Subject: Dialog windows showing semi randomly on the task bar on Linux
Posted by Alboni on Wed, 26 Sep 2018 23:47:59 GMT
View Forum Message <> Reply to Message

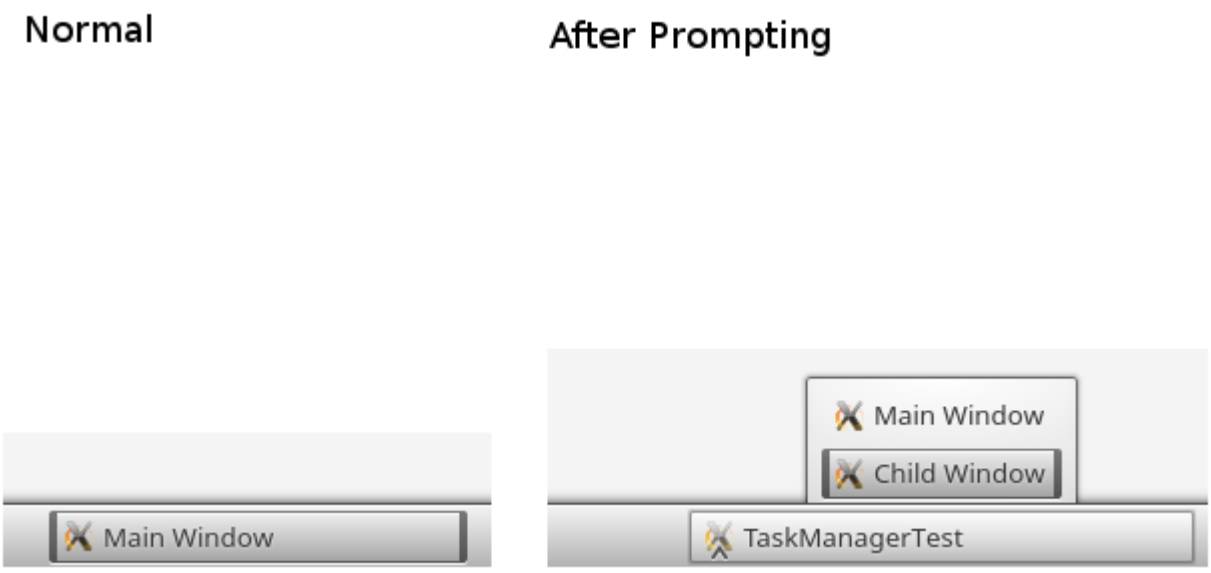
This issue happens only on Linux and obviously on a flavor of Linux with a traditional desktop with a bar. I am using kde.

The issue:
The main window of the program shows on the task bar. This is expected.
If the program executes a dialog window, the dialog window doesn't show up on the task bar. This is also expected.
If the program presents a prompt before executing the dialog, the dialog and main window show up separately on the task bar. The prompt doesn't;
This is confusing, because they can now also be activated separately and unexpected.

Upp 9251, Linux Mint 17.3 64 bit KDE. Effect is also seen on newer versions of Linux.

File Attachments

1) prompt.png, downloaded 1068 times



2) TaskManagerTest.tar.gz, downloaded 411 times

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Alboni on Thu, 27 Sep 2018 00:30:52 GMT
View Forum Message <> Reply to Message

XFCE on Xenial

File Attachments

1) xfce.png, downloaded 938 times



Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Alboni on Thu, 27 Sep 2018 12:41:20 GMT
View Forum Message <> Reply to Message

so this shows only the program on the task bar.

```
void MyApp::MyFunction()
{
  SomeDialog dlg;
  dlg.Execute();
}
```

and this shows the program and the dialog separately on the task bar

```
void MyApp::MyFunction()
{
  PromptOK("a message");
  SomeDialog dlg;
  dlg.Execute();
}
```

That has to be a bug.

But.. if I do this, it doesn't happen.

```
TimeCallback timer;
void MyApp::MyFunction1()
{
  PromptOK("a message");
  timer.Set(1, THISBACK(MyFunction2));
}
void MyApp::MyFunction2()
{
  SomeDialog dlg;
  dlg.Execute();
}
```

(and none of it happens on Windows)

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Oblivion on Thu, 27 Sep 2018 14:08:10 GMT
View Forum Message <> Reply to Message

Hello Alboni,

As far as I can see, the below function seems to be a standalone function (not a member/method of any TopWindow, or its derivative).
Therefore you are probably opening two top windows sequentially. hence the two taskbar entries.

```
void MyFunction()
{
  PromptOK("a message"); // The first topmost window.
  SomeDialog dlg;
  dlg.Execute();        // The second topmost window.
}
```

Best regards,
Oblivion

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Alboni on Thu, 27 Sep 2018 14:12:31 GMT
View Forum Message <> Reply to Message

They are part of a class.
I will change the message to reflect this.
The full example can be downloaded in the first message.

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Oblivion on Thu, 27 Sep 2018 14:30:06 GMT
View Forum Message <> Reply to Message

Yep it is reproducible under Gnome too.
I'm not sure if this is an error, Mirek will know better, of course. But in the meantime all you need to do is pass the owner and then execute/run the child.

This should work:

```
#include <CtrlLib/CtrlLib.h>

using namespace Upp;

#define LAYOUTFILE <TaskManagerTest/TaskManagerTest.lay>
```

```
#include <CtrlCore/lay.h>

class TaskManagerTest : public WithTaskManagerTestLayout<TopWindow> {
public:
 typedef TaskManagerTest CLASSNAME;
 TaskManagerTest()
 {
  CtrlLayout(*this, "Main Window");
  button1<<=THISBACK1(Press, false);
  button2<<=THISBACK1(Press, true);
 }
 void Press(bool ok)
 {
  if (ok)
   PromptOK("Notice the child window having a seperate item on the taskbar after closing this
message");

  WithChildLayout<TopWindow> w;
  CtrlLayout(w, "Child Window");
  w.Open(this);
  w.Execute();
 }
};


GUI_APP_MAIN
{
 TaskManagerTest().Run();
}
```

Best regards,
Oblivion

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by Alboni on Thu, 27 Sep 2018 15:12:39 GMT
View Forum Message <> Reply to Message

Thanks. That works in the example, but not in my real world app. Adding dialog.Open(this) before Execute() makes the dialog appear on the wrong monitor and then the app segfaults.
Weird

Subject: Re: Dialog windows showing semi randomly on the task bar on Linux
Posted by mirek on Wed, 31 Oct 2018 13:52:26 GMT

Hopefully fixed, please try.

That said, Oblivion suggested correct fix - the real cause of the problem is that if you do not supply "owner" with Open, current active window is used as "owner". Here, the Prompt confused U++ to lose track about what current active window is.