User Interface - Feature #3469

final improvements and polish for the embedded mode web client

02/07/2018 11:26 AM - Greg Shah

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:	Constantin Asofiei		**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:				
billable:	No		**version:**	
vendor_id:	GCD			

Description

Related issues:

Related to User Interface - Feature #3258: improve javascript upcalls from th...	**New**
Related to User Interface - Bug #3582: Add missing support for dynamic menu s...	**Closed**
Related to User Interface - Feature #3700: in embedded mode, show a loading o...	**New**

History

#1 - 02/07/2018 11:29 AM - Greg Shah

Based on a large customer GUI application, we can see that the following improvements are needed for embedded mode:

1. module navigation - improve this to structure it in a hierarchical manner which mimics the navigation of the current application, also make it visually more attractive/more seamlessly integrating it into the UI of the embedded client
2. toolbar - most large GUI applications are very dependent upon the toolbar, can we use automation to read this and create a JS version that is attractive, modern and is hooked into the application event processing
3. logout prompt/integration - pressing the logout button in the embedded mode should run the application's logout processing (it would be OK to configure the pathname for the logout program)
4. menu bar - better visual integration with the rest of the embedded UI, optional flag to allow disabling the menus
5. footer control - provide a mechanism for the application to provide static content that can be placed in the footer links, copyright notice...)

Without 1-3, I don't think the large GUI application can use embedded mode. 4 and 5 are "polish" items that make the solution more complete.

Constantin: What are your thoughts?

#2 - 02/19/2018 11:42 AM - Constantin Asofiei

Some thoughts about this:

1. module navigation - I'm working on determining the 4GL code which computes the available modules (I suspect it contains some kind of user-specific filtering). Beside this, I need to know what kind of JS component to use for it - is it OK some kind of a JS tree? Anyway, it will be pluggable.
2. toolbar - this is specific to the window and/or visited tab page. I think I can build some TRPL rules to emit a 4GL function to return the handle for a frame which contains the toolbar. But, there are some issues:
   - if business logic decides to disable/show/whatever a button AFTER it was built on JS side, it complicates things. I can check in the 4GL embedded driver if, for an incoming 4GL event request, the target widget is still visible/enabled on the business-logic side, and ignore the request if is not. But, otherwise, is not that easy to make the JS side aware of 4GL changes in the toolbar's state, without some more complex changes.
   - the toolbar usually contains buttons; if this is all automatic, I expect that the JS widgets will have the same L&F (i.e. image, text, three-d, to

the extent possible) as the 4GL counterpart. So, we need to transfer to the JS side as many widget configuration as possible (image, text, location, size, etc) to be able to build the toolbar.

3. logout prompt/integration - this should be simple, just add a custom function to be called on the logout button press

#3 - 02/28/2018 12:08 PM - Constantin Asofiei

Greg, I think I need a SESSION:DISABLE-REDRAW attribute. This would prevent any window to be shown on screen, until this attribute is set to false. The reason I need this: I want to be able to run an ADM program (which i.e. builds module navigation) without its associated window being drawn on screen. This will help getting the module information without having to re-write business logic related to user rights and other stuff.

#4 - 02/28/2018 12:15 PM - Greg Shah

I'm OK with this idea.

I assume it will be compatible with the WINDOW:DISABLE-REFRESH? In other words, when SESSION:DISABLE-REFRESH is TRUE, WINDOW:DISABLE-REFRESH will return TRUE even if it was not assigned explicitly? I think the opposite would not be correct. If SESSION:DISABLE-REFRESH is FALSE, then WINDOW:DISABLE-REFRESH will return its specific value.

#5 - 02/28/2018 12:19 PM - Constantin Asofiei

Greg Shah wrote:

> I'm OK with this idea.
>
> I assume it will be compatible with the WINDOW:DISABLE-REFRESH? In other words, when SESSION:DISABLE-REFRESH is TRUE,
> WINDOW:DISABLE-REFRESH will return TRUE even if it was not assigned explicitly? I think the opposite would not be correct. If
> SESSION:DISABLE-REFRESH is FALSE, then WINDOW:DISABLE-REFRESH will return its specific value.

The idea of the SESSION:DISABLE-REDRAW is that it affects all future created windows, not just current ones - it will catch even window realization on driver-side - i.e. will post-pone it until SESSION:DISABLE-REDRAW gets set to FALSE.

So I'm not touching WINDOW:DISABLE-REDRAW or any other widget-level DISABLE-REDRAW attribute, this SESSION:DISABLE-REDRAW is a kind of global flag which disables everything on the UI.

