application. You drop TGranPrimo component in your application, and virtually , you have remote control access to it.

Gp offer all remote control features and extensions like off-line debug (send a resume of program execution by e-mail), Bridge function, etc...

It don't use Indy or IntraWeb library.

SpyIntraWeb

Spy Library for IntraWeb users , with purpose of remote control of IW server.

There is two basic classes (the engine and the viewer)

TSpyIntraWeb. Is the Remote control engine.

TiWGranPrimo. Is the TiwControl extension wich connect with TSpyIntraWeb .

The render html result, it depend of TSpyIntraWeb.SpyFuncion property.

Spyfuncion values:

- **sfjprimo** . Remote desktop.access by java applet.
- **sfCPUMachine** : Remote system information.
- **sfSeeScreen** : static remote screen image.(With the TiwGranprimo control size). You can do mouse clicks.
- **sfEditObject** : Remote object Inspector.
- **sfTreeObject** = remote object tree.
- **sfFiles**= driver, folder and files tree,
- **TsfLocalNet** . Local net Information.

You can have several tspyintraweb and TiwGranprimo components in the Iw form, showing different windows information. (screen, files... together).

SpyObjects basic instructions

1/ Configure . The main parameters in Gp :

Port : The port number where the computer is listening . (default 80).

Home Page: name of the main page address, (where you enter the password to access to the computer).

password: password to access the computer.

if PassWord is NULL, then you can access to the computer, writing only the host name or address.

(PassWord is 'hola' by default, and 'granprimo.es' is the home page).

2/ access to the computer, by writing

http://hostname/homepage (in the explorer direction bar)

http://193.153.251.45/granprimo.es (example)

(note: you can change "granprimo.es" by another word like "menu")

You can change the Port (default is 80).

But then you must write the port number in the direction, example:

http://193.153.251.45:69/granprimo.es

if you clear the password then only need

http://193.153.251.45/

but all the people can write the same and access to the computer.

3/ If you access to the homepage with the browser, the first page ask you for the password and screen configuration to access to the computer.

If the password is Ok then you access to the main menu page where you can choose the next action.

4/ In home page there is some fields to fill

username: isn't necessary.

password: is needed if Gp password property is set.

zoom: scale % of screen ,(between 10 and 100) lower values is a little screen, hi values is a bigger screen . If you write nothing is in default zoom (60%).

jpeg: jpeg precision.(between 10 and 100) lower values is a bad image and hi values is better images, (but less speed).

5/ you can try Gp in the local computer by writing

127.0.0.1/granprimo.es (if you have password)

or

127.0.0.1 (only if you haven't password)

in the navigator you must write "hola" (case sensitive) in password).

Custom home configuration,

You can configure your own response page. Example :

```
<HTML ><BODY >
<#mitag:#mainmenu>
<table><caption><h3>This is my own home page</h3></caption>
  <Tr ><td VALIGN=TOP><#mitag:/granprimo.htm></td>
    <td ><#mitag:/cpu.htm></td > </TR>
</table>
<#mitag:/files.htm>
</BODY></HTML>
```

You can see a new tag called #mitag, with this special Tag you can use makros with #namemakro and other page references with /link.htm

Proxy connection.

You have a computer A with GranPrimo (Gp A) installed but you can't access to it, (because is behind a firewall or router).

You have another computer B (with Gp B) that can be access by you and the computer A.

Then Gp A can open a 'proxy' connection with Gp B.

Gp B open a random listen port that relay the http connections to Gp A (like a bridge) , and capture the result in Gp A to response.

Then you connect (with your internet browse) with Gp B (to the normal listen port).

You obtain the normal access menu to Gp B and a linked list of another Gp connected to it.

With this link (that is a random port in Gp B) you access to the another computer and emulate a full connection to computer A that can't be access by you.

A special situation is that you can't access to computer A, but computer A can access to you. Then your own computer can be like Gp B

Gp A open a 'proxy' connection to you. Then , you access to your local GP (with 127.0.0.1 direction) and can link with computer A.

SpyObjects in depth

Basically, spyobjects unit provide a render html engine, for construction of dynamic complex HTML pages. With extension to handle remote control commands.

