

Bugs - Bug #3355

The tab pane button for HOTEL GUI is not included in the focus traversal chain

10/11/2017 03:42 PM - Sergey Ivanovskiy

Status:	Closed	Start date:	10/11/2017
Priority:	Normal	Due date:	
Assignee:	Sergey Ivanovskiy	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		case_num:	
billable:	No	version:	
vendor_id:	GCD		
Description			

History

#2 - 10/11/2017 04:48 PM - Sergey Ivanovskiy

I debugged this case and found that the nearest parent frame for the tab button have empty tab item list and it breaks the focus processing here in AbstractContainer

```
protected void focusWorker(boolean direction)
{
    Iterable<Widget<O>> iter = null;

    List<Widget<O>> widgets = getTabItemList();

    if (current != null)
    {
        Widget<O> anchor = current;
        if (anchor instanceof Frame)
        {
            anchor = ((Frame) anchor).getContentPane();
        }
        iter = (direction) ? Iterables.directFrom(widgets, anchor) :
            Iterables.reverseFrom(widgets, anchor);
    }
    else
    {
        iter = (direction) ? widgets : Iterables.reverse(widgets);
    }

    if (iter == null)
    {
        throw new IllegalStateException("Focused widget does not belong to container");
        .....
    }
}
```

and the next code is not executed

```
for (Widget<O> widget : iter)
{
    if (checkWidget(widget, direction))
        return;
}

setFocusInt(null);

// find the nearest focus-traversable parent
AbstractContainer<O> parent = (AbstractContainer<O>) parent();
while (parent != null && !parent.focusTraversable())
{
    parent = (AbstractContainer<O>) parent.parent();
}
```

```

    if (parent != null)
    {
        if (direction)
            parent.nextFocus();
        else
            parent.prevFocus();
        return;
    }
    else if (widgets.size() > 0)
    {
        iter = (direction) ? widgets : Iterables.reverse(widgets);

        for (Widget<O> widget : iter)
        {
            if (checkWidget(widget, direction))
                return;
        }
    }
}

```

#3 - 10/11/2017 05:00 PM - Sergey Ivanovskiy

This diff

```

=== modified file 'src/com/goldencode/p2j/ui/client/widget/AbstractContainer.java'
--- src/com/goldencode/p2j/ui/client/widget/AbstractContainer.java      2017-08-15 09:03:27 +0000
+++ src/com/goldencode/p2j/ui/client/widget/AbstractContainer.java      2017-10-11 20:52:08 +0000
@@ -1645,15 +1645,13 @@
     iter = (direction) ? widgets : Iterables.reverse(widgets);
 }

-     if (iter == null)
-     {
-         throw new IllegalStateException("Focused widget does not belong to container");
-     }
-
-     for (Widget<O> widget : iter)
-     {
-         if (checkWidget(widget, direction))
-             return;
+     if (iter != null)
+     {
+         for (Widget<O> widget : iter)
+         {
+             if (checkWidget(widget, direction))
+                 return;
+         }
+     }

    setFocusInt(null);

```

is a good candidate for a fix, because with this fix hotel gui tab management can traverse from the tab pane button to the next tab widget on the tab

pane, but the focus is still it doesn't come back to this button because of another issue that the frame containing the tab pane button has no tab items. May be it was created dynamically.

#4 - 10/11/2017 05:09 PM - Sergey Ivanovskiy

One note. The "Reset Filters" button on the "Reservation" tab panel isn't included in the focus traversal chain for this tab. It works properly according to 4GL system.

#5 - 10/14/2017 03:49 AM - Sergey Ivanovskiy

Added test-dyn-widgets.p in testcases/uast/next_tab_item that shows that tab items on the client side are not set for dynamic frames. It seems that it is also very needed to fix asap.

#6 - 10/14/2017 04:27 AM - Sergey Ivanovskiy

One of the case is that when a dynamic frame is created, its field group is null and any new created widget for which its frame attribute is assigned is lost because the field group is not created. Just it is very confusing.

```
on choose of b1 in frame f1 do:
```

```
create frame hFrame.  
h = frame f1:handle.  
h = h:first-child.  
hFrame:parent=h.  
hFrame:title="child frame #" + String(count + 1).  
hFrame:row=3 + 3 * count.  
hFrame:col=1.  
hFrame:width-chars=40.  
hFrame:height-chars=3.  
hFrame:visible=true.  
create button hButton.  
hButton:label = "ok #" + String(count + 1).  
hButton:frame = hFrame.  
hButton:sensitive=true.  
hButton:visible=true.  
count = count + 1.  
message "ok".  
end.
```

