

## User Interface - Bug #2701

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

### default font text sizing issue

09/10/2015 03:18 PM - Greg Shah

<b>Status:</b> Closed	<b>Start date:</b>
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b> Constantin Asofieci	<b>% Done:</b> 100%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> GUI Support for a Complex ADM2 App	<b>case_num:</b>
<b>billable:</b> No	
<b>vendor_id:</b> GCD	
<b>Description</b>	
<b>Related issues:</b>	
Related to User Interface - Feature #1794: implement font support	<b>Closed</b>
Related to User Interface - Bug #2765: find proper freely licensed replacemen...	<b>Closed</b>
Related to User Interface - Bug #2766: implement a text metrics server process	<b>New</b>

### History

#### #1 - 09/10/2015 03:25 PM - Greg Shah

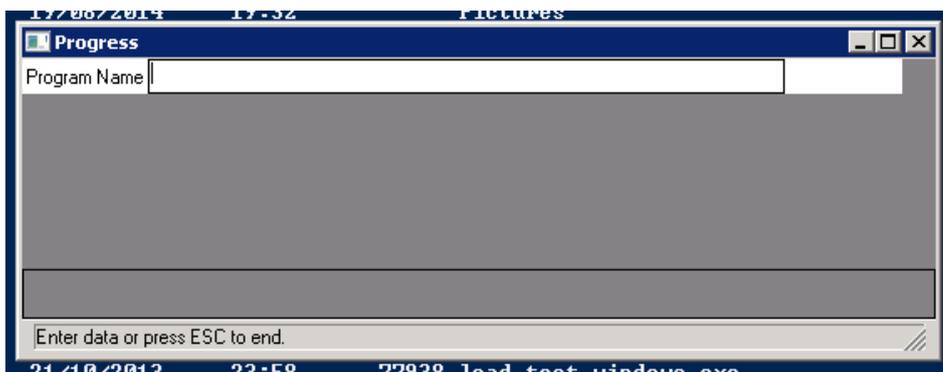
- File ask\_gui\_at\_initial\_load\_in\_p2j\_web\_and\_swing\_20150902.png added

- File ask\_gui\_at\_initial\_load\_in\_4gl\_20150902.png added

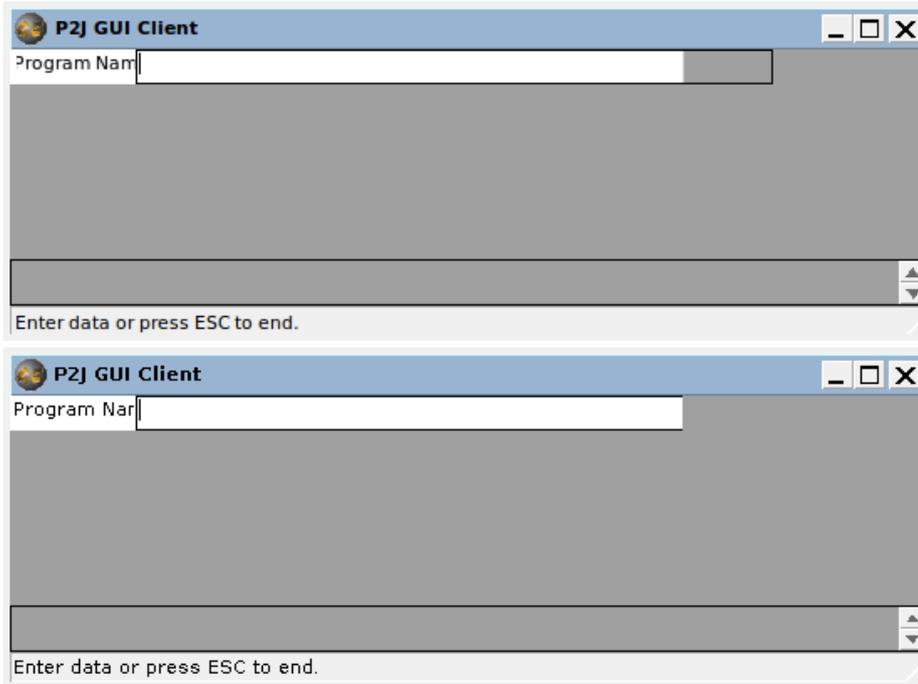
The testcases/uast/ask-gui-fake-label.p uses a non-enabled fillin as a label/static text field. The "Program Name" string is a literal in the code, so it does not dynamically change the text. In this example, the size of that field is fixed.

