

User Interface - Bug #2715

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

demo/calc-static.p drawing differences as compared to the 4GL version

09/14/2015 05:38 PM - Greg Shah

Status: Closed	Start date:
Priority: Normal	Due date:
Assignee: Greg Shah	% Done: 100%
Category:	Estimated time: 0.00 hour
Target version: GUI Support for a Complex ADM2 App	case_num:
billable: No	
vendor_id: GCD	
Description	
Related issues:	
Related to User Interface - Bug #2730: taller than default fill-in height cau...	Closed
Related to User Interface - Bug #2727: fill-in text selection is broken	New

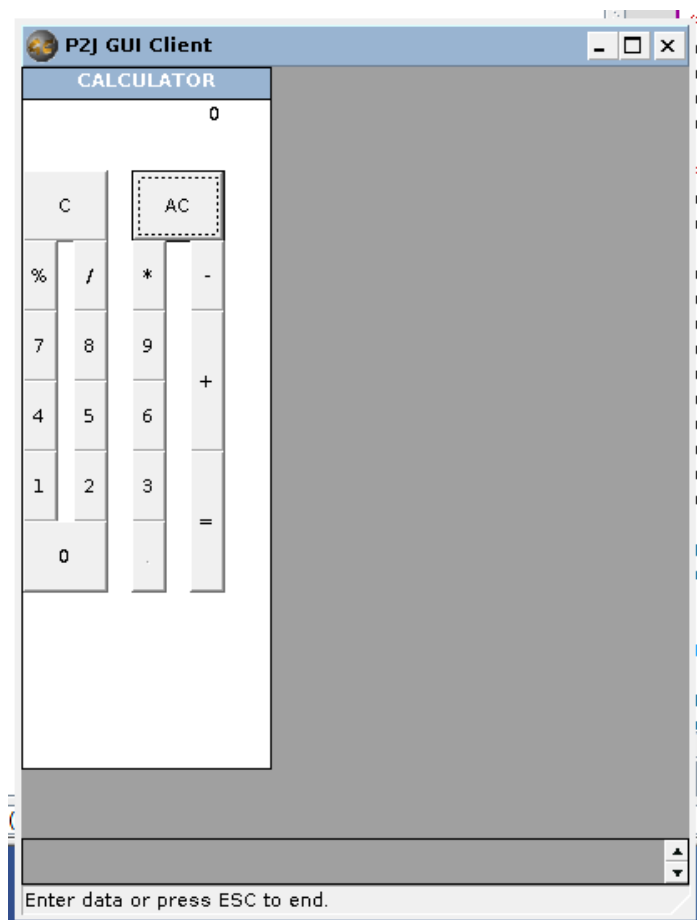
History

#1 - 09/14/2015 05:56 PM - Greg Shah

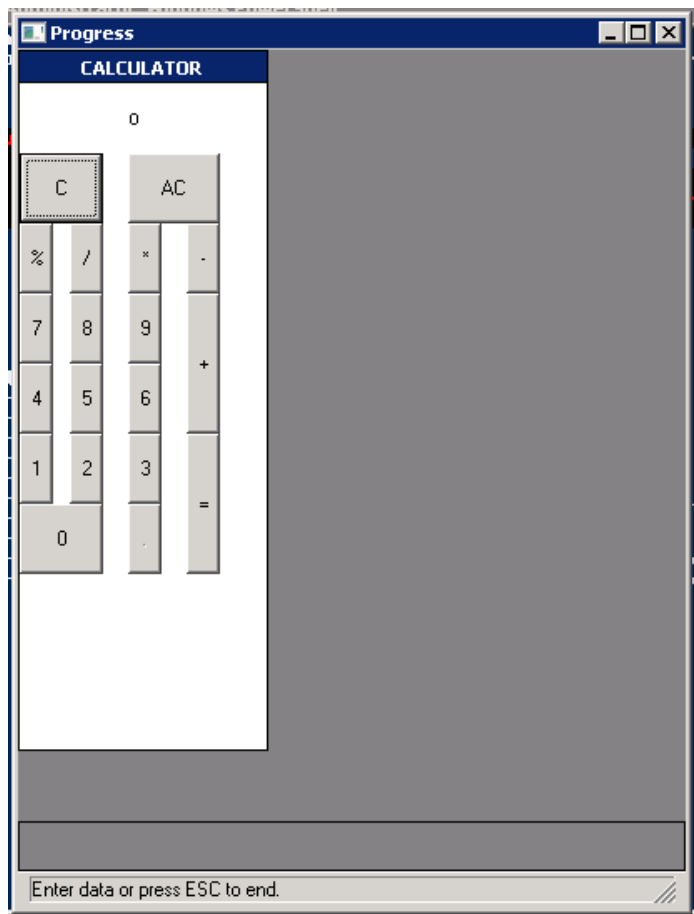
- File calc-static.p_in_p2j_swing_20150914.png added

