Subject: Just one instance of application running (SingleApp)
Posted by Sc0rch on Mon, 25 Jan 2010 13:42:03 GMT
View Forum Message <> Reply to Message

Header:

```
#ifndef SINGLE_APP_H
#define SINGLE_APP_H

#include <CtrlLib/CtrlLib.h>
using namespace Upp;

class UniqueWindow : public TopWindow
{
public:
 typedef UniqueWindow CLASSNAME;
 UniqueWindow(){}
};

inline bool CreateSingleApp(String name, String unique, bool message = true)
{
#ifdef PLATFORM_WIN32
 name << unique;
 if(::FindWindow(NULL, name))
 {
  if (message)
   Exclamation("Another instance of application already exists!");
  return false;
 }
 Single<UniqueWindow>().SetRect(-1, -1, 1, 1);
 Single<UniqueWindow>().Hide();
 Single<UniqueWindow>().Title(name);
 Single<UniqueWindow>().Open();
#endif
 return true;
}

#endif // SINGLE_APP_H
```

Using:

```
GUI_APP_MAIN
{
 if (!CreateSingleApp("SingleApp Test", "##SingleApp##1.0"))
  return;

 MainWindow().Run();
```

```
}
```

File Attachments

1) SingleApp.rar, downloaded 339 times

Subject: Re: Just one instance of application running (SingleApp)
Posted by koldo on Mon, 25 Jan 2010 15:11:36 GMT
View Forum Message <> Reply to Message

Sc0rch wrote on Mon, 25 January 2010 14:42Header:

```
#ifndef SINGLE_APP_H
#define SINGLE_APP_H

#include <CtrlLib/CtrlLib.h>
using namespace Upp;

class UniqueWindow : public TopWindow
{
public:
 typedef UniqueWindow CLASSNAME;
 UniqueWindow(){}
};

inline bool CreateSingleApp(String name, String unique, bool message = true)
{
#ifdef PLATFORM_WIN32
 name << unique;
 if(::FindWindow(NULL, name))
 {
  if (message)
   Exclamation("Another instance of application already exists!");
  return false;
 }
 Single<UniqueWindow>().SetRect(-1, -1, 1, 1);
 Single<UniqueWindow>().Hide();
 Single<UniqueWindow>().Title(name);
 Single<UniqueWindow>().Open();
#endif
 return true;
}

#endif // SINGLE_APP_H
```

Using:

```
GUI_APP_MAIN
{
 if (!CreateSingleApp("SingleApp Test", "##SingleApp##1.0"))
  return;

 MainWindow().Run();
}
```

Hello ScOrch

If you use GetWindowIdFromCaption() (SysInfo package) instead of FindWindow, this code will serve you for Linux too.

Sorry for the propaganda

Best regards
Koldo

Subject: Re: Just one instance of application running (SingleApp)
Posted by Sc0rch on Mon, 25 Jan 2010 17:09:31 GMT
View Forum Message <> Reply to Message

koldo wrote on Mon, 25 January 2010 21:11
Hello Sc0rch

If you use GetWindowIdFromCaption() (SysInfo package) instead of FindWindow, this code will serve you for Linux too.

Sorry for the propaganda

Best regards
Koldo

Hmm, function not work, what I'm doing wrong?

```
if (GetWindowIdFromCaption(name, true) >= 0)
 {
  if (message)
   Exclamation("Another instance of application already exists!");
  return false;
 }
```

Subject: Re: Just one instance of application running (SingleApp)
Posted by koldo on Mon, 25 Jan 2010 21:52:23 GMT
View Forum Message <> Reply to Message

Sc0rch wrote on Mon, 25 January 2010 18:09koldo wrote on Mon, 25 January 2010 21:11
Hello Sc0rch

If you use GetWindowIdFromCaption() (SysInfo package) instead of FindWindow, this code will serve you for Linux too.

Sorry for the propaganda

Best regards
Koldo

Hmm, function not work, what I'm doing wrong?

```
if (GetWindowIdFromCaption(name, true) >= 0)
 {
  if (message)
   Exclamation("Another instance of application already exists!");
  return false;
 }
```

Hello Sc0rch

GetWindowIdFromCaption(name, true) tries to find a window with title == name.

GetWindowIdFromCaption(name, false) will get the window handle of the window with a title that just contains name.

If it does not work could you give me more details ?.

Best regards
Koldo

Subject: Re: Just one instance of application running (SingleApp)
Posted by Sc0rch on Tue, 26 Jan 2010 04:41:44 GMT
View Forum Message <> Reply to Message

You said:
koldo wrote on Tue, 26 January 2010 03:52
Hello Sc0rch

GetWindowIdFromCaption(name, true) tries to find a window with title == name.