Some of html codes are coded like enum. types.

For example, html tags like <td>, <text> , <script> are using like a enum. type called TlinkTag, for convention this typed are building with the originiar html tag word with 'LT' added in the begin, (example TABLE tag is called **ltTable** by spyobjects engine).

```
tlinkTag = (ltnone, ltp, ltt, ltth_,ltr, ltop, ltul, lti, ltbb,  
            ltcap,lttable_,lform,lselect,ltextarea,lhtml,lthead,  
            ltitle,ltbody,lframeset,lscript,lText,linput,  
            ltButton,lspan,ltoptio,ltdiv,limg,ltaa);
```

(note : some tags with only one char, like <a> or are coded like ltaa and ltbb).

Some params

'size','border','VALIGN','bgcolor','name','method','action','width','value','align','type'... have a integer representation too. Then addparam work both with string label and enum types. You can obtain the query data fields by calling queryk (enumparam) function.

Because is multilingual, some words or expression, are coded in integer constant too.

The base object for render is **tspyhtml** .

which function is similar to IntraWeb TIWHTMLTag but there is some differences,

use TlinkTag types instead strings .

- This is good for fast construction, (is better handle enum types than string types). The render instruction make the rest.

- Is more secure for typed bugs.

(If String construction, you can write some like

```
mySH := tSpyHtml.createtag ('TABNLE');
```

```
// an error in 'n' for fast type, you can execute the program, but it don't work
```

```
mySH := tSpyHtml.createtag (ltTABNLE);
```

```
// I can't compile the program, then I fix the error and run and work.
```

Have some extension for handle integer constant like string.

- Example. mySH.contents.addtagk (lth5,kstrWellcome);

Say make a <h5> tag with the constant integer KstrWellcome (which represent a 'Wellcome' or 'Bienvenido' string it depending of language)

Table extension for optimal code.

- mySH.contents.addcell ('my cell')

instead

```
with mySH.Contents.AddTag('TD') do
```

```
Contents.AddText('my cell');
```

TspyPrimo encapsulate the kernel for build dynamic pages, answer to normal 'string like' http commands in different ways (but web compatible) , handle system web commands (like mouse movement, clicks...) , handle the list of thread process which answer to the system... and much more.

That is, encapsulate all the features of a remote control web server but without real connections. Then, you can use TspyPrimo in your system (because it not interferes with your communication system). In this way , I develop 3 extension for TspyPrimo.

TgranPrimo which is a stand-alone HTTP TCP/IP server (extend TspyPrimo)

TspyIndy which is a Indy adaptation (extend TidhttpServer)

TspyIntraWeb which is a IntraWeb adaptation.

TspyPrimo use then **TmiResponserequest** wich is similar to encapsulate the TidHTTPRequestInfo and TidHTTPResponseInfo functionally , and use TspyHtml streams for handle responses.

Another important classes are TmiWebActionItem and TspyWebActions that encapsulate the commands path recognition and add new commands handlers.

There is some example of how to use TmiResponserequest and TmiWebActionItem in aliasweb.pas unit.

Finally there is a base thread class TspyThread which provide a manage thread function in a threadlist TmiThreadList where is inserted all the TspyThread descendant.

The TspyThread descendant are process that work in different threads. Basically, in Tgranprimo component there is several communication TspyThread extensions. Listener threads for incoming connections, Answer threads for process the answer to this connections, Clients or Host threads for not keep-alive connections between different Gp systems, sending e-mail connections, IRC connections...

Remote Java Desktop

In **Remote Java Desktop** function, GranPrimo respond a Html page with a Java Class reference to an applet called spyviewer. This applet provide soft screen refresh , mouse movement & click, drag & drop emulation, full remote key events. (That is, things heave to make in normal JavaScript DHTML code).

When you use it, you must have a bit of patience, because the screen response to your events are relaying to another computer and wait the response.

You must be carefully in Drag&Drop operations, because you don't see the drag border.

But the fast of the refresh it depend of the net connection and server performance.

When you finish with the work in this page, is better that you change the page or leave the web browser, because spyviewer applet continue the refresh operations with remote system while you are in this page.