

## User Interface - Feature #2560

Feature # 2252 (Closed): implement GUI client support

### implement GUI DIALOG-BOX support

04/29/2015 04:57 PM - Greg Shah

<b>Status:</b>	Closed	<b>Start date:</b>	04/29/2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Hynek Cihlar	<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>vendor_id:</b>	GCD
<b>billable:</b>	No		
<b>Description</b>			
<b>Related issues:</b>			
Related to User Interface - Bug #2729: alert-box usage is broken			<b>Closed</b>

### History

#### #1 - 09/17/2015 06:21 PM - Hynek Cihlar

- Status changed from New to WIP

#### #2 - 09/19/2015 07:30 AM - Hynek Cihlar

I came up with a design that limits the amount of changes to current ChUI dialog implementation and GUI widget structure with respect to the relatively short time frame.

The idea is to create a new class `FrameWindowGuiImpl` that will host a `FrameGuiImpl` instance, will manage its lifetime, route necessary events, etc. The changes to the current structures and logic should be minimal, most importantly the places where `FrameConfig.dialog` is handled will have to be modified to account for the new GUI use cases.

#### #3 - 09/20/2015 07:15 AM - Greg Shah

The approach seems reasonable.

#### #4 - 09/20/2015 06:27 PM - Hynek Cihlar

Created task branch 2560a from trunk revision 10937.

#### #5 - 09/28/2015 06:50 AM - Hynek Cihlar

Rebased task branch 2560a from P2J trunk revision 10940. The current revision in 2560a is now 10945.

#### #6 - 10/07/2015 10:57 AM - Greg Shah

Please provide a detailed list of the remaining work/testing that needs to be done before you believe this is ready for code review. I'd also like an estimate of the time needed for those items.

#### #7 - 10/07/2015 11:08 AM - Hynek Cihlar

Left known issues and todo tasks are:

- Fix multiple show/hide of the same frame/dialog-box. 0.25 MD
- Fix default dialog-box position. 0.25 MD
- More tests and fixing potential differences of event handling in the dialog-box. I've found already some some minor differences. So far I can say this will take 1.5 MD.
- Majic regression testing. This is already in progress with passing Ctrl+C. 0.5 MD

**#8 - 10/07/2015 11:10 AM - Hynek Cihlar**

Hynek Cihlar wrote:

Left known issues and todo tasks are:

- Fix multiple show/hide of the same frame/dialog-box. 0.25 MD
- Fix default dialog-box position. 0.25 MD
- More tests and fixing potential differences of event handling in the dialog-box. I've found already some some minor differences. So far I can say this will take 1.5 MD.
- Majic regression testing. This is already in progress with passing Ctrl+C. 0.5 MD

I forgot that [#1811](#) has some conflicting changes, this will need to be merged and retested. I expect 1 MD for this.

**#9 - 10/07/2015 01:36 PM - Greg Shah**

I forgot that [#1811](#) has some conflicting changes, this will need to be merged and retested. I expect 1 MD for this.

Are you wanting 1811r to merge to trunk before 2560a?

**#10 - 10/07/2015 01:39 PM - Hynek Cihlar**

Greg Shah wrote:

I forgot that [#1811](#) has some conflicting changes, this will need to be merged and retested. I expect 1 MD for this.

Are you wanting 1811r to merge to trunk before 2560a?

Yes, let's merge 1811r first. It is a bigger chunk of changes than 2560a.

**#11 - 10/11/2015 03:32 PM - Hynek Cihlar**

Rebased task branch 2560a from P2J trunk revision 10945. The current revision in 2560a is now 10967.

**#12 - 10/15/2015 06:45 PM - Hynek Cihlar**

Rebased task branch 2560a from P2J trunk revision 10947. The current revision in 2560a is now 10974.

**#13 - 10/16/2015 11:00 AM - Greg Shah**

Code Review Task Branch 2560a Revision 10976

Generally, I don't have many issues with this update.

1. Our window support is now more complicated than in the past. I don't think it is clear to readers of the code how and why the hierarchy is designed the way it is. There is frequent usage of `instanceof` to differentiate between various levels in the hierarchy, where normally there would be overridden methods that just implement differently. The purpose behind `TopLevelWindow`, `TitledWindow`, `Window` and so forth is not easily understandable. I don't think it is written down anywhere and I'm getting confused myself. I think it also leads to a very fragile implementation which is why it is constantly regressing and breaking as we extend the system. Since we are "under the gun" to get things done, we need to schedule some time (probably after the next 2 weeks) for you to write up some documentation (hopefully in javadoc) to summarize the core ideas behind the approach. Hopefully that will be reduced after 2560a is merged to trunk, but I think the documentation will be quite important to help us deal with the fragility in the time before we can find a better/less fragile implementation.

2. The `FocusManager.getFirstEnabledWidget()` TODO about dialog windows seems pretty important. What is the implementation plan?

3. What is the reason for adding `import org.apache.commons.lang.*;` to `WindowManager`, `FrameGuiImpl`?

4. `FrameGuiImpl.location()` and `FrameGuiImpl.physicalLocation()` are missing javadoc.