- File calc-static.p_in_4gl_20150914.png added

Consider this output from demo/calc-static.p as executed in the P2J swing client in trunk rev 10935:



Now consider the same program executed in the 4GL:



I see the following issues:

- wrong button is highlighted (C in 4GL, AC in P2J)
- interior button selection highlight is a dashed rectangle in P2J but is a small dotted rectangle in the 4GL
- the buttons are drawn such that the exterior button selection highlight (a solid black 1 px rectangle around the entire outside of the button) is partially covered on the bottom of the button (it is overdrawn by the buttons below it) but in the 4GL it is completely visible
- button background color is wrong
- active window and active frame title background and foreground colors are wrong
- window border/status area border color is wrong (this is probably the same as the button background color)
- main content pane color is wrong
- the output field that displays the numeric value (0 in the images) is either positioned wrong and/or it is justified incorrectly
- the message area should not have scrollbar/buttons when there is no overflowing content

I've only noted issues which are not being worked in other tasks. The border 3D and the status area border drawing differences are being dealt with in [#2679](#) and [#2680](#) respectively. The color differences can be dealt with here.

LE: I am marking off the issues as we fix them.

#2 - 09/15/2015 10:41 AM - Greg Shah

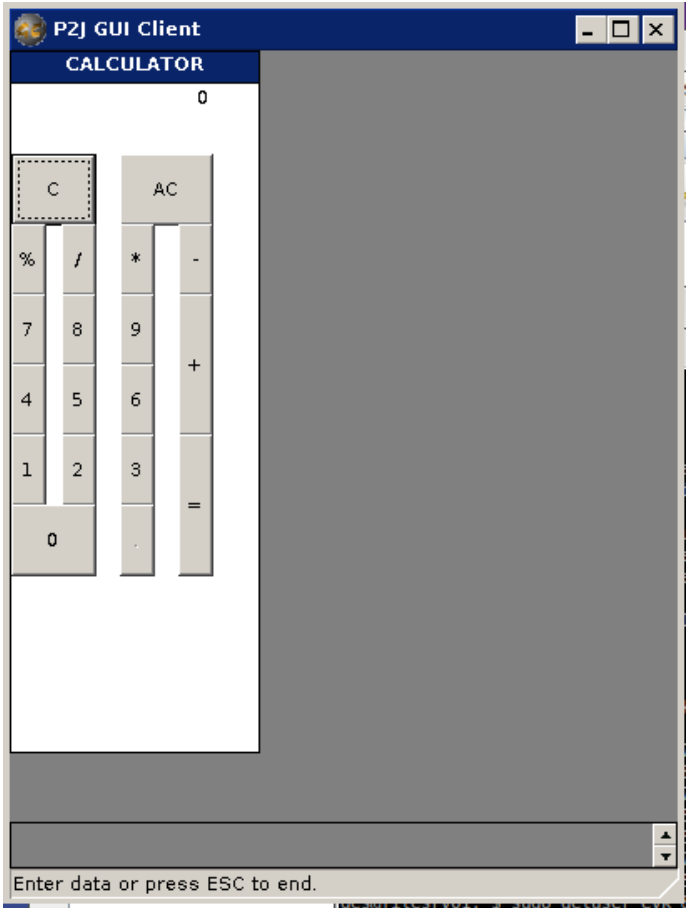
FYI, the same problems exist in task branch 1811q revision 11008 for both Swing and web clients. There are no differences or additions.