converted into

```
DynamicWidgetFactory.createFrame(hFrame);  
h.assign(f1Frame.asWidgetHandle());  
h.assign(h.unwrapTree().firstChild());  
hFrame.unwrapWidget().setParentHandle(h);  
hFrame.unwrapWidget().setTitle(concat("child frame #", valueOf(plus(count, 1))));  
hFrame.unwrapWidget().setRow(new decimal(plus(3, multiply(3, count))));  
hFrame.unwrapWidget().setColumn(new decimal(new integer(1)));  
hFrame.unwrapSizeable().setWidthChars(new decimal(new integer(40)));  
hFrame.unwrapSizeable().setHeightChars(new decimal(new integer(3)));  
hFrame.unwrapWidget().setVisible(new logical(true));  
DynamicWidgetFactory.createButton(hButton);  
hButton.unwrapCommonField().setLabel(concat("ok #", valueOf(plus(count, 1))));  
hButton.unwrapWidget().setFrameHandle(hFrame);  
hButton.unwrapSensitive().setSensitive(new logical(true));  
hButton.unwrapWidget().setVisible(new logical(true));  
count.assign(plus(count, 1));  
message("ok");
```

I supposed that `hButton.unwrapWidget().setFrameHandle(hFrame);` should add this button to the corresponding field group, but actually this field group is null.

#7 - 10/14/2017 04:37 AM - Sergey Ivanovskiy

According to the code in GenericFrame

```
/**
 * Get the first child of the frame which will be the field group that
 * contains all non-header widgets.
 *
 * @return The first field group.
 */
@Override
public handle firstChild()
{
    // we create this "pseudo-widget" the first time it is needed
    if (fieldGroup == null)
    {
        // TODO this field is initialized in #coreInitialize method. Is this situation legal?
        GenericWidget<?>[] widgetsArr =
            n2w.values().toArray(new GenericWidget[n2w.values().size()]);
        fieldGroup = wrapWidgetsToFieldGroup(widgetsArr, null);
        fieldGroup.setWidgetAdder(this::addWidget);
    }

    return new handle(fieldGroup);
}
```

the field group is created only on FIRST-CHILD invocation. It seems that field groups should be created with GenericFrame. Are there invisible issues if field groups are created with generic frames?

#8 - 10/14/2017 07:39 AM - Sergey Ivanovskiy

I found that FrameWidget have no access to its own field group but only to its parent field group. It seems that it is a design issue, correct?

#9 - 10/14/2017 12:19 PM - Greg Shah

Are there invisible issues if field groups are created with generic frames?

I don't know of any. It certainly seems like we should create them at the same time.

I found that FrameWidget have no access to its own field group but only to its parent field group. It seems that it is a design issue, correct?

Yes.

I think the issue is caused by the fact that we added field-group later (when we hit customer code that needed it) and so it was never really "designed" properly. We just did the minimum needed at the time.

Please fix it.

#10 - 10/14/2017 12:27 PM - Sergey Ivanovskiy

GenericFrame.coreInitialize(Class<?> configClass, Object configInstance, String frameName, boolean dynamic) with dynamic = true doesn't initialize this field group, please look at

```
// dynamic frame initialization should stop here
if (dynamic)
{
    initBackgroundGroup();
// doesn't initialize      firstChild(); ?
    return true;
}
```

Investigating this issue.

#11 - 10/14/2017 03:12 PM - Sergey Ivanovskiy

Please review committed revision 11178 (3287a). I checked <customer_name_redacted> poc, it seems the buttons are in focus traversal chain. Planning to test these changes with regression tests on devsrv01.

