

User Interface - Bug #2763

Bug # 2677 (New): fix drawing and functional differences between P2J GUI and 4GL GUI

3D drawing of window borders is regressed in 1811s revision 10994

10/19/2015 05:43 PM - Greg Shah

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Hynek Cihlar	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	GUI Support for a Complex ADM2 App	case_num:	
billable:	No	version:	
vendor_id:	GCD		
Description			

History

#1 - 10/19/2015 05:44 PM - Greg Shah

After rebase to trunk 10948 (the new dialog box support), 1811s (revision 10994) shows a regression where the 3D border of windows is no longer being drawn properly.

#2 - 10/19/2015 06:00 PM - Greg Shah

Interestingly, this affect only seems to occur while tinyInput() is running. But it causes both the dialog box and the owning window's border to be drawn without 3D effects.

Once you dismiss the dialog, the main window draws its 3D border again.

If you run a testcase that doesn't have a dialog, then the 3D border is drawn fine (e.g. hello.p).

#3 - 10/19/2015 06:05 PM - Hynek Cihlar

Greg Shah wrote:

Interestingly, this affect only seems to occur while tinyInput() is running. But it causes both the dialog box and the owning window's border to be drawn without 3D effects.

Once you dismiss the dialog, the main window draws its 3D border again.

If you run a testcase that doesn't have a dialog, then the 3D border is drawn fine (e.g. hello.p).

I will look at this. Btw. 3D border should be drawn in DIALOG-BOX and ALERT-BOX.

#4 - 10/20/2015 04:34 PM - Hynek Cihlar

Eugenie, WindowGuilImpl.draw() calls super.draw() twice. First before drawBorder() and second time after @drawBorder(). This causes the border to be redrawn and the 3D visuals drawn over. Can you please confirm the two calls to super were not intentional?

#5 - 10/20/2015 04:38 PM - Constantin Asofiei

Hynek Cihlar wrote:

Eugenie, WindowGuilImpl.draw() calls super.draw() twice. First before drawBorder() and second time after @drawBorder(). This causes the border to be redrawn and the 3D visuals drawn over. Can you please confirm the two calls to super were not intentional?

I think this is a rebase merge issue: super.draw() needs to be called after drawBorder().

#6 - 10/20/2015 04:44 PM - Hynek Cihlar

Constantin Asofiei wrote:

Hynek Cihlar wrote:

Eugenie, WindowGuilImpl.draw() calls super.draw() twice. First before drawBorder() and second time after @drawBorder(). This causes the border to be redrawn and the 3D visuals drawn over. Can you please confirm the two calls to super were not intentional?

I think this is a rebase merge issue: super.draw() needs to be called after drawBorder().

It should be called before or else the 3D border would be drawn over.

#7 - 10/20/2015 04:53 PM - Eugenie Lyzenko

Constantin Asofiei wrote:

Hynek Cihlar wrote:

Eugenie, WindowGuilImpl.draw() calls super.draw() twice. First before drawBorder() and second time after @drawBorder(). This causes the border to be redrawn and the 3D visuals drawn over. Can you please confirm the two calls to super were not intentional?

I think this is a rebase merge issue: super.draw() needs to be called after drawBorder().

I have no idea why this code is so weird now. And I'm not sure this came from rebase, may be something happens between:

```
...
** 046 HC 20151013 Changes to extend GUI multi-window focus management and ACTIVE-WINDOW system
**                handle processing to modal windows.
** 047 CA 20151015 Fixed window resize via attributes (they target the workspace, not the full
**                window) .
...
```

I have working copy before 047 with proper version of this class so I'm telling for sure.

So I confirm to have no intention from my side to make this change for any purpose. This looks really strange because if border overlaps super.draw() why to overlap the border again?

#8 - 10/20/2015 05:00 PM - Eugenie Lyzenko

Hynek Cihlar wrote:

Constantin Asofiei wrote:
It should be called before or else the 3D border would be drawn over.

I think the sequence is important if the calls have intersections. If so the super.draw() is the first, the border - second.

#9 - 10/20/2015 05:07 PM - Hynek Cihlar

1811s revision 11004 fixes the disappearing 3D border. It also adds 3D border to ALERT-BOX and DIALOG-BOX.

#10 - 10/20/2015 05:20 PM - Hynek Cihlar

- % Done changed from 0 to 100

#11 - 10/20/2015 06:05 PM - Greg Shah

There is one regression caused by 11004. Try simpler_alert_box.p and after selecting one of the buttons, try to dismiss the main window that is waiting at a "Procedure complete. Press space bar to continue." prompt. It doesn't work. In fact, you have to kill the client forcibly to get rid of it.

This same feature works fine if you don't bring up an alert-box.

#12 - 10/21/2015 05:59 PM - Hynek Cihlar

Greg Shah wrote:

There is one regression caused by 11004. Try `simpler_alert_box.p` and after selecting one of the buttons, try to dismiss the main window that is waiting at a "Procedure complete. Press space bar to continue." prompt. It doesn't work. In fact, you have to kill the client forcibly to get rid of it.

This same feature works fine if you don't bring up an alert-box.

This is due a regression which broke event processing in alert-box, solved in [#2770](#).

#13 - 10/21/2015 06:00 PM - Hynek Cihlar

Hynek Cihlar wrote:

Greg Shah wrote:

There is one regression caused by 11004. Try `simpler_alert_box.p` and after selecting one of the buttons, try to dismiss the main window that is waiting at a "Procedure complete. Press space bar to continue." prompt. It doesn't work. In fact, you have to kill the client forcibly to get rid of it.

This same feature works fine if you don't bring up an alert-box.

This is due a regression which broke event processing in alert-box, solved in [#2770](#).

I meant being solved in [#2770](#).

#14 - 10/22/2015 08:56 AM - Greg Shah

- Status changed from New to Closed

#15 - 11/16/2016 12:13 PM - Greg Shah

- Target version changed from Milestone 12 to GUI Support for a Complex ADM2 App