#3 - 09/17/2015 12:54 PM - Greg Shah

- File *calc-static.p_in_p2j_swing_w_fixed_colors_20150917.png* added

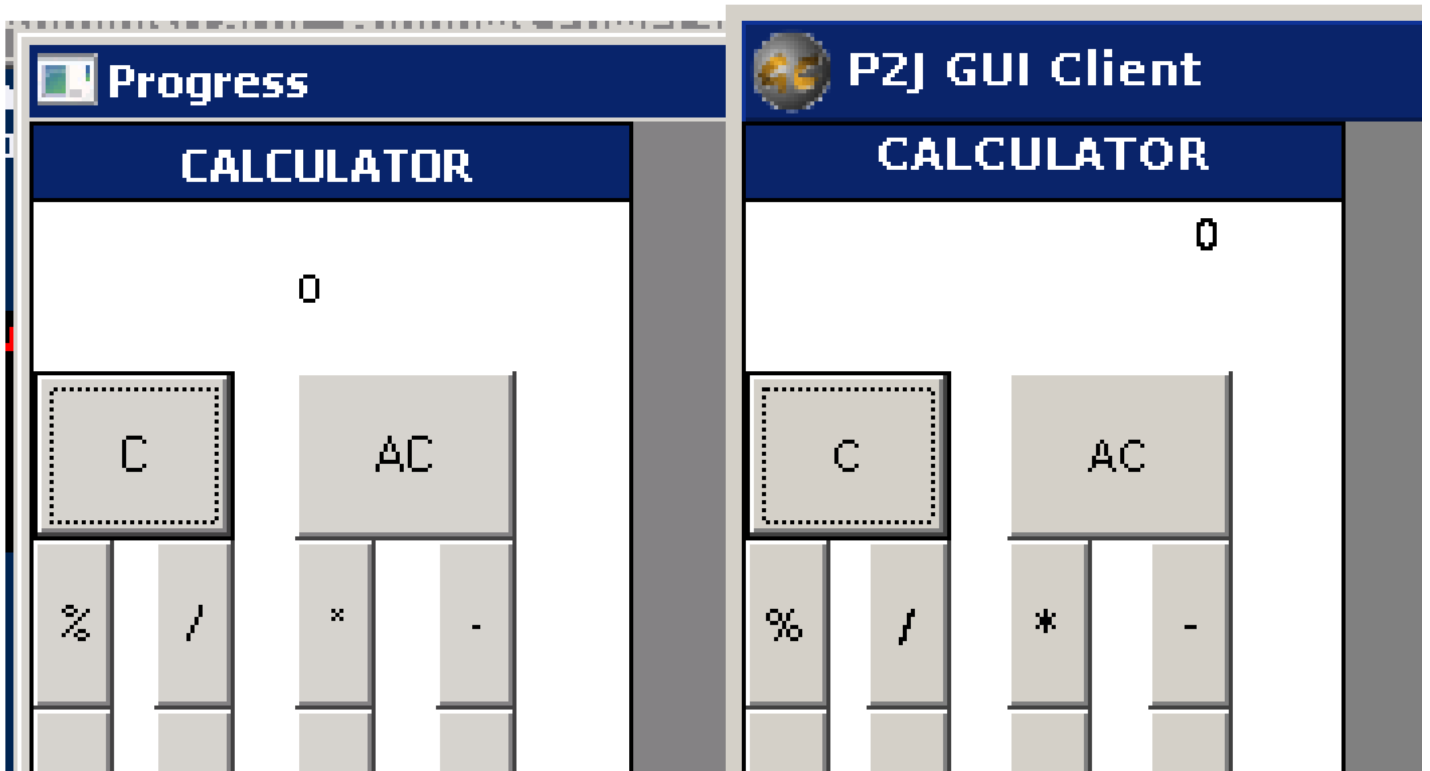
I have updated the system colors to match the windows classic theme (I captured them with the new tools/dump_syscolor.c program). Based on these changes, the colors now match exactly. I will edit note 1 to cross out the items that are fixed.