#12 - 10/15/2017 08:15 AM - Sergey Ivanovskiy

It seems that the main regression was passed in first try /results/20171014_174923/ there are 3 failed tests

```
1. 1. gso_tests: 1. gso_453: failure in step 14: 'timeout before the specific screen buffer became available (
Mismatched data at line 18, column 34. Expected ' ' (0x0020 at relative Y 18, relative X 34) and found ']' (0x
005D at relative Y 18, relative X 34), template: screens/navigation/syman_menu.txt.)'
2. gso_430: failure in step 9: 'timeout before the specific screen buffer became available (Mismatched data at
line 18, column 0. Expected 'r' (0x250C at relative Y 18, relative X 0) and found ' ' (0x0000 at relative Y 1
8, relative X 0), template: screens/gso/gso_430/early_clock_approve_focus_step1.txt.)'
2. tc_tests: tc_job_002: failure in step 41: 'Line size mismatch (line # = 11, base = 0, actual = 91).'gso_tes
ts: 1. gso_453: failure in step 14: 'timeout before the specific screen buffer became available (Mismatched da
ta at line 18, column 34. Expected ' ' (0x0020 at relative Y 18, relative X 34) and found ']' (0x005D at relat
ive Y 18, relative X 34), template: screens/navigation/syman_menu.txt.)'
2. gso_430: failure in step 9: 'timeout before the specific screen buffer became available (Mismatched data at
line 18, column 0. Expected 'r' (0x250C at relative Y 18, relative X 0) and found ' ' (0x0000 at relative Y 1
8, relative X 0), template: screens/gso/gso_430/early_clock_approve_focus_step1.txt.)'
2. tc_tests: tc_job_002: failure in step 41: 'Line size mismatch (line # = 11, base = 0, actual = 91).'
```

and in the second try these failed gso_tests tests were passed results/20171015_030142. Thus I think that there are no regressions.

#13 - 10/15/2017 08:51 AM - Greg Shah

Code Review 3287a Revision 11178

I'm OK with the changes. Some thoughts:

1. I prefer to have the initialization of the field group be unconditional (not just for the dynamic case) in `coreInitialize()`. Do you know a reason to leave it conditional? If not conditional, then the `firstChild()` code would not need to call `getOrCreateFieldGroup()` and in fact we wouldn't need `getOrCreateFieldGroup()` at all (see item 2 below).
2. I prefer for the initialization of the field group to be more directly done. Calling `firstChild()` just to get its side-effect is very unclear. Please move the field-group creation out of `getOrCreateFieldGroup()` and into a method called `createFieldGroup()`. Then call this directly from `coreInitialize()`. This can be made conditional (see item 1 above) in which case the `getOrCreateFieldGroup()` will have to remain but it would call `createFieldGroup()` OR if it is unconditional, then `getOrCreateFieldGroup()` can be removed.
3. In `FieldGroup.addNewTabItem()`, please add a comment explaining why the `LT.updateTabItemList()` is being called even when there are no state changes occurring.
4. `FrameWidget` has no actual changes. I think it can be reverted.
5. Please add history entries to all files.

#14 - 10/15/2017 08:52 AM - Greg Shah

Constantin/Hynek: Do you have any thoughts on [#3355-13](#) item 1?

#15 - 10/16/2017 06:30 AM - Hynek Cihlar

Greg Shah wrote:

Constantin/Hynek: Do you have any thoughts on [#3355-13](#) item 1?

`initBackgroundGroup()` and `firstChild()` are called for the non-dynamic case, too. Both calls in the non-dynamic case can probably be moved up and put into single location for both dynamic and non-dynamic cases.

I also agree that having to call `firstChild()` to initialize the field group may not be clear enough.

#16 - 10/16/2017 11:37 AM - Sergey Ivanovskiy

I investigated that P2J trunc now doesn't change sibling widgets order if MOVE-TO-TOP/MOVE-TO-BOTTOM is applied to the target widget, the

revision 11176 has unrelated changes but it seems that it works in some previous revisions. I tested the sibling list on the GUI 4GL and found that the widgets list returned by FIRST-CHILD, NEXT-SIBLING must change its order if MOVE-TO-TOP/MOVE-TO-BOTTOM is applied to the target widget. I used uast/next_tab_item/test-next-tab-item-list-2.p for this test. The same test proves that z-order widgets and tab items list are different at least for the static frame. This procedure from test-next-tab-item-list-2.p lists sibling for the given frame and this list must be changed if MOVE-TO-TOP/MOVE-TO-BOTTOM is applied to the target widget

```
/** lists all siblings of the first child in the provided container's field group */
PROCEDURE nextSibling:
  DEF INPUT PARAMETER VhContainer AS HANDLE.
  DEFINE VAR VhWidget AS HANDLE.
  DEFINE VAR i AS INT INITIAL 0.
  IF VALID-HANDLE(VhContainer) THEN
    DO:
      DO WHILE VhContainer:TYPE = "WINDOW":U:
        VhContainer = VhContainer:FIRST-CHILD.
      END.
      ASSIGN VhWidget = VhContainer:CURRENT-ITERATION.
      CREATE tt.
      tt.ii = i.
      tt.ty = VhWidget:TYPE.
      tt.nn = VhWidget:NAME.
      tt.xx = VhWidget:COLUMN.
      tt.yy = VhWidget:ROW.
      tt.ww = VhWidget:WIDTH.
      tt.hh = VhWidget:HEIGHT.
      tt.pp = 0.
      i = i + 1.

      VhWidget = VhWidget:FIRST-CHILD.

      DO WHILE VALID-HANDLE(VhWidget) :
        CREATE tt.
        tt.ii = i.
        tt.ty = VhWidget:TYPE.
        tt.nn = VhWidget:NAME.
        tt.xx = VhWidget:COLUMN.
        tt.yy = VhWidget:ROW.
        tt.ww = VhWidget:WIDTH.
        tt.hh = VhWidget:HEIGHT.
        tt.pp = VhWidget:TAB-POSITION.
        i = i + 1.

        MESSAGE "type: " + STRING(VhWidget:TYPE) + ", x: " + STRING(VhWidget:X) + " y: " + STRING(Vh
Widget:Y) +
          ", width-pixels: " + STRING(VhWidget:WIDTH-PIXELS) + ", height-pixels: " + STRING(V
hWidget:HEIGHT-PIXELS).
        ASSIGN VhWidget = VhWidget:NEXT-SIBLING.
      END.
    END.
  END PROCEDURE.
```

