User Interface - Bug #2988

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

The web click events on the caption buttons can invoke the resize action.

02/09/2016 12:38 AM - Sergey Ivanovskiy

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Sergey Ivanovskiy		**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:	Cleanup and Stabilization for GUI			
billable:	No		**case_num:**	
vendor_id:	GCD		**version:**	

Description

Related issues:

Related to User Interface - Bug #3075: The window icon popup menu isn't displ...   **Closed**

History

#1 - 02/09/2016 12:42 AM - Sergey Ivanovskiy

Clicking on the minimized caption button can invoke the resize action for the window under the caption button if the minimized window is on the top of the other. It looks like the corresponding mouse events are bubbled and caught by the next listener. In the considered case it is a resize listener.

#2 - 02/09/2016 12:44 AM - Sergey Ivanovskiy

- Assignee set to Sergey Ivanovskiy

#3 - 04/18/2016 10:29 AM - Sergey Ivanovskiy

The current version 1811u has the following issue if we minimize the top most window of equal-sized stacked windows, then all windows are minimized. It can be reproduced for the ./demo/simple_windows.p

#4 - 04/18/2016 01:11 PM - Constantin Asofiei

If you do this by pressing the minimize caption button, then this might be a problem of "leaking" the event to windows underneath the top one. Only one window should treat the event (the topmost one, under the location of the mouse).

#5 - 04/18/2016 01:16 PM - Sergey Ivanovskiy

Yes, agree. WindowStateChangedWorker registers "dblclick", "click", "mousedown", "mouseup" listeners. If windows are stacked, then the top most window gets "mousedown" and invoke the "minimize" action, but then "mouseup" listener of the second window handles "mouseup" event and executes the "minimize" action.

#6 - 04/18/2016 01:30 PM - Sergey Ivanovskiy

In this case the top window is collapsed, but the second window becomes the top window and gets "mouseup" event. I investigated this case with Firefox by logging and it couldn't be watched by setting the break point.

#7 - 04/18/2016 01:31 PM - Constantin Asofiei

Hm... I think we need to find a way so that all the mouse events (down, up, click, dblclick) work with the same initial source, and don't re-compute the source for each event.

#8 - 04/19/2016 12:01 AM - Sergey Ivanovskiy

Planning to dispatch mouse events via a global document listener, MouseEventDispatcher from p2j.screen.js.

#9 - 04/19/2016 01:49 AM - Sergey Ivanovskiy

According to the tests performed for Firefox and Chromium these browsers support the determined sequence of the following mouse events: mousedown, mouseup and click. If we press the left mouse button within the browsers window and release it out the browser, then the global document listener for these events catches events in a sequence: mousedown, mouseup and click. Thus it can be used to determine the target window that continues to be valid during this events circle. This target window will be used to fix this bug's issue.

The other solution to use the global dispatcher differs from the existing design. MouseEventDispatcher can dispatch mousedown, mouseup and click events to the registered OS events handlers: WindowIconify, WindowMaximize and WindowClose, but this design requires to move the others OS events handlers: MouseMovable, MousePopupable and MouseEditable to use the global dispatcher.

#10 - 04/19/2016 08:46 AM - Sergey Ivanovskiy

The current progress state is that WindowIconify, WindowMaximize, WindowClose, MouseMovable, MousePopupable (except MouseEditable that is invoked from MouseHandler) and MouseResizable are forced to use the global dispatcher. MouseMovable and MouseResizable are in progress.

#11 - 04/19/2016 04:11 PM - Sergey Ivanovskiy

- File 2988_1.txt added

Please review the committed revision 11012. These WindowIconify, WindowMaximize, WindowClose, MouseMovable, MousePopupable OS mouse widgets get required mouse events from the global mouse events dispatcher. The other listeners MouseHandler (uses canvas mouse listeners) and MouseEditable and MouseResizable (uses global document mouse listener) are not changed.

#12 - 04/19/2016 04:28 PM - Sergey Ivanovskiy

It is possible to delegate this code

```
// register default mouse listeners last
for(var mouseOp in mouseOps)
{
    if (mouseOps.hasOwnProperty(mouseOp))
    {
        // cannot use an array here because some indexes are negative numbers
        this.mouseAwareWidgets[mouseOp] = {};
        if (mouseOp !== "drag" && mouseOp !== "click" && mouseOp !== "mouseup" && mouseOp !== "mousedown")
        {
            this.canvas.addEventListener(mouseOp, this.raiseMouseEvent(this, mouseOp));
        }
    }
}
```

to MouseEventDispatcher.