Here is the P2J Swing result:



Task branch 1181q revision 11017 has changes that fix the button line stroke and properly implemented the height calculations which previously were 1 pixel too much (the bottom of the button was 1 pixel further down than in the 4GL).

Here is the result:



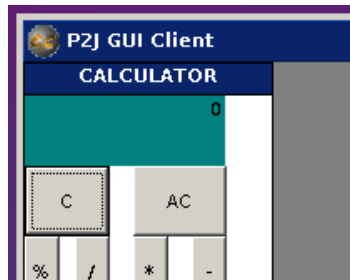
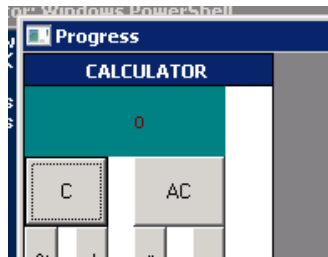
Please note that the button interior focus highlight (small dotted stroked rectangle) is not pixel-for-pixel identical. That is the only remaining difference. We will fix that in [#2724](#).

#5 - 09/18/2015 09:23 AM - Greg Shah

- File `calc-static.p_in_p2j_swing_w_result_widget_changed_bgcolor_20150918.png` added

- File `calc-static.p_in_4gl_w_result_widget_changed_bgcolor_20150918.png` added

I changed the bgcolor and fgcolor for the result fillin and that shows that the size and position of the fillin are both correct.



The problem is where we are drawing the text within the widget.