It seems that the new bug must be created.

#17 - 10/16/2017 11:52 AM - Greg Shah

I would prefer if this problem is fixed as part of this task. It seems pretty important to fix.

#18 - 10/16/2017 07:41 PM - Sergey Ivanovskiy

Greg Shah wrote:

Code Review 3287a Revision 11178

Fixed, committed revision 11179.

#19 - 10/17/2017 09:39 AM - Sergey Ivanovskiy

Greg Shah wrote:

I would prefer if this problem is fixed as part of this task. It seems pretty important to fix.

It seems that this issue is related to protected boolean `useHandleChainSiblings()` for `FieldGroup` it returns true, then the server side logic uses `HandleChain`, but its own managed widgets are untouched. Please look at `BaseEntity.getNextSibling()/getPrevSibling`. `GenericWidget.moveToTop/moveToBottom` delegates this functionality to `HandleChain.moveInChain` after the client places widget on the top/bottom. But for `FillInWidget` protected boolean `useHandleChainSiblings()` return false and it uses `FieldGroup` internal widgets when it iterates over next-sibling chain. It seems we have a gap here.

#20 - 10/17/2017 10:10 AM - Sergey Ivanovskiy

It looks like we should set `useHandleChainSiblings() === true` for all regular widgets because `FieldGroup` has this too now. Then the next-sibling search will be consistent. Correct?

#21 - 10/17/2017 01:42 PM - Constantin Asofiei

Sergey Ivanovskiy wrote:

It looks like we should set `useHandleChainSiblings() === true` for all regular widgets because `FieldGroup` has this too now. Then the next-sibling search will be consistent. Correct?

`HandleChain` was being used for sibling support ONLY in the same widget type (i.e. `FRAME`, `FILL-IN`, etc) - I made some additions so that it looks for siblings with the same parent, but as you found it doesn't cover all cases.

The solution might be correct for `useHandleChainSiblings() === true`, but ask yourself this: in FWD server-side, are the `FILL-IN`, etc widgets parented to the `FRAME` or to the `FIELD-GROUP`? If this parent is correct, then it might work.

Another solution would be to ensure that `FieldGroup` widget order is in sync with `move-to-top/move-to-bottom`.

#22 - 10/18/2017 12:12 AM - Sergey Ivanovskiy

rev. 11179 passed ctrl-c-regression tests by two test runs. The main-regression had several failed tests gso_190 and tc_dc_slot_024, tc_dc_slot_029, tc_job_002, tc_job_clock_004, tc_pay_wr_inq_001, tc_job_matlcron_001, tc_job_matlcron_003. tc_pay_wr_inq_001 looks suspected but all they have timeouts reasons. Planning to run main-regression again.