#13 - 04/20/2016 04:23 AM - Sergey Ivanovskiy

Vadim, could you help what mouse events are listen by popups? I am doing changes for the web client to dispatch the target mouse events using the global dispatcher?

For an example, the popup menu is opened on clicking the right mouse button and if the mouse is over this popup, then its menu item is highlighted. And if the mouse is down out of this popup, then it is closed. Please give me a direction what java ui classes are responsible for popups and their mouse events handling?

#14 - 04/20/2016 05:11 AM - Sergey Ivanovskiy

Nevermind, I found MousePopupable and the other related classes.

#15 - 04/20/2016 09:47 AM - Constantin Asofiei

Sergey, the changes in 1811u rev 11012 make sense. The only question is this: you are assuming the Window's name (i.e. the canvas with the P2J Window, correct?) will contain the ID and will use the pwindow_<id> format. But what about this code:

```
options.id      = options.id      || ("pwindow_" + wid);
```

Wouldn't this allow the id to be specified via some external configuration, which might not follow this pattern?

#16 - 04/20/2016 10:00 AM - Sergey Ivanovskiy

Constantin Asofiei wrote:

> Sergey, the changes in 1811u rev 11012 make sense. The only question is this: you are assuming the Window's name (i.e. the canvas with the P2J Window, correct?) will contain the ID and will use the pwindow_<id> format. But what about this code:
> [...]
>
> Wouldn't this allow the id to be specified via some external configuration, which might not follow this pattern?

Since all windows created by our API satisfy this pattern "pwindow_", we can fix this by options.id = "pwindow_" + wid;. Now I am doing the proposal in note 12. There are difficulties. One is the OS widgets are processed differently from the mouse widgets and it can be fixed by adding the OS widgets to mouse widgets list, because all of them now get mouse events from MouseEventsDispatcher. Planning to fix them today and to prepare the code for the review.

#17 - 04/20/2016 11:39 AM - Sergey Ivanovskiy

We can use a map of custom html element's ids to their application window's ids in order to fix the usage of "pwindow_" + wid.

#18 - 04/20/2016 03:21 PM - Sergey Ivanovskiy

- File 2988_2.txt added

Committed revision 11013 rolled back my changes to 11011. It needs to find another solution. The previous commit to use the general dispatcher isn't fitted because it has bad impacts on the web client performance, it needs to find the target source widget from the objects holder each time a new event is delivered. For example, windows are moved slower than they are executed before and there are difficulties to deliver all required messages correctly. I added the last dirty diff for you to evaluate this way.

#19 - 04/20/2016 05:20 PM - Sergey Ivanovskiy

- File control_timer_fix.txt added

Committed revision 11014 fixed an infinite process to connect to the down java web client.

#20 - 04/20/2016 05:49 PM - Sergey Ivanovskiy

- File WindowIconLeftMousePopup.png added

The current P2j trunc Swing client doesn't open a window icon popup menu that should be opened on the left mouse click. The Web client opens this popup menu, but it doesn't work. It means that if we click on the menu item, then its action isn't executed. The same holds for the 1811u web client.

#21 - 04/21/2016 12:26 AM - Sergey Ivanovskiy

Committed revision 11015 has small fixes for misspelled listen event of MousePopupable and incorrectly built event's object of MouseMovable.

#22 - 04/21/2016 01:52 AM - Sergey Ivanovskiy

Planning to rebase to the current P2j trunc.

#23 - 04/21/2016 02:30 AM - Sergey Ivanovskiy

Now 1811u is from 11006 rev of trunc and is pushed up to revision 11021.

#24 - 04/25/2016 11:25 AM - Sergey Ivanovskiy

Committed revision 11037 minimizes a target window on mouseup. The bug scenario described in note-1 can't be reproduced on the current version.

#25 - 04/25/2016 02:26 PM - Greg Shah

Can I close this?

#26 - 04/25/2016 02:28 PM - Sergey Ivanovskiy

Yes, because the note-1 isn't reproduced.

#27 - 04/25/2016 02:29 PM - Greg Shah

- Status changed from New to Closed

- Target version set to Milestone 16

- Start date deleted (02/09/2016)

- % Done changed from 0 to 100

#28 - 11/16/2016 12:22 PM - Greg Shah

- Target version changed from Milestone 16 to Cleanup and Stabilization for GUI

Files

2988_1.txt	17.7 KB	04/19/2016	Sergey Ivanovskiy

2988_2.txt	22.1 KB	04/20/2016	Sergey Ivanovskiy
control_timer_fix.txt	1.65 KB	04/20/2016	Sergey Ivanovskiy
WindowIconLeftMousePopup.png	11.4 KB	04/20/2016	Sergey Ivanovskiy