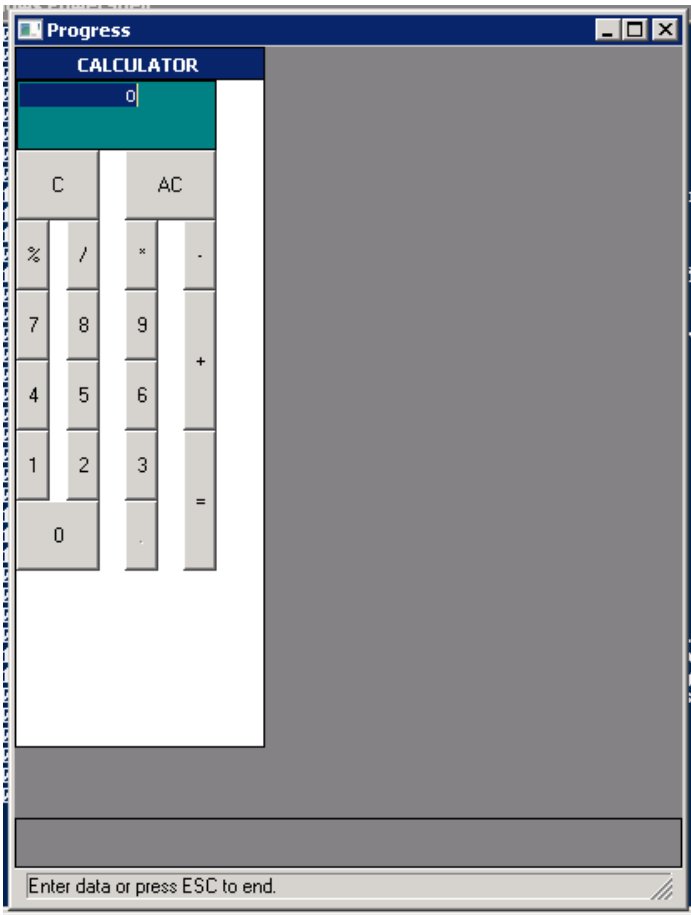
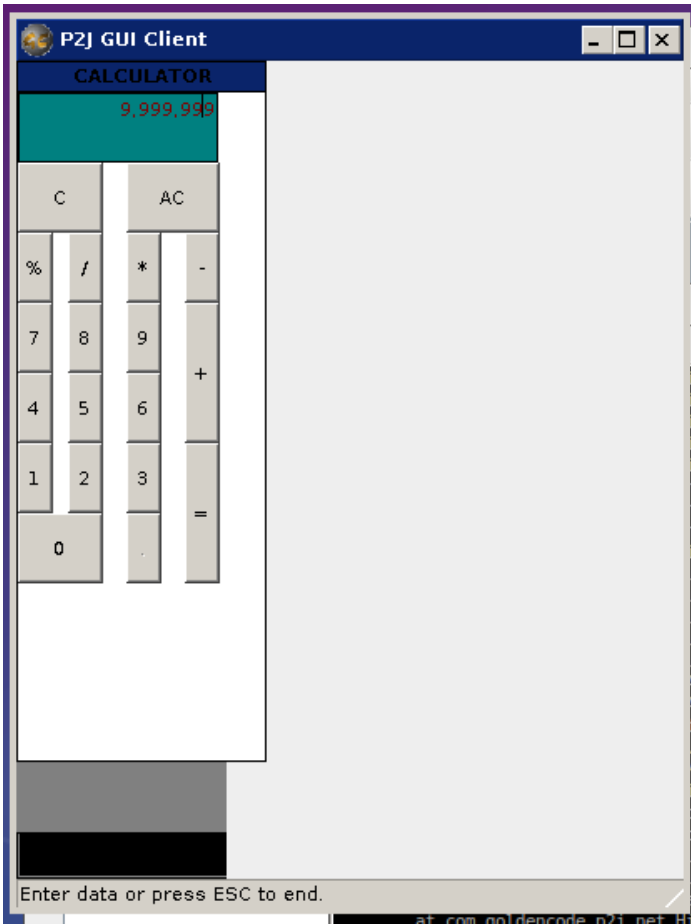
It also seems like we are not honoring the fgcolor properly (or we have different default colors than the windev01 test system).

#6 - 09/18/2015 11:08 AM - Constantin Asofiei

Greg, I think this is a problem of aligning the "fill-in content". It should help making the `tfRes FILL-IN` editable and see where the editable part gets aligned. The problem might be in `FillInGuiImpl.textNativeLocation`, for right-aligned widgets. I think `right-align` doesn't move the content to the right edge, but instead the content's draw area remains fixed - it's width is computed based on the format (the default width) and the content is right-aligned within this width.

#7 - 09/18/2015 02:48 PM - Greg Shah

I modified the testcase to leave the result fillin as enabled. This is the result:



The position of the editable portion of the fillin is different. Strangely, when the field is enabled in the 4GL, it is still aligned on the leftmost part of the fillin, but it is at the top of the vertical space. When not enabled, the editable portion displays vertically centered.

This latest version of 1811q (revision 11022) also displays the correct fgcolor.

Another deviation is that the widget selection highlight is drawn for the entire field in the 4GL but just for the current text in P2J. This cannot be seen from the screen capture above since I typed the 9,999,999 so that I could show the size of the editable area (since the highlight was too small).

#8 - 09/18/2015 02:48 PM - Greg Shah

- File *calc-static.p_in_p2j_swing_w_enabled_result_widget_changed_bgcolor_20150918.png* added

- File *calc-static.p_in_4gl_w_enabled_result_widget_changed_bgcolor_20150918.png* added

#9 - 09/18/2015 03:13 PM - Greg Shah

At least part of the problem is in `FillInGuiImpl.editableNativeDimension()`. It just takes the full physical size of the widget and subtracts the insets. This value is used to right-align the contents. Based on the other usage of that method, it may be needed to create a new method (I'm not sure).