5. `ModalWindow.isRealized()` and `ModalWindow.realize()` are missing javadoc.

6. `GuiWindowCommons` needs class-level javadoc.

**#14 - 10/16/2015 11:23 AM - Hynek Cihlar**

Greg Shah wrote:

Code Review Task Branch 2560a Revision 10976

Generally, I don't have many issues with this update.

1. Our window support is now more complicated than in the past. I don't think it is clear to readers of the code how and why the hierarchy is designed the way it is. There is frequent usage of `instanceof` to differentiate between various levels in the hierarchy, where normally there would be overridden methods that just implement differently.

Yes, agree, but please note that my changes effectively add 4 `instanceof`'s and remove 2. The others are changes of methods moved to more appropriate places in the window class hierarchy and `WindowManager`. I will document those new `instanceof`'s to clarify, why they were needed.

The purpose behind `TopLevelWindow`, `TitledWindow`, `Window` and so forth is not easily understandable. I don't think it is written down anywhere and I'm getting confused myself. I think it also leads to a very fragile implementation which is why it is constantly regressing and breaking as we

extend the system.

I agree.

2. The `FocusManager.getFirstEnabledWidget()` TODO about dialog windows seems pretty important. What is the implementation plan?

Good point. I need to find the right use case and then decide how to deal with the TODO. I wouldn't want to delay the intergration of 2560a because of this though.

3. What is the reason for adding `import org.apache.commons.lang.*;` to `WindowManager`, `FrameGuiImpl`?

In `WindowManager` the reason was the reference of `ArrayUtils.toPrimitive` but I think that can be replaced with the inhouse `Utils` class.

In `FrameGuiImpl` the reason is the use of `ObjectUtils.equals`. I will replace this with `Objects.equals`.

**#15 - 10/16/2015 11:27 AM - Greg Shah**

I need to find the right use case and then decide how to deal with the TODO. I wouldn't want to delay the intergration of 2560a because of this though.

This is reasonable. You can put that into 1811s which is where we are collecting many GUI fixes.

**#16 - 10/16/2015 08:17 PM - Hynek Cihlar**

Checked-in 2560a revision 10980:

- addressed the points from review,
- fixed all failed chui tests from the last Majic regression run,
- other unrelated GUI fixes and improvements.

Note that there are some missing file header entries. I will address this tomorrow. Regression test in progress.

**#17 - 10/17/2015 06:37 AM - Hynek Cihlar**

One more chui regression from the last Majic test run, being fixed now.

**#18 - 10/17/2015 09:16 AM - Greg Shah**

Code Review Task Branch 2560a Revision 10980

I'm good with the changes.

**#19 - 10/18/2015 09:51 AM - Greg Shah**

Code Review Task Branch 2560a Revision 10984

I'm fine with the changes. I assume you'll javadoc and header them before you are done.

In regard to the tracing utility, we have wanted to add tracing for a while. This version is a bit too specific to UI, but it is fine for now. I think we will extend/improve it later. One thought: make Trace only contain very generic stuff (everything in it now except for window, x1, x2, y1, y2) and allow the user to sub-class it to add more specific data. Also Trace should be called something like TracePoint or TraceEvent to make it clear that it represents a captured data point.

**#20 - 10/18/2015 10:53 AM - Hynek Cihlar**

- File *Trace.png* added

Greg Shah wrote:

Code Review Task Branch 2560a Revision 10984

I'm fine with the changes. I assume you'll javadoc and header them before you are done.

Yes.

In regard to the tracing utility, we have wanted to add tracing for a while. This version is a bit too specific to UI, but it is fine for now. I think we will extend/improve it later. One thought: make Trace only contain very generic stuff (everything in it now except for window, x1, x2, y1, y2) and allow the user to sub-class it to add more specific data. Also Trace should be called something like TracePoint or TraceEvent to make it clear that it represents a captured data point.

I crafted this to solve a particular regression. I needed to query tracing entries linked to particular screen coordinates. The intended usage was to keep the entries in memory and then query the them with Memory Analyzer and its OQL capabilities.

Example:

```
SELECT tick, toString(msg), x1, y1, toString(clazz), toString(method), toString(anything), line FROM com.goldencode.util.Trace WHERE ((x1 <= 6) and (y1 <= 11) and (x2 >= 6) and (y2 >= 11))
```

Returns all trace entries related to the screen point 6x11, see the attached screenshot.

I intentionally chose the identifiers short to limit typing when writing the OQL expressions. But I agree TraceEvent is more descriptive than Trace.

I didn't originally plan to keep it but if you think it may be useful I will put it to a more appropriate package, maybe com.goldencode.p2j.

**#21 - 10/18/2015 01:48 PM - Hynek Cihlar**

2560a revision 10986 fixes the found ChUI regression. The problem was caused by a latent bug in browse's focus processing. Majic regression testing restarted.

**#22 - 10/18/2015 09:01 PM - Greg Shah**

I didn't originally plan to keep it but if you think it may be useful I will put it to a more appropriate package, maybe `com.goldencode.p2j`.

Please use `com.goldencode.trace` instead.