When I run it, I see the following results.

4GL:



P2J Web (above) and Swing (below):



Although both P2J and the 4GL have the same size for the not-a-label fillin, the text in P2J simply does not fit.

**#2 - 09/11/2015 07:44 AM - Greg Shah**

- Subject changed from "fake label" usage text sizing issue to default font text sizing issue
- File `ask_gui_at_initial_load_in_4gl_20150910.png` added
- File `ask_gui_at_initial_load_in_p2j_swing_20150910.png` added
- File `ask_gui_at_initial_load_in_p2j_web_20150910.png` added

----- Forwarded Message -----

Subject: Re: label problem or something I did wrong?  
Date: Thu, 10 Sep 2015 23:03:58 +0300  
From: Constantin Asofiei <[ca@goldencode.com](mailto:ca@goldencode.com)>  
To: [ges@goldencode.com](mailto:ges@goldencode.com)

Greg,

P2J does everything it's supposed to do, and the problem is at the driver. I'm guessing you are using the default font-table, correct? In this case, in Ubuntu or Browser, there is no "Ms Sans Serif" font (which is the DEFAULT-FONT), and Swing/Browser will default to some other font. That's why you don't see text drawn as expected.

The issue in [#2701](#) is the same as this one, the font at the driver level is missing.

To solve this, there is the pending "determine fonts compatible with the legacy bitmap fonts and register them as aliases" issue.

Thanks,  
Constantin

On 9/10/2015 10:37 PM, Greg Shah wrote:

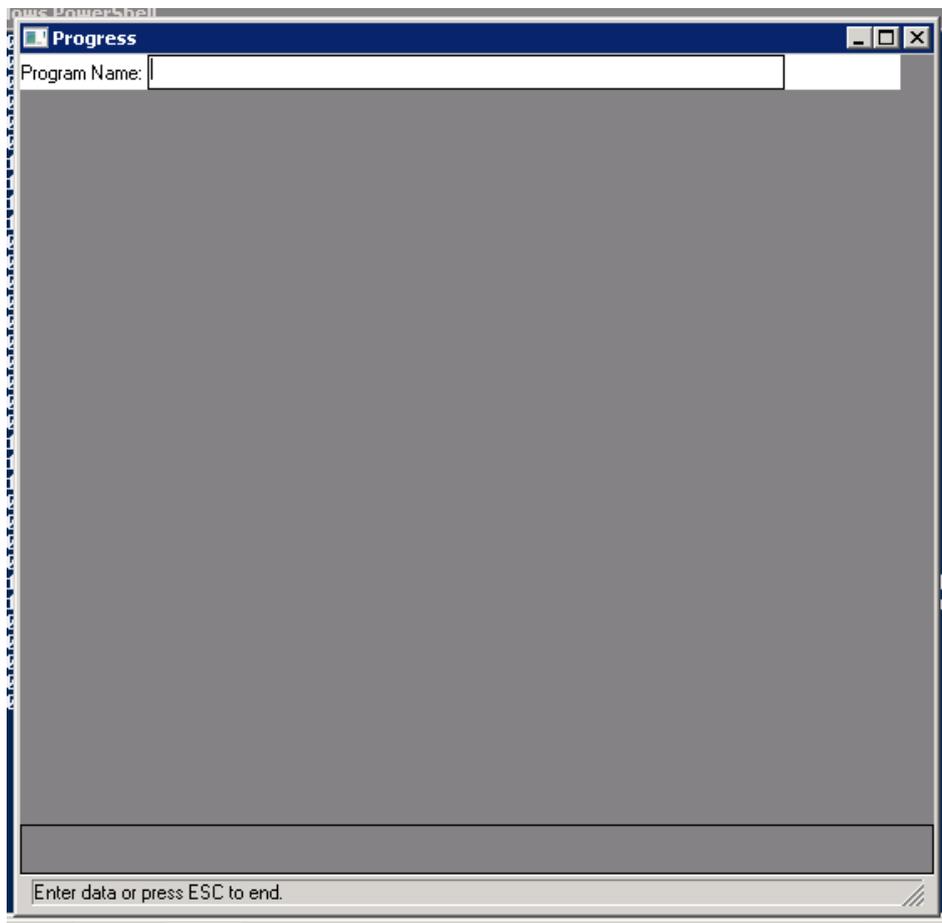
Constantin,

I'm working my way through the set of GUI regression testcases. About a week ago, I ran conversion on the set of tests. Then I captured text metrics and copied the resulting text-metrics.xml into testcases/uast/src/. I re-converted (probably not necessary) and re-jarred the testcases.jar. I looked in the jar and the new version of the text-metrics.xml is there.