#6 - 02/28/2018 12:32 PM - Greg Shah

> Beside this, I need to know what kind of JS component to use for it - is it OK some kind of a JS tree? Anyway, it will be pluggable.

Yes, some kind of tree is OK. In fact I think it is needed to make things work well for a user.

Many customers are trying to use toolkits like Consultingwerk's WinKit or BinteQ's Fly2Pro to put a .NET "skin" or "shell" around their existing 4GL GUI code.

A common improvement by these customers is some kind of side-panel (on the left) with a tree control that allows the user to select the current module. This can be completely JS based and has no legacy visualization requirements. The customers I've spoken with don't care if our approach looks different from the .NET stuff since the .NET stuff is just being used as a stop-gap measure to provide their users with something more up to date than a 1990s style GUI. Of course, the 2007-style .NET GUI is hardly "modern" by web standards. Because of this we have flexibility in how we visualize this. The default should be attractive and functional. A pluggable approach is also very nice.

If we use this sidebar approach, it should have a responsive design and the user should be able to easily shrink/expand it to that they can maximize space usage when it is not needed.

Eric can also help evaluate the ideas for the visualization of this.

But, otherwise, is not that easy to make the JS side aware of 4GL changes in the toolbar's state, without some more complex changes.

I understand. We will "cross that bridge" if and when we come to it.

I expect that the JS widgets will have the same L&F (i.e. image, text, three-d, to the extent possible) as the 4GL counterpart. So, we need to transfer to the JS side as many widget configuration as possible (image, text, location, size, etc) to be able to build the toolbar.

I understand. However, we may not need the full set of details.

One of the most common .NET skinning improvements is to turn toolbars into a .NET ribbon-bar (of the style of MS Office).

My personal opinion is that this is a terrible UI choice. The ribbon bar is not very user friendly and it certainly is not a good choice for a "modern UI" to be used in web/mobile. The customers to which I've spoken about this agree and are open to an approach that is fully web/mobile enabled.

We have an opportunity to improve things here and direct pixel-level matching is not needed.

#7 - 02/28/2018 12:33 PM - Greg Shah

I'm not touching WINDOW:DISABLE-REDRAW or any other widget-level DISABLE-REDRAW attribute, this SESSION:DISABLE-REDRAW is a kind of global flag which disables everything on the UI.

OK.

#8 - 02/28/2018 01:32 PM - Constantin Asofiei

For both the toolbar and tree, for now I'll be using Dijit and look for something else once I have all the framework stable.

Greg Shah wrote:

> We have an opportunity to improve things here and direct pixel-level matching is not needed.

For the toolbar, the legacy code has associated labels, too, beside the image. The problem here is that I'm not sure how to replace the image, and not use the legacy one. Can we use some mapping from the legacy ico name (i.e. plus.ico) to have some specific CSS rules?

If we want to use the legacy image, I will need to add some APIs to retrieve it from the client-side.

#9 - 02/28/2018 02:59 PM - Greg Shah

> For both the toolbar and tree, for now I'll be using Dijit and look for something else once I have all the framework stable.

A working placeholder is fine for now.

Eric will be helping with the replacement, most likely using D3. We have discussed some ideas:

https://bl.ocks.org/mbostock/1093025

- Good
  - the vertical design is intuitive to use and business oriented
  - a working expand/collapse model
- Bad
  - links are handled badly
  - nodes are too wide and are ugly
  - spacing is bad

https://bl.ocks.org/kerryrodden/766f8f6d31f645c39f488a0befa1e3c8

- Good
  - Breadcrumbs are a nice modern approach to orienting a user.
- Bad
  - Sunburst design is too busy to be useful for text.
  - Limits of the circular design (see below).

https://www.jasondavies.com/coffee-wheel/

- Good
  - Displays a large amount of information about the menu hierarchy at one time.
  - Using a larger variety of colors to differentiate groupings is very attractive.
  - The animation is awesome.
- Bad
  - Not intuitive to know how to drill in and out.
  - Possibly too much color usage.
  - Limits of the circular design (see below).

http://mbostock.github.io/d3/talk/20111116/pack-hierarchy.html

- Good
  - Grouping of related items is useful.
  - Zooming in and out is intuitive.