**#23 - 10/19/2015 04:22 AM - Hynek Cihlar**

The regression results of 2560a revision 10986 show two suspicious failed tests, `gso_245` and `tc_codes_crew_003`. When tested manually they pass. I am rerunning the main part to confirm these are negative failures.

**#24 - 10/19/2015 07:42 AM - Greg Shah**

Please post the final, clean code to bzt.

**#25 - 10/19/2015 09:30 AM - Hynek Cihlar**

Greg Shah wrote:

Please post the final, clean code to bzt.

Checked in, task branch 2560a revision 10987.

**#26 - 10/19/2015 09:35 AM - Greg Shah**

Code Review Task Branch 2560a Revision 10987

I'm good with the changes.

What is the status of testing?

**#27 - 10/19/2015 10:09 AM - Hynek Cihlar**

Greg Shah wrote:

Code Review Task Branch 2560a Revision 10987

I'm good with the changes.

What is the status of testing?

gso\_245 and tc\_codes\_crew\_003 passed, I think we can consider the branch stable enough.

**#28 - 10/19/2015 10:11 AM - Greg Shah**

Have you re-run any of the GUI testcases to see if there are regressions?

What is the status of [#2702](#), [#2752](#), [#2729](#), [#2564](#) note 90 and the move window test?

**#29 - 10/19/2015 11:04 AM - Hynek Cihlar**

Greg Shah wrote:

Have you re-run any of the GUI testcases to see if there are regressions?

I have run the dialog-box specific testcases, some browse, combo box and selection test cases. I didn't run all the GUI test cases though.

What is the status of [#2702](#), [#2752](#), [#2729](#), [#2564](#) note 90 and the move window test?

There are frame layout issues which affect tiny input as well as dialog-box. There have been changes in #1811/1811s which may resolve these. When 1811s is released these will have to be re-checked.

[#2702](#): Tiny input tested and works with the limitation above.

I didn't check [#2752](#)/#2704, there didn't seem to be any connection with this redmine issue.

[#2729](#): alert-box retested and works. I now retested editorWidget.p and the alert-box part works ok.

[#2564](#): I won't be able to retest this before the related p2j changes are merged to trunk. I wasn't able to convert the test files.

Also please note that I will need to do more testing on the production source code.

**#30 - 10/19/2015 11:13 AM - Greg Shah**

As far as you know, does this update cause any new regressions to GUI?

**#31 - 10/19/2015 11:14 AM - Hynek Cihlar**

Greg Shah wrote:

As far as you know, does this update cause any new regressions to GUI?

I don't know of any new regressions.

**#32 - 10/19/2015 11:17 AM - Greg Shah**

OK, please merge to trunk now.

**#33 - 10/19/2015 11:23 AM - Greg Shah**

In your commit message, make sure to also reference [#2702](#) and [#2729](#) (e.g. using Refs #2560, #2702, #2729.).

**#34 - 10/19/2015 11:40 AM - Hynek Cihlar**

#2560a checked in to trunk as revision 10948.

**#35 - 10/19/2015 11:42 AM - Greg Shah**

There are frame layout issues which affect tiny input as well as dialog-box. There have been changes in #1811/1811s which may resolve these. When 1811s is released these will have to be re-checked.

Other than this, are there any other open items for this task?

**#36 - 10/19/2015 11:51 AM - Hynek Cihlar**

Greg Shah wrote:

There are frame layout issues which affect tiny input as well as dialog-box. There have been changes in #1811/1811s which may resolve these. When 1811s is released these will have to be re-checked.

Other than this, are there any other open items for this task?

Yes, the production code retest.



**#37 - 10/19/2015 11:54 AM - Hynek Cihlar**

Hynek Cihlar wrote:

Greg Shah wrote:

There are frame layout issues which affect tiny input as well as dialog-box. There have been changes in #1811/1811s which may resolve these. When 1811s is released these will have to be re-checked.

Other than this, are there any other open items for this task?

Yes, the production code retest.

Plus the TODO: dialog-box left in the source code.

**#38 - 10/19/2015 11:55 AM - Hynek Cihlar**

Hynek Cihlar wrote:

Hynek Cihlar wrote:

Greg Shah wrote:

There are frame layout issues which affect tiny input as well as dialog-box. There have been changes in #1811/1811s which may resolve these. When 1811s is released these will have to be re-checked.

Other than this, are there any other open items for this task?

Yes, the production code retest.

Plus the TODO: dialog-box left in the source code.

**#39 - 10/19/2015 11:59 AM - Greg Shah**

Any additions or changes will be done in 1811s (or if they are ready after we need to go into testing for 1811s, then in some other future branch).

**#40 - 10/19/2015 12:01 PM - Hynek Cihlar**

Ok, I will check the known issues after 1811s is rebased.

**#41 - 01/23/2016 06:29 AM - Greg Shah**

- % Done changed from 0 to 100

- Status changed from WIP to Closed

Any remaining issues are being resolved in other tasks.

**#42 - 11/16/2016 12:12 PM - Greg Shah**

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

**Files**

---

Trace.png	235 KB	10/18/2015	Hynek Cihlar
-----------	--------	------------	--------------