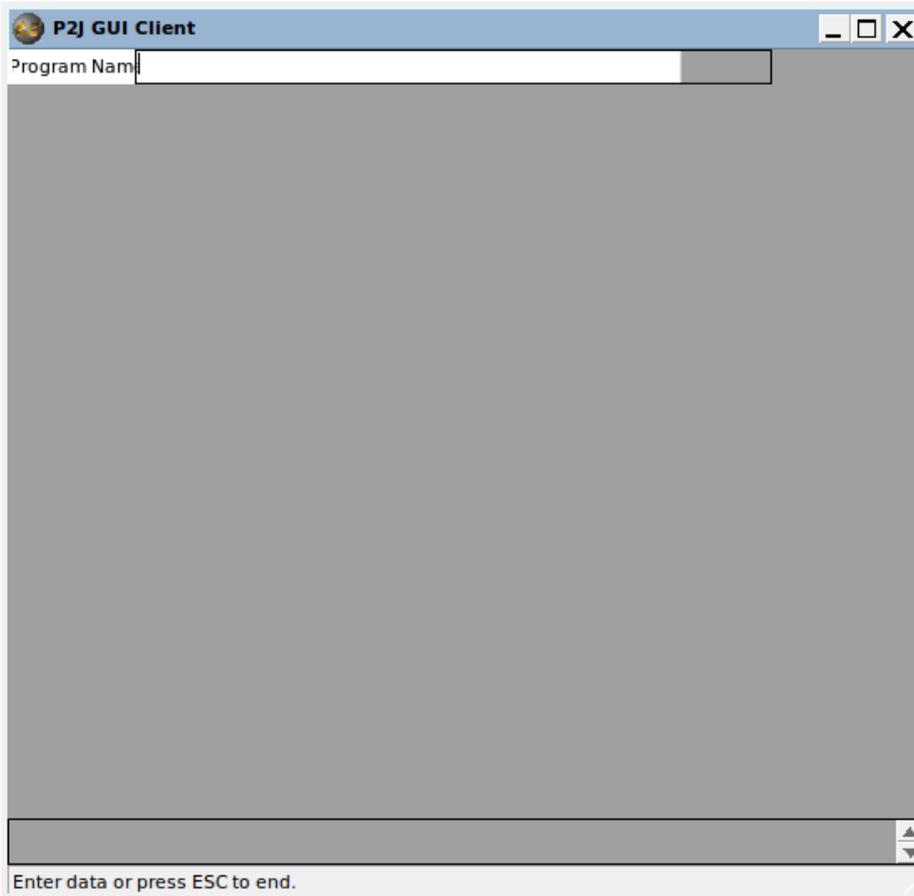
At that point, I assume that I've done everything needed to get our text metrics correct.

When I run ask-gui.p (the newly checked in version, not the one now called ask-gui-fake-label.p), I see the following results.

4GL:



P2J Web:



P2J Swing:



There are multiple problems with the sizing/drawing of the fillin and the frame background. But my question for now is: why doesn't the "Program Name" label text get sized properly?

This is not the same thing as the [#2701](#) that I just reported. I have modified ask-gui.p to use real labels and have captured text metrics for them. Attached is the text-metrics.xml that I captured (and which is in the testcases.jar). It does indeed have the "Program Name" text there.

Have I done something wrong here? Or is this just a bug?

