

User Interface - Bug #2861

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

fix widget z-order inside frame, when focus is switching - there are differences between ChUI and GUI clients

11/20/2015 05:20 AM - Constantin Asofiei

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Constantin Asofiei	% 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 - 11/20/2015 05:21 AM - Constantin Asofiei

When focus is switching between overlapped widgets inside the same frame (these widgets may be a nested frame, too), in ChUI the widget/nested frame reflecting the new focus is brought on top, in z-order. In GUI, the z-order does not get changed.

See uast/frame-z-order/nested-frames.p

#2 - 11/25/2015 04:20 AM - Constantin Asofiei

- Status changed from New to WIP
- Assignee set to Constantin Asofiei

#3 - 11/25/2015 05:55 AM - Constantin Asofiei

The issues in nested_frames.p are not exactly z-order related. The problem is with explicit widget drawing, when there are stacked widgets, on the same location, and the widget is not the one on top (there are other widgets intersecting it). Code like this, in ThinClient.processEventsWorker:15166:

```
src.draw();
src.requestSync();
```

draws the widget directly, regardless if there are other widgets on top of it. This can be seen also in TC.widgetDrawingOn/Off usage.

I think the solution is to add a clipping rectangle to the driver, with only the widget's visible part.

#4 - 11/25/2015 10:47 AM - Constantin Asofiei

No, adding a clipping rectangle with only the widget's visible part will not work - as the visible part may not be a rectangle at all!

So, the approach is this: all direct drawing in ThinClient is replaced with eventDrawingBracket and repaint() calls, with widgetDrawingOn() and widgetDrawingOff() APIs being obsolete (thus removed).

This fixes nested-frames.p, I'm testing the GUI screen now. If the screen is OK, next I will commit these changes and start a runtime testing.

#5 - 11/25/2015 12:53 PM - Constantin Asofiei

- *Status changed from WIP to Review*

2677b rev 10966 contains a fix for this and #2878.

I'm starting runtime testing with this revision.

#6 - 11/25/2015 01:08 PM - Greg Shah

Code Review Task Branch 2677b Revision 10966

The changes are good. The resulting code is much cleaner. Hopefully the regression testing will not find any issues.

#7 - 11/26/2015 03:06 AM - Constantin Asofiei

The changes exposed an existing issue in Frame.drawDownFrame: frameScroll.minimumSize reports incorrect height; instead, frameScroll.getScrollableSize needs to be used.

#8 - 11/26/2015 03:08 AM - Constantin Asofiei

2677b rev 10968 fixes the regression in note 7. Runtime testing is in progress again.

#9 - 11/26/2015 09:36 AM - Constantin Asofiei

Runtime testing passed with only false negatives (I assume). I'm running main part again to confirm.

#10 - 11/26/2015 02:14 PM - Constantin Asofiei

Second run has only a common failure: GSO 17's gso_17.sum report has an extra row... I'm inclined to assume is a false negative, I don't see how my changes can impact this.

#11 - 11/27/2015 06:53 AM - Greg Shah

- *% Done changed from 0 to 100*

- *Status changed from Review to Closed*

Agreed, this is done.

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

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