- Bad
    - The color scheme is poor.
    - The use of circles is not optimal for text.

https://rawgit.com/lgrkvst/d3-sunburst-menu/master/demo/d3-sunburst-demo.html

- Good
    - A working menu design that is not far off.
    - Semi-intuitive to use, but does take just a bit of experimentation to learn.
    - Eric likes the color gradients.
- Bad
    - Circular design is hard to use because of the text-heavy nature of a menu. The varying text orientation makes it difficult to read and thus it is harder to use. We can't use a circular design for our menu.
    - The font and the "bouncy" animation make it a bit too cartoon-like.
    - The ability to move it around the screen is not useful and it actually makes it harder to use in my opinion.

My intuition suggests that the following might work:

- Start with the interaction model of the collapsible indented tree.
- Remove the links.
- The default visualization is simply the top level menu items (first level of the tree) as a stack of rectangular items. Perhaps they have different colors without going too far with the color scheme.
- Hover or click on any of these will expand that top level item to be a bigger rectangle that contains its entire sub-hierarchy inside it. The 2nd level of the tree would be larger boxes that each contain the 3rd level items and so forth. The assumption here is that the application will not have too many levels to usefully display.
- Move the mouse away or click outside of the expanded top level item will collapse it.
- A click on a contained leaf item will trigger its action.
- All contained items will use different shades of the same base color of the top level item. The top level item probably has the lightest shade and the lowest level items are darkest.
- Use animation to add some sizzle without going so far as to make it annoying (or slow) to use.

#10 - 02/28/2018 03:03 PM - Greg Shah

In regard to the toolbar, we do not want to implement the features of the ribbon bar. Instead, we are focused on making a modern looking/attractive toolbar replacement that is very native web/mobile. It is likely that the buttons images will need to be replaced to look good in whatever control is built.

> The problem here is that I'm not sure how to replace the image, and not use the legacy one. Can we use some mapping from the legacy ico name (i.e. plus.ico) to have some specific CSS rules?

> If we want to use the legacy image, I will need to add some APIs to retrieve it from the client-side.

I think we should assume that the legacy images will be used. But we would encourage the customer to replace the images to match the modern look

and feel of the web toolbar we provide.

I do think these will need to be sent down or made available as graphics resources from the Java client.

#11 - 03/05/2018 12:51 PM - Constantin Asofiei

Something else which I'd like to discuss - what kind of security mechanism do we need, to ensure only certain users can reach a certain program? At a minimum, I can enforce the FWD 4GL embedded driver code to check against the programs which are part of the module tree. The idea is: I'd like some pluggable code to stand for rights verification, because I don't want to allow the 4GL code to run any program received from the JS side, without a security mechanism on the 4GL side.

Actually, as the p2j.embedded.js acts like a JS AppServer client (and can make any calls into the 4GL side via MSG_P2J_INVOKE), this security mechanism should be on a lower level... for the JS AppServer case, can I assume is the responsibility of the 4GL business logic in the target program to check for the rights to run the target program?

#12 - 03/05/2018 02:37 PM - Greg Shah

- Related to Feature #3258: improve javascript upcalls from the embedded client added

#13 - 03/05/2018 02:58 PM - Greg Shah

> what kind of security mechanism do we need, to ensure only certain users can reach a certain program

This is an issue that definitely does need to be resolved. I had planned it as part of #3258-1 (see item 1 in that note).

> Actually, as the p2j.embedded.js acts like a JS AppServer client (and can make any calls into the 4GL side via MSG_P2J_INVOKE), this security mechanism should be on a lower level.

Yes, exactly right.

> for the JS AppServer case, can I assume is the responsibility of the 4GL business logic in the target program to check for the rights to run the target program?

No, I think we need to implement the lower level approach as part of FWD. In other words, I think this is a general problem that needs to be fixed more broadly with a security resource plugin. All appserver clients (Java and JS) should have protection such that only specific entry points can be called remotely. For Java appserver support, don't we handle that today using RemoteEntryPointResource and checking in ControlFlowOps?

Does that only work for external procedures?

#14 - 03/05/2018 03:08 PM - Constantin Asofiei

First, thanks, I forgot about the RemoteEntryPointResource - that adds only a 1st-level protection, and there are no checks of what is executed via a program exposed to the JS AppServer client.

Greg Shah wrote:

> what kind of security mechanism do we need, to ensure only certain users can reach a certain program

> This is an issue that definitely does need to be resolved. I had planned it as part of #3258-1 (see item 1 in that note).

My problem here is that I need to protect what gets invoked via the fwd-embedded-driver.p... this program has APIs which receive as argument a program name and it invokes it with no security checks (currently). FWD can't use RemoteEntryPointResource, because the call is seen as internal, not from a remote side.