Thanks,  
Greg

### #3 - 10/13/2015 03:47 PM - Greg Shah

- Assignee set to Constantin Asofiei

- Target version set to Milestone 12

### #4 - 10/14/2015 02:40 PM - Constantin Asofiei

1811s plus testcases project rev 1378 (almost) solved this issue. For Swing, the test is identical, but for Web, the text is anti-aliased, and its drawn width is larger than the legacy width.

testcases project rev 1378 contains custom font definitions for these fonts:

MS Sans Serif  
Courier New  
Segoe UI  
Microsoft Sans Serif  
FixedSys  
Tahoma

The custom-fonts added to simple/server/directory.xml is this:

```
<node class="container" name="custom-fonts">
  <node class="container" name="font1">
    <node class="string" name="font">
      <node-attribute name="value" value="Tahoma"/>
    </node>
    <node class="string" name="file">
      <node-attribute name="value" value="fonts/tahoma.ttf"/>
    </node>
  </node>
  <node class="container" name="font2">
    <node class="string" name="font">
      <node-attribute name="value" value="Tahoma"/>
    </node>
    <node class="boolean" name="bold">
      <node-attribute name="value" value="TRUE"/>
    </node>
    <node class="string" name="file">
      <node-attribute name="value" value="fonts/tahomabd.ttf"/>
    </node>
  </node>
  <node class="container" name="font3">
    <node class="string" name="font">
      <node-attribute name="value" value="Courier New"/>
    </node>
    <node class="string" name="file">
      <node-attribute name="value" value="fonts/cour.ttf"/>
    </node>
  </node>
  <node class="container" name="font4">
    <node class="string" name="font">
```

```

        <node-attribute name="value" value="Courier New"/>
    </node>
    <node class="boolean" name="bold">
        <node-attribute name="value" value="TRUE"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/courbd.ttf"/>
    </node>
</node>
<node class="container" name="font5">
    <node class="string" name="font">
        <node-attribute name="value" value="Segoe UI"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/segoeui.ttf"/>
    </node>
</node>
<node class="container" name="font6">
    <node class="string" name="font">
        <node-attribute name="value" value="Segoe UI"/>
    </node>
    <node class="boolean" name="bold">
        <node-attribute name="value" value="TRUE"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/segoeuid.ttf"/>
    </node>
</node>
<node class="container" name="font7">
    <node class="string" name="font">
        <node-attribute name="value" value="Microsoft Sans Serif"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/micross.ttf"/>
    </node>
</node>
<node class="container" name="font8">
    <node class="string" name="font">
        <node-attribute name="value" value="Fixedsys"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/fsex301-l2.ttf"/>
    </node>
</node>
<node class="container" name="font9">
    <node class="string" name="font">
        <node-attribute name="value" value="MS Sans Serif"/>
    </node>
    <node class="string" name="file">
        <node-attribute name="value" value="fonts/micross.ttf"/>
    </node>
</node>
</node>

```

I still need to find a replacement for the System font.

Note that for FixedSys I've used fsex301-l2.ttf provided by <http://www.fixedsysexcelsior.com/> , with a "fixed" version from comment 16 here: <http://askubuntu.com/questions/210283/how-to-use-fixedsys-in-the-gnome-terminal-or-whenever-monospaced-fonts-are-requ>

Another good read is this: <https://bugs.launchpad.net/ubuntu/+bug/200671>

LE: GES fixed the "Segue" typo.

#### #5 - 10/14/2015 03:18 PM - Constantin Asofiei

Only way until now to disable anti-aliasing on Firefox is by placing a ~/.fonts.conf file with this content:

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<!-- /etc/fonts.conf file to configure system font access -->
<fontconfig>

