

## User Interface - Bug #2995

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

### if the first mouse after app startup is on a combo-box, then an abend occurs

02/16/2016 11:01 AM - Greg Shah

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Igor Skornyakov	<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>			

## History

#1 - 02/16/2016 11:03 AM - Greg Shah

- File mouse.p added

This was reported by Igor:

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

Subject: Issue with COMBO-BOX

Date: Thu, 11 Feb 2016 07:25:32 -0500

From: Igor ...

To: Greg ...

Greg,

I've noticed the following issue. If I started the converted mouse.p (attached) and the first mouse action after startup is any COMBO-BOX activation the following exception happens:

```
Feb 11, 2016 7:19:29 AM Dispatcher.processInbound()
SEVERE: {main} Unexpected throwable.
java.lang.reflect.InvocationTargetException
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at com.goldencode.p2j.util.MethodInvoker.invoke(MethodInvoker.java:76)
at com.goldencode.p2j.net.Dispatcher.processInbound(Dispatcher.java:705)
at com.goldencode.p2j.net.Conversation.block(Conversation.java:319)
at com.goldencode.p2j.net.Conversation.waitForMessage(Conversation.java:257)
at com.goldencode.p2j.net.Queue.transactImpl(Queue.java:1128)
at com.goldencode.p2j.net.Queue.transact(Queue.java:585)
at com.goldencode.p2j.net.BaseSession.transact(BaseSession.java:223)
at com.goldencode.p2j.net.HighLevelObject.transact(HighLevelObject.java:163)
at com.goldencode.p2j.net.RemoteObject$RemoteAccess.invokeCore(RemoteObject.java:1425)
at com.goldencode.p2j.net.InvocationStub.invoke(InvocationStub.java:97)
at com.sun.proxy.$Proxy4.standardEntry(Unknown Source)
at com.goldencode.p2j.main.ClientCore.start(ClientCore.java:281)
at com.goldencode.p2j.main.ClientCore.start(ClientCore.java:102)
at com.goldencode.p2j.main.ClientDriver.start(ClientDriver.java:201)
at com.goldencode.p2j.main.CommonDriver.process(CommonDriver.java:398)
at com.goldencode.p2j.main.ClientDriver.process(ClientDriver.java:95)
at com.goldencode.p2j.main.ClientDriver.main(ClientDriver.java:267)
Caused by: java.lang.NullPointerException
at com.goldencode.p2j.ui.chui.ThinClient.waitForWorker(ThinClient.java:11105)
at com.goldencode.p2j.ui.client.ComboBox.activate(ComboBox.java:1113)
at com.goldencode.p2j.ui.client.gui.ComboBoxGuiImpl.access$700(ComboBoxGuiImpl.java:84)
at com.goldencode.p2j.ui.client.gui.ComboBoxGuiImpl$4.run(ComboBoxGuiImpl.java:440)
at com.goldencode.p2j.ui.chui.ThinClient.waitForEvent(ThinClient.java:13006)
at com.goldencode.p2j.ui.chui.ThinClient.waitForWorker(ThinClient.java:10963)
```

at com.goldencode.p2j.ui.chui.ThinClient.waitFor(ThinClient.java:10537)  
at com.goldencode.p2j.ui.chui.ThinClient.waitFor(ThinClient.java:10491)  
... 21 more

If I e.g. first click at any(?) other widget everything is OK. The preliminary analysis reveals that the registry fails to find DropDown.

This can be of course a result of my changes but it is not likely.

Should I continue investigation?

Thank you,  
Igor.

## #2 - 02/16/2016 11:04 AM - Greg Shah

From Igor:

After added synchronization for access to the WidgetRegistry.widgetList the problem seem to disappear. May be it makes sense to use a concurrent version of the LinkedHashMap ( <https://code.google.com/archive/redirect/a/code.google.com/p/concurrentlinkedhashmap?movedTo=https:%2F%2Fgithub.com%2Fben-manes%2Fconcurrentlinkedhashmap>)

## #3 - 02/16/2016 11:07 AM - Greg Shah

I prefer not to add a 3rd party code dependency for this. The 3rd party code also seems more designed for caching. The widget registry is not a cache, it is the authoritative store for mapping IDs and widgets. Using soft references and evicting elements is not desired behavior.

Please post the diff you used to resolve the issue. Have you included this change in 2565a?

## #4 - 02/16/2016 12:04 PM - Igor Skorniyakov

- File wr.diff added

Please find the diff file attached. The fix is committed to the task branch 2565a

## #5 - 02/17/2016 11:01 AM - Greg Shah

- File wr.txt added

Code Review Task Branch 2565a Revisions 11019/11020

Overall, the approach seems good.

1. Why are WidgetRegistry.getComponents() and WidgetRegistry.removeWidget() synchronized when all access to widgetList is already protected by the ReentrantReadWriteLock?
2. The import com.codahale.metrics.\*; should be removed.
3. Please remove the commented logging in WidgetRegistry.getComponent().
4. In the future, please note the revision that contains the change, so the reviewer doesn't have to search for it.

5. The best way to post diffs is as a text file. Capture the diffs like this:

```
bzr diff -r11018..11020 src/com/goldencode/p2j/ui/client/WidgetRegistry.java > wr.txt
```

Then attach the wr.txt to the entry (as I have done in this task history entry). By doing it this way, the viewer can see the diff in the browser. The way you did it, I had to download the diff AND it didn't even have the target filename (WidgetRegistry) in it, so the reader might now know where the diffs come from.

#### #6 - 02/17/2016 02:34 PM - Igor Skornyakov

Greg Shah wrote:

Code Review Task Branch 2565a Revisions 11019/11020

1. Why are WidgetRegistry.getComponents() and WidgetRegistry.removeWidget() synchronized when all access to widgetList is already protected by the ReentrantReadWriteLock?
2. The import com.codahale.metrics.\*; should be removed.
3. Please remove the commented logging in WidgetRegistry.getComponent().

Done.

Committed to the task branch 2565a revno 11024.

4. In the future, please note the revision that contains the change, so the reviewer doesn't have to search for it.

5. The best way to post diffs is as a text file. Capture the diffs like this:

```
bzr diff -r11018..11020 src/com/goldencode/p2j/ui/client/WidgetRegistry.java > wr.txt
```

Then attach the wr.txt to the entry (as I have done in this task history entry). By doing it this way, the viewer can see the diff in the browser. The way you did it, I had to download the diff AND it didn't even have the target filename (WidgetRegistry) in it, so the reader might now know where the diffs come from.

OK, thank you.

#### #7 - 03/01/2016 01:01 PM - Greg Shah

- % Done changed from 0 to 100
- Status changed from New to Closed

Resolved by trunk revision 10976 (merged from 2565a).

**#8 - 11/16/2016 12:12 PM - Greg Shah**

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

**Files**

---

mouse.p	6.79 KB	02/16/2016	Greg Shah
wr.diff	3.07 KB	02/16/2016	Igor Skornyakov
wr.txt	5.89 KB	02/17/2016	Greg Shah