GetWindowIdFromCaption(name, false) will get the window handle of the window with a title that just contains name.

If it does not work could you give me more details ?.

Best regards
Koldo

I've tested both variants, works well for direct use, like:

```
TopWindow wnd;
wnd.Title("Test");
wnd.Open();

DUMP(GetWindowIdFromCaption("Test"));
```

But, not working in this code:

```
#include <CtrlLib/CtrlLib.h>
#include <SysInfo/SysInfo.h>
using namespace Upp;

class UniqueWindow : public TopWindow
{
public:
 UniqueWindow() {}
};

inline bool CreateSingleApp(const String& unique, const String& message)
{
 if (GetWindowIdFromCaption(unique, false) >= 0)
 {
  if (!message.IsEmpty())
   Exclamation(message);
  return false;
 }

 Single<UniqueWindow>().Title(unique).SetRect(-1, -1, 1, 1);
 Single<UniqueWindow>().Hide();
 Single<UniqueWindow>().Open();
 return true;
}
```

```
GUI_APP_MAIN
{
 if (!CreateSingleApp("SingleApp Test##SingleApp##1.0", "Another instance of application "
  "already exists!"))
 {
  return;
 }

 TopWindow wnd;
 wnd.SetRect(Size(200, 100));
 wnd.Run();
}
```

Test, and say, please, maybe it not works only for me.

Subject: Re: Just one instance of application running (SingleApp)
Posted by koldo on Tue, 26 Jan 2010 08:37:11 GMT
View Forum Message <> Reply to Message

Hello Sc0rch

I have compiled your original version and final version and none of them work for me

I have changed them a little and now they work for me:

First version (Windows only):

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;

class UniqueWindow : public TopWindow
{
public:
 typedef UniqueWindow CLASSNAME;
 UniqueWindow() {
  Title("SingleApp Test");
 }
 bool IsSingleApp() {
  if(::FindWindow(NULL, GetTitle().ToString()))
   return false;
  return true;
 }
};

GUI_APP_MAIN
{
```

```
 if (!UniqueWindow().IsSingleApp()) {
  Exclamation("Another instance of application already exists!");
  return;
 }
 UniqueWindow().Run();
}
```

Second version (Windows and Linux):

```
#include <CtrlLib/CtrlLib.h>
#include <SysInfo/SysInfo.h>
using namespace Upp;

class UniqueWindow : public TopWindow
{
public:
 typedef UniqueWindow CLASSNAME;
 UniqueWindow() {
  Title("SingleApp Test");
 }
 bool IsSingleApp() {
  if(GetWindowIdFromCaption(GetTitle().ToString()) > 0)
   return false;
  return true;
 }
};

GUI_APP_MAIN
{
 if (!UniqueWindow().IsSingleApp()) {
  Exclamation("Another instance of application already exists!");
  return;
 }
 UniqueWindow().Run();
}
```

Only change is to add #include <SysInfo/SysInfo.h> and instedad of FindWindow() I have used GetWindowIdFromCaption().

Best regards
Koldo

Subject: Re: Just one instance of application running (SingleApp)
Posted by Sc0rch on Tue, 26 Jan 2010 10:11:16 GMT
View Forum Message <> Reply to Message

Eurika, Koldo! Function GetWindowIdFromCaption works only for visible windows! So your variant is better, thanks!

Code is clear, but I think this:

```
 bool IsSingleApp() {
  return GetWindowIdFromCaption(GetTitle().ToString()) <= 0;
 }
```

looks a little bit better than:

```
 bool IsSingleApp() {
  if(GetWindowIdFromCaption(GetTitle().ToString()) > 0)
   return false;
  return true;
 }
```

or not?

Best regards,
Anton

Subject: Re: Just one instance of application running (SingleApp)
Posted by koldo on Tue, 26 Jan 2010 10:17:44 GMT
View Forum Message <> Reply to Message

Sc0rch wrote on Tue, 26 January 2010 11:11Eurika, Koldo! Function GetWindowIdFromCaption works only for visible windows! So your variant is better, thanks!

Code is clear, but I think this:

```
 bool IsSingleApp() {
  return GetWindowIdFromCaption(GetTitle().ToString()) <= 0;
 }
```

looks a little bit better than:

```
 bool IsSingleApp() {
  if(GetWindowIdFromCaption(GetTitle().ToString()) > 0)
   return false;
  return true;
 }
```

or not?

Best regards,
Anton

Of course your is better