<!-- Antialias -->

<match target="font"><edit mode="assign" name="antialias">
<bool>>false</bool>
</edit>
</match>
```

The solution was found here: <http://ubuntuforums.org/showthread.php?t=1298180>

But this disables anti-aliasing for the entire browser... the only good news is with antialiasing off, the text is drawn correctly.

The more I dig, the less confident I am that there is a way to disable this at the CSS/canvas/canvas context level.

#### #6 - 10/14/2015 03:38 PM - Greg Shah

But this disables anti-aliasing for the entire browser... the only good news is with antialiasing off, the text is drawn correctly.

I appreciate the information. It is good to know that if we could disable anti-aliasing, that the implementation would work.

We have to assume this will not be an acceptable solution. Our default approach assumes our system will work without changing the user's browser configuration and without plugins or extensions.

The more I dig, the less confident I am that there is a way to disable this at the CSS/canvas/canvas context level.

Let me think on this some. I also want to run enough different testcases to get an idea of how much this will be a real problem.

**#7 - 10/15/2015 05:34 AM - Constantin Asofiei**

About bitmap fonts in general; what they did for wine is use FontForge to transform the bitmap (.fon) files to true-type (.ttf) files. Details are here: <http://wiki.winehq.org/CreateFonts#head-deff07f4a7aefdb379154cc4d0de0bd87fd440ec>

So is it acceptable (in licensing/copyright terms) to take the .fon files from Windows and converting them to .ttf?

Note that the .fon file for the System font in windows is vgasys.fon.

**#8 - 10/15/2015 09:29 AM - Greg Shah**

So is it acceptable (in licensing/copyright terms) to take the .fon files from Windows and converting them to .ttf?

No, it is most likely not allowed. Unless the user's license for these fonts allows them to create a derivative work, it would not be allowed. The chances of Microsoft having provided that right in their license for those fonts is approximately 0%.

Someone that is already licensed to those files **may** be able to use them with P2J (if we supported bitmap fonts). It would depend on the license they've got for that font, which may not be easy to determine since it may have been licensed as part of a larger system (e.g. Windows OS). It is possible that the license would not even allow this, but we can't know without looking at the specific license.

In what ways does the default 4GL GUI environment depend on this font?

For the current project, we know that we should not need to draw using the system font. For the current project, what impact does this have on sizing or other processing?

**#9 - 10/15/2015 09:31 AM - Constantin Asofiei**

Greg Shah wrote:

In what ways does the default 4GL GUI environment depend on this font?

4GL depends on bitmap fonts (System, FixedSys, MS Sans Serif) for layout/sizing and text drawing.

For the current project, we know that we should not need to draw using the system font. For the current project, what impact does this have on sizing or other processing?

If drawing is not performed using System font (and I think none of the bitmap fonts are used for drawing), then we have nothing else to do. The sizing/layout depends only on the System font's metrics, which we have already captured.

**#10 - 10/15/2015 10:10 AM - Constantin Asofiei**

Greg Shah wrote:

Someone that is already licensed to those files **may** be able to use them with P2J (if we supported bitmap fonts).

BTW, a question about the .ttf files in server/fonts - except the excelsior version, all are taken from the Windows fonts folder - so, again, what options do we have in using these fonts outside the Windows platform?

**#11 - 10/15/2015 10:46 AM - Greg Shah**

a question about the .ttf files in server/fonts - except the excelsior version, all are taken from the Windows fonts folder - so, again, what options do we have in using these fonts outside the Windows platform

Yes, that is most likely an issue. It is very likely that you would not be allowed to distribute those files to other systems. As Golden Code, we certainly can't provide them to anyone.

I think it is important to find replacement fonts that are freely licensed.

A starting place:

[https://en.wikipedia.org/wiki/Liberation\\_fonts](https://en.wikipedia.org/wiki/Liberation_fonts)

[https://en.wikipedia.org/wiki/Croscore\\_fonts](https://en.wikipedia.org/wiki/Croscore_fonts)

This is the original company that created most of those fonts, I think:

[https://en.wikipedia.org/wiki/Ascender\\_Corporation](https://en.wikipedia.org/wiki/Ascender_Corporation)

**#12 - 10/15/2015 12:22 PM - Greg Shah**

BTW, in the directory entries above, I think "Segue UI" should be "Segoe UI".

**#13 - 10/20/2015 09:53 AM - Greg Shah**

- *Status changed from New to Closed*

The font replacement work will be done in a separate task [#2765](#).

**#14 - 11/10/2015 04:55 AM - Greg Shah**

- % Done changed from 0 to 100

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

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

**Files**

---

ask_gui_at_initial_load_in_4gl_20150902.png	2.89 KB	09/10/2015	Greg Shah
ask_gui_at_initial_load_in_p2j_web_and_swing_20150902.png	9.41 KB	09/10/2015	Greg Shah
ask_gui_at_initial_load_in_p2j_web_20150910.png	7.49 KB	09/11/2015	Greg Shah
ask_gui_at_initial_load_in_p2j_swing_20150910.png	7.9 KB	09/11/2015	Greg Shah
ask_gui_at_initial_load_in_4gl_20150910.png	4.42 KB	09/11/2015	Greg Shah