> for the JS AppServer case, can I assume is the responsibility of the 4GL business logic in the target program to check for the rights to run the target program?

> No, I think we need to implement the lower level approach as part of FWD. In other words, I think this is a general problem that needs to be fixed more broadly with a security resource plugin. All appserver clients (Java and JS) should have protection such that only specific entry points can be called remotely. For Java appserver support, don't we handle that today using RemoteEntryPointResource and checking in ControlFlowOps?

> Does that only work for external procedures?

Yes, that is only for external procedures. There are no limitations on what is invoked via those external procedures or its functions/internal procs, even if it was ran persistent.

#15 - 03/05/2018 04:16 PM - Constantin Asofiei

I think I got an idea for fwd-embedded-driver.p - we check a global var (holding a program handle); if the handle is valid, tries to invoke a pre-determined function (i.e. fwdCheckRights returns log (input program-name as char)) - if this function returns true for a specific program, then it will invoke it.

#16 - 03/05/2018 05:56 PM - Greg Shah

First, thanks, I forgot about the RemoteEntryPointResource - that adds only a 1st-level protection, and there are no checks of what is executed via a program exposed to the JS AppServer client.

Does the RemoteEntryPointResource get checked for the JS path today?

I agree that we want to make this a more thorough implementation.

My problem here is that I need to protect what gets invoked via the fwd-embedded-driver.p... this program has APIs which receive as argument a program name and it invokes it with no security checks (currently).

Long term, I would prefer to have fwd-embedded-driver.p written as Java code that is loaded from the FWD jar file. Is there a reason that won't work?

we check a global var (holding a program handle); if the handle is valid, tries to invoke a pre-determined function (i.e. fwdCheckRights returns log (input program-name as char)) - if this function returns true for a specific program, then it will invoke it.

Let's make sure we are using the SecurityManager for access control. Worst case we can write Java code for checking permissions as we did with the first big ChUI application (e.g. axxx.txxxx.mxxxx.security.*Resource, axxx.txxxx.mxxxx.security.*Rights and usage from axxx.txxxx.mxxxx.cfg.Config to do the security checks). Better case, we leverage existing ACLs and resource plugins.

I don't have a problem with direct integration into fwd-embedded-driver.p.

#17 - 03/06/2018 01:53 AM - Constantin Asofiei

Greg Shah wrote:

> First, thanks, I forgot about the RemoteEntryPointResource - that adds only a 1st-level protection, and there are no checks of what is executed via a program exposed to the JS AppServer client.

> Does the RemoteEntryPointResource get checked for the JS path today?

Yes.

> Long term, I would prefer to have fwd-embedded-driver.p written as Java code that is loaded from the FWD jar file. Is there a reason that won't work?

Hm... it should work, we can map this to a 4GL program name directly in SourceNameMapper, with all its APIs.

> Let's make sure we are using the SecurityManager for access control. Worst case we can write Java code for checking permissions as we did with the first big ChUI application (e.g. axxx.txxxx.mxxxx.security.*Resource, axxx.txxxx.mxxxx.security.*Rights and usage from axxx.txxxx.mxxxx.cfg.Config to do the security checks). Better case, we leverage existing ACLs and resource plugins.

The problem here is that we will be duplicating existing business logic security... and we are moving this away from legacy db-level user security to something else. My first thought is to use something simple which relies on the existing legacy security. I'll think about it.

Or do you want to replace the existing ACLs, etc with directory-based users/ACLs/etc?

#18 - 03/06/2018 08:37 AM - Greg Shah

> The problem here is that we will be duplicating existing business logic security... and we are moving this away from legacy db-level user security to something else. My first thought is to use something simple which relies on the existing legacy security. I'll think about it.

I was thinking this was an additional set of lower-level protections that ensure that some minimum consistent protection is provided.

Or do you want to replace the existing ACLs, etc with directory-based users/ACLs/etc?

The application-level security should still exist. In this case, is the application's 4GL code doing the security checks before the menu/toolbar action is dispatched?

#19 - 03/06/2018 08:51 AM - Constantin Asofiei

Greg Shah wrote:

I was thinking this was an additional set of lower-level protections that ensure that some minimum consistent protection is provided.

We just need to ensure we have a mechanism in place to allow security checks for programs ran via APIs from fwd-embedded-driver.p. We could provide a mix support - i.e. if there is a specific function available in some global handle var, delegate the checks to that, otherwise use our internal checks (which can be the RemoteEntryPointResource).

The application-level security should still exist. In this case, is the application's 4GL code doing the security checks before the menu/toolbar action is dispatched?