The use of `gd.getTextWidth()` is operating on the specific text that is set in the fill-in, right? How do I find the actual editable size (based on the format string)?

#10 - 09/18/2015 03:16 PM - Constantin Asofiei

Greg Shah wrote:

At least part of the problem is in `FillInGuiImpl.editableNativeDimension()`. It just takes the full physical size of the widget and subtracts the insets. This value is used to right-align the contents. Based on the other usage of that method, it may be needed to create a new method (I'm not sure).

The use of `gd.getTextWidth()` is operating on the specific text that is set in the fill-in, right? How do I find the actual editable size (based on the format string)?

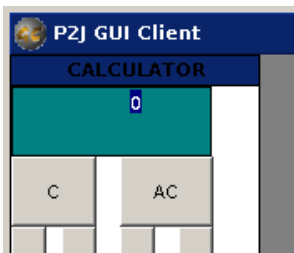
The default width of the FILL-IN is computed based on the format string in `FillInGuiImpl:protected static int nativeWidth`. I think this should be always anchored to widget's top-left corner and the fill-in content should be drawn relative to it.

#11 - 09/18/2015 05:43 PM - Greg Shah

- File *calc-static.p_in_p2j_swing_w_enabled_result_widget_changed_bgcolor_horizontal_fix_20150918.png* added

Task branch 1811q revision 11025 is an attempt to fix the positioning of right justified fill-ins that have larger widths explicitly set. The result is close horizontally to the 4GL version, though not 100% exactly correct. The vertical centering and the focus highlight are still not fixed.

Constantin: please review.



#12 - 09/18/2015 05:46 PM - Constantin Asofiei

Greg Shah wrote:

Task branch 1811q revision 11025 is an attempt to fix the positioning of right justified fill-ins that have larger widths explicitly set. The result is close > > Constantin: please review.

The changes look OK.

... the focus highlight are still not fixed.

The highlight is caused by the fact that currently right-aligned fill-in's are not left-padded with spaces.

#13 - 09/21/2015 02:40 PM - Greg Shah

As of task branch 1811q revision 11029, all the original problems noted for this task are cleared. I will open separate Redmine tasks for the vertical centering and selection highlight issues.

#14 - 09/21/2015 02:41 PM - Greg Shah

- Status changed from New to Closed

#15 - 11/10/2015 04:58 AM - Greg Shah

- % Done changed from 0 to 100

- Assignee set to Greg Shah

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

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

Files

calc-static.p_in_4gl_20150914.png	4.42 KB	09/14/2015	Greg Shah
calc-static.p_in_p2j_swing_20150914.png	6.29 KB	09/14/2015	Greg Shah
calc-static.p_in_p2j_swing_w_fixed_colors_20150917.png	7.83 KB	09/17/2015	Greg Shah
calc-static.p_button_drawing_comparison_4gl_vs_p2j_swing_after_fix_20150917.png	56.20 KB	09/17/2015	Greg Shah
calc-static.p_in_4gl_w_result_widget_changed_bgcolor_20150918.png	1.71 KB	09/18/2015	Greg Shah
calc-static.p_in_p2j_swing_w_result_widget_changed_bgcolor_20150918.png	21.75 KB	09/18/2015	Greg Shah
calc-static.p_in_4gl_w_enabled_result_widget_changed_bgcolor_20150918.png	50.98 KB	09/18/2015	Greg Shah
calc-static.p_in_p2j_swing_w_enabled_result_widget_changed_bgcolor_20150918.png	8.67 KB	09/18/2015	Greg Shah
calc-static.p_in_p2j_swing_w_enabled_result_widget_changed_bgcolor_20150918.png	2.14 KB	09/18/2015	Greg Shah