#23 - 10/18/2017 06:36 AM - Sergey Ivanovskiy

This main-regression test was failed without results due to this reason

```
[exec] ESC M : R is 7, tm is 7, bm is 20
[exec] Non-testing errors (CONNECT_FAILURE, complete == true) occurred during test execution:
[exec] com.jcraft.jsch.JSchException: channel is not opened.
[exec]     at com.jcraft.jsch.Channel.sendChannelOpen(Channel.java:765)
[exec]     at com.jcraft.jsch.Channel.connect(Channel.java:151)
[exec]     at com.goldencode.harness.transport.SSH2Transport.connect(SSH2Transport.java:79)
[exec]     at com.goldencode.harness.transport.TransportManager.connect(TransportManager.java:69)
[exec]     at com.goldencode.harness.Driver.<init>(Driver.java:83)
[exec]     at com.goldencode.harness.TestSet.run(TestSet.java:257)
[exec]     at java.lang.Thread.run(Thread.java:745)
[exec] ** MAIN part has failed!
```

check-logs:

```
[exec] No exceptions found in server and client logs.
```

BUILD FAILED

```
/opt/secure/clients/timco/testing/build_rt.xml:528: exec returned: 1
```

Total time: 237 minutes 43 seconds

#24 - 10/18/2017 06:36 AM - Sergey Ivanovskiy

Starting main-regression again on devsrv01.

#25 - 10/18/2017 11:24 AM - Sergey Ivanovskiy

What happens with devsrv01, this test main-regression failed with this exception again?

```
[exec] Non-testing errors (CONNECT_FAILURE, complete == true) occurred during test execution:
[exec] com.jcraft.jsch.JSchException: channel is not opened.
[exec]     at com.jcraft.jsch.Channel.sendChannelOpen(Channel.java:765)
[exec]     at com.jcraft.jsch.Channel.connect(Channel.java:151)
[exec]     at com.goldencode.harness.transport.SSH2Transport.connect(SSH2Transport.java:79)
```

```
[exec]          at com.goldencode.harness.transport.TransportManager.connect
(TransportManager.java:69)
[exec]          at com.goldencode.harness.Driver.<init>(Driver.java:83)
[exec]          at com.goldencode.harness.TestSet.run (TestSet.java:257)
[exec]          at java.lang.Thread.run (Thread.java:745)
[exec] ** MAIN part has failed!
```

check-logs:

```
[exec] ===== Exceptions in file: server_gso_0.log =====:
```

#26 - 11/01/2017 01:39 PM - Greg Shah

3287a was merged into the trunk as rev 11183.

Can I close this task?

#27 - 11/01/2017 01:46 PM - Sergey Ivanovskiy

Yes, this particular issue was fixed. But these [#3355-19](#), [#3355-20](#) and [#3355-21](#) were not fixed because the solution was not ready.

#28 - 11/20/2017 03:05 AM - Sergey Ivanovskiy

Created task branch 3355a to work on [#3355-19](#), [#3355-20](#) and [#3355-21](#).

#29 - 11/20/2017 04:04 PM - Sergey Ivanovskiy

Constantin Asofiei wrote:

Sergey Ivanovskiy wrote:

It looks like we should set `useHandleChainSiblings() === true` for all regular widgets because `FieldGroup` has this too now. Then the next-sibling search will be consistent. Correct?

`HandleChain` was being used for sibling support ONLY in the same widget type (i.e. `FRAME`, `FILL-IN`, etc) - I made some additions so that it looks for siblings with the same parent, but as you found it doesn't cover all cases.

The solution might be correct for `useHandleChainSiblings() === true`, but ask yourself this: in FWD server-side, are the `FILL-IN`, etc widgets parented to the `FRAME` or to the `FIELD-GROUP`? If this parent is correct, then it might work.

Another solution would be to ensure that `FieldGroup` widget order is in sync with `move-to-top/move-to-bottom`.

Constantin, I decided to fix this issue by synchronizing field group widgets with their current z-order list. Please review 11202(3355a). Planning to run regression tests after 3240c.

#30 - 11/21/2017 10:09 AM - Greg Shah

Code Review Task Branch 3355a Revision 11202

I am fine with the changes.

Constantin: any objections?

#31 - 11/21/2017 11:12 AM - Constantin Asofiei

Greg Shah wrote:

Constantin: any objections?

The changes make sense.

#32 - 11/21/2017 11:19 AM - Greg Shah

Constantin: How about merging these into 3369a?

#33 - 11/21/2017 11:42 AM - Constantin Asofiei

Greg Shah wrote:

Constantin: How about merging these into 3369a?

Yes. Sergey, please merge into 3369a.

#34 - 11/21/2017 01:43 PM - Sergey Ivanovskiy

OK, my patch was failed, I need to apply changes manually.

#35 - 11/27/2017 03:45 PM - Greg Shah

- Assignee set to *Sergey Ivanovskiy*
- Status changed from *New* to *Closed*
- % Done changed from *0* to *100*

Branch 3369a was merged to trunk as revision 11206.