The app has no security checks at the ADM window code (at least I can't find one) - what is user-specific is only the module window. But I think they delegate the RUN call to a specific program which does some custom work... I think this requires a mixed support, too, to either delegate the call to a customer-specific procedure (if it exists) or, otherwise, use a direct RUN call.

#20 - 03/06/2018 08:56 AM - Greg Shah

I assume we should also check to ensure that only toolbar items that we have sent down to the JS side can be invoked. In other words, the application has created a list of valid toolbar items to display. We don't want a user to use JS code to upcall and invoke something to which the application wouldn't have already given them access. How do we map the toolbar items to the action invoked? Is there an ID that comes from the JS side that can be checked against the list of IDs that were sent?

#21 - 03/06/2018 09:01 AM - Constantin Asofiei

Greg Shah wrote:

> I assume we should also check to ensure that only toolbar items that we have sent down to the JS side can be invoked. In other words, the application has created a list of valid toolbar items to display. We don't want a user to use JS code to upcall and invoke something to which the application wouldn't have already given them access. How do we map the toolbar items to the action invoked? Is there an ID that comes from the JS side that can be checked against the list of IDs that were sent?

The toolbar and menubar events are executed via fwd-embedded-driver.p and rely on a APPLY <event> to handleId. (where <event> is choose, menu-drop, value-changed) - so, unless someone guesses a handle ID for some other resource, one can't call something not intended for that JS element's associated handled ID.

But your idea to secure this should be implemented - allow only known handle IDs to be processed by those fwd-embedded-driver.p APIs.

#22 - 03/07/2018 10:08 AM - Constantin Asofiei

The quicker solution is to just copy all icon resources to the embedded app, and allow the toolbar to load the image configured at the button's IMAGE attribute. Otherwise, getting the image as would be drawn in the legacy toolbar in the ADM window, is pretty tricky...

#23 - 03/07/2018 10:26 AM - Constantin Asofiei

I've got at least a case where a toolbar button shows another ADM window, with no way of going back - as the decorations are removed, its 'exit' button just closes this window and the 4gl business logic doesn't know that it needs to re-show another window... at some point there was a mentioning of some 'breadcrumbs' support, I think we need this.

#24 - 03/07/2018 12:38 PM - Greg Shah

- File material_components_toolbar_example.png added

One approach to a web and mobile friendly toolbar:

https://material-components-web.appspot.com/toolbar/index.html

The tabs list could be put into the hamburger menu on the toolbar. This will free up a lot of space and make the overall approach more consistent.

#25 - 03/07/2018 12:39 PM - Greg Shah

- File material_components_toolbar_example.png added

#26 - 03/07/2018 12:40 PM - Greg Shah

- File deleted (material_components_toolbar_example.png)

#27 - 03/12/2018 04:01 PM - Constantin Asofiei

I'm trying to build a static index.html using Google's material components toolbar, and its style conflicts with bootstrap. Also, other the components are conflicting with dojo (i.e. dojo-ready).

So looks like we either use material components without dojo/bootstrap or some other toolbar.

#28 - 03/12/2018 05:23 PM - Constantin Asofiei

I'm trying to implement the menubar using Google's material components, but it doesn't support cascading menus by itself.

http://www.material-ui.com/#/components/menu has some more complex components (i.e. cascading menus), but is meant for NodeJS...

#29 - 03/12/2018 06:30 PM - Constantin Asofiei

- File index.html added

Greg/Eric, so I've been trying to see what I can get from the material components - see attached, a static HTML file (all JS/CSS resources are hot-linked); the issue I'm having are these:

1. not compatible with bootstrap/dojo
2. require a custom menubar implementation - it doesn't have cascading menus

   Hover or click on any of these will expand that top level item to be a bigger rectangle that contains its entire sub-hierarchy inside it. The 2nd level of the tree would be larger boxes that each contain the 3rd level items and so forth. The assumption here is that the application will not have too many levels to usefully display.

You mean the behaviour should be something like this? https://codyhouse.co/demo/multi-level-accordion-menu/index.html

#30 - 03/12/2018 06:59 PM - Eric Faulhaber

Constantin Asofiei wrote:

> You mean the behaviour should be something like this? https://codyhouse.co/demo/multi-level-accordion-menu/index.html

I like this generally (particularly the polish added by the minimal animation in the jQuery implementation). Some improvements I would suggest to fit the needs of the large GUI application (and probably most ADM/ADM2 apps):

- Some sort of vertical containment, so long lists/subtrees don't scroll off the bottom of the screen. Perhaps contain it in a vertical scroll pane? I'm not sure about this one; it may violate some web application best practice. But it seems we have a "tall" set of subtrees, based on the organization of the large GUI application's modules, and limited vertical real estate.
- Collapse any other open submenu branch when a new one is opened, so the vertical size doesn't grow indefinitely. If in a scroll pane, animated

vertical centering of the newly opened sublist when it is opened.

- Look and feel customizations (font, font size, menu item width and height, colors, etc.).
- If we want to maximize the width available to the iframe, is it possible to have this menu hidden until a narrow visible portion is hovered over or clicked on? Then have it overlay the iframe temporarily until a menu item is selected? After the menu item is selected, the overlay is removed and the new screen is loaded into the iframe. Ideally, the overlay action would be minimally animated. Again, just an idea, but it may violate some best practice.

However, I realize all these suggestions may take this from something we can use out of the box to something we have to custom implement. I think this is pretty good for a first pass.

#31 - 03/12/2018 07:05 PM - Eric Faulhaber

Constantin Asofiei wrote:

> the issue I'm having are these:
> not compatible with bootstrap/dojo

bootstrap is nice because we have a few people already familiar with it, but it is not a hard requirement IMO. If we have something else to handle all the basic layout and media queries and it's reasonable to use, I'm ok with that.

I would prefer to not use dojo anyway; although I used their UI components for the code analytics tool, they feel clumsy/outdated to me and they were hard to integrate. On the last point, I haven't integrated any other JS UI frameworks, so maybe that's common.

> require a custom menubar implementation - it doesn't have cascading menus

How hard would it be to integrate something like the codyhouse menu?

#32 - 03/12/2018 07:35 PM - Eric Faulhaber

Greg and I also were discussing that the icon toolbar and what we currently expose as the bootstrap menu (i.e., the items converted from the ADM tabs) probably belong in the same horizontal toolbar/ribbon control above the iframe. Although the toolbar content may change partially as you navigate the former "tab" items to switch between ADM screens in the iframe, these two sets of navigation/function controls generally have the same life cycle. That is, both sets are associated with an individual module.

I envision the tool bar icons arranged horizontally in a top row, with the "tab" menu items as text arranged horizontally in a second row beneath it. Not sure how they should be aligned horizontally.

In the material demo you created, do these tab items correspond to the "Page {1|2|3}" menu items under the hamburger menu? Based on just running the smoke tests over and over, I get the sense that typical workflows involve switching tabs frequently, so we probably want to avoid forcing the user to open the hamburger menu before using them.

Not sure where the current "File" and "Help" top-level menus belong yet. If they are used less frequently, perhaps embedding them in the hamburger menu would be more appropriate.

Also, is the "Logout" menu item we have (in the current embedded mode) in the bootstrap menu of every screen appropriate? I think that became the "EXIT" menu item in the demo. We have to be careful as we rework the UI to not confuse users between the concept of exiting an individual module and exiting the application altogether.

I see an "Exit (F12)" menu item under the "File" menu in some of the ADM windows, but I haven't checked exhaustively through all the application's modules. Can one really log out of the application from every screen directly, or are there some workflows where you can't/shouldn't? I guess if none of the screens presented off the main menu window are modal, it always is possible to change focus to the main menu window from wherever and exit the app. So, maybe that's ok.

#33 - 03/12/2018 08:08 PM - Eric Faulhaber

I think this set of icons would be useful for the toolbar: https://materialdesignicons.com/. However, these are font icons, not images, so we would need a way to map from the existing artwork resources to these glyphs. It is not our responsibility to switch out all the artwork, but I think if we can present an example for one or two modules, it would be a good inspiration.

I first looked at http://www.flaticon.com, but IMHO most of the icons there are too busy for this purpose. I like the simplicity of the material design glyphs.

#34 - 03/13/2018 06:02 AM - Constantin Asofiei

Eric Faulhaber wrote:

> require a custom menubar implementation - it doesn't have cascading menus

> How hard would it be to integrate something like the codyhouse menu?

I couldn't find yet some appropriately licence component to use for the menubar and the left-side module (can't use codyhouse, I posted it to make sure the L&F I was thinking is on track with what you have in mind). For the module, we can use a drawer (https://material-components-web.appspot.com/drawer/index.html) to show/hide it.

I agree with the other notes: the menubar is better placed in the hamburger, while the pages and toolbar buttons above the iframe.

#35 - 03/14/2018 09:39 AM - Constantin Asofiei

This cascading accordion menu looks interesting, under MIT licence:
https://www.jqueryscript.net/menu/Smooth-Multilevel-Accordion-Menu-Plugin-For-jQuery-vmenu.html Can I use it?

#36 - 03/14/2018 10:18 AM - Greg Shah

I am OK with it, but Eric should decide.

#37 - 03/14/2018 10:59 AM - Eric Faulhaber

Yes, it looks good. The only thing that is a little disconcerting to me is the location of the twisty on the right instead of the left. Is this configurable? I assume other style characteristics (font, colors, etc.) are managed with CSS?

Also, it seems to be driven by nested <ul> HTML list tags instead of JSON data, but I guess that's ok. Is the idea to send JSON data down from the server and to manipulate the DOM to insert the list tags?

#38 - 03/14/2018 11:03 AM - Eric Faulhaber

Eric Faulhaber wrote:

> The only thing that is a little disconcerting to me is the location of the twisty on the right instead of the left.

This was more distracting for me on my phone than on a desktop, for some reason. I'm ok with it as is.

#39 - 03/14/2018 11:03 AM - Constantin Asofiei

Eric Faulhaber wrote:

> Yes, it looks good. The only thing that is a little disconcerting to me is the location of the twisty on the right instead of the left. Is this configurable?

It can be changed, in the CSS/image to stay on the right.

> I assume other style characteristics (font, colors, etc.) are managed with CSS?

Yes.

> Also, it seems to be driven by nested <ul> HTML list tags instead of JSON data, but I guess that's ok. Is the idea to send JSON data down from the server and to manipulate the DOM to insert the list tags?

Yes, there's no issue in creating dynamic <ul> and <li> tags from the JSON data.

#40 - 03/14/2018 11:20 AM - Eric Faulhaber

Sounds good, let's go with it.

#41 - 03/14/2018 11:43 AM - Constantin Asofiei

Greg, please review 3469a rev 11243.

#42 - 03/19/2018 03:57 PM - Greg Shah

Code Review Task Branch 3469a Revision 11245

The changes are quite good.

1. What is the idea being TC.processRepaints() overriding the disableRedraw flag (and not setting it back)?

2. Please add a file header to PDFPrintOutputStorage.

3. GuiOutputManager is missing a history entry.

#43 - 04/02/2018 02:22 AM - Constantin Asofiei

3469a was rebased from trunk 11243. Rev 11253 fixes the review notes.

#44 - 04/02/2018 07:28 AM - Greg Shah

Summary of open items (from Constantin):

1. Finish the JS for the new embedded widgets (toolbar, menubar, module tree) using Google Material Components.

2. Logout issue in #3469-1.

3. There are a few FWD issues in embedded mode, when navigating 'deeper' in z-order: if a window is shown above the embedded one, this one is hidden and not easily brought back to surface. This is valid regardless of how the window is opened (from the menubar/toolbar or from a button in the embedded window). This might be an issue of just showing the next window in z-order.

4. #3218 issues.

#45 - 04/02/2018 10:55 AM - Constantin Asofiei

Greg Shah wrote:

> 1. What is the idea being TC.processRepaints() overriding the disableRedraw flag (and not setting it back)?

The point here is to disable SESSION:DISABLE-REDRAW once we reach a point where blocking for input is required; as the screen will not be drawn and the user will not be able to interact.

#46 - 04/03/2018 08:30 AM - Constantin Asofiei

I'm using this project to get the material-design-icons webfonts: https://github.com/Templarian/MaterialDesign-Webfont

Is the licence OK? https://github.com/Templarian/MaterialDesign-Webfont/blob/master/license.md

#47 - 04/03/2018 09:26 AM - Greg Shah

> Is the licence OK? https://github.com/Templarian/MaterialDesign-Webfont/blob/master/license.md

Yes, it is OK. I assume you are going to embed the font(s). You will need to add a .txt file in that same directory that contains the license and explains which fonts are embedded/distributed under that license.

#48 - 04/03/2018 10:07 AM - Constantin Asofiei

Eric, for the menubar, beside https://jqueryui.com/menu/ I can't find anything suitable (and out-of-the-box, to some extent). Even this menu, if I try to show it from the left-side hamburger menu, I think there are some z-order issues (as it interacts with material-components).

Can you give me some hints what else I should search for, related to menubar?

#49 - 04/03/2018 02:35 PM - Eric Faulhaber

Constantin Asofiei wrote:

> Eric, for the menubar, beside https://jqueryui.com/menu/ I can't find anything suitable (and out-of-the-box, to some extent). Even this menu, if I try to show it from the left-side hamburger menu, I think there are some z-order issues (as it interacts with material-components).
>
> Can you give me some hints what else I should search for, related to menubar?

Why not keep it simple and use the same menu control for the hamburger menu as we use for the module tree? It would "slide out" and either overlay the content or push it to the right until a menu item is selected.

#50 - 04/03/2018 02:57 PM - Constantin Asofiei

Eric Faulhaber wrote:

> Constantin Asofiei wrote:
>
> > Eric, for the menubar, beside https://jqueryui.com/menu/ I can't find anything suitable (and out-of-the-box, to some extent). Even this menu, if I try to show it from the left-side hamburger menu, I think there are some z-order issues (as it interacts with material-components).

Can you give me some hints what else I should search for, related to menubar?

Why not keep it simple and use the same menu control for the hamburger menu as we use for the module tree? It would "slide out" and either overlay the content or push it to the right until a menu item is selected.

OK, I'll see how I can reuse this.

#51 - 04/03/2018 03:01 PM - Constantin Asofiei

Eric wrote:

But as before, we probably want to blend the iframe background with the browser client background.

Yes, I'll fix this.

The default purple for the top area is a little jarring. I wonder if we shouldn't pull a color ..., so it matches better.

I'll look for something more appropriate.

I was at first a little confused by the text under the toolbar. At first glance, I thought the text was meant to label the icons, but as you look closer, clearly they are separate. Should we visually separate them more? Not sure how to accomplish this.

I'll look for a proper separator.

The layout (especially of the top area) seems width-sensitive. What happens when it becomes narrower?

Nothing... they are not grouped in i.e. a drop-down menu. Will look how to do it.

Is the "title" text a breadcrumb? Is it informative only, or interactive?

It's informative, it displays the current module name and the selected page.

I assume the iframe dimensions are still fixed at this point, correct?

Yes, still fixed.

#52 - 04/03/2018 03:06 PM - Greg Shah

Perhaps the hamburger menu is better located to the right of the application's logo? It seems awkward to the left of it.

#53 - 04/03/2018 03:12 PM - Eric Faulhaber

Greg Shah wrote:

> Perhaps the hamburger menu is better located to the right of the application's logo? It seems awkward to the left of it.

I agree that it is a bit strange there, but I think it has to be on either the left or right edge of the browser for the menu effect to look natural. Perhaps it is always fixed as the right-most tool bar item (at least visually, though it might be a separate item physically, to allow the rest of the tool bar items to be swapped together with the module)?

#54 - 04/17/2018 01:00 AM - Eric Faulhaber

Constantin, there is a TRPL function generate_code_for_page_getPageHandles in rules/adm/adm_windows.xml. I see the getPageHandles methods in converted classes, but I don't see where generate_code_for_page_getPageHandles is ever invoked in FWD. Where does this happen?

I ask because I am seeing differences in getPageHandles output between conversion runs, where the assignSingle(h, <N>, <variable>) calls inside the pnum switch statement change their order. I am trying to determine whether this is a regression caused by an AnnotatedAst change I made in my current branch, or the natural variation in the iteration of an unordered map.

I am not seeing any other such ordering discrepancies in the entire code base, only in this specific location in getPageHandles (though across multiple source files).

#55 - 04/17/2018 02:19 AM - Constantin Asofiei

Eric Faulhaber wrote:

> Constantin, there is a TRPL function generate_code_for_page_getPageHandles in rules/adm/adm_windows.xml. I see the getPageHandles methods in converted classes, but I don't see where generate_code_for_page_getPageHandles is ever invoked in FWD. Where does this happen?

See adm_windows.xml line 207, and line 739.

> I ask because I am seeing differences in getPageHandles output between conversion runs, where the assignSingle(h, <N>, <variable>) calls inside the pnum switch statement change their order. I am trying to determine whether this is a regression caused by an AnnotatedAst change I made in my current branch, or the natural variation in the iteration of an unordered map.

> I am not seeing any other such ordering discrepancies in the entire code base, only in this specific location in getPageHandles (though across multiple source files).

I think this is because I'm using a HashMap in "handle-ref-ids" key of pageDetails - see line 662, which then gets iterated by generate_code_for_page_getPageHandles. I'll switch this to a LinkedHashMap to stabilize the generated Java code.

#56 - 04/17/2018 04:41 AM - Constantin Asofiei

Eric, if you want to be sure, convert with this patch applied to trunk, then with the patch applied to your branch and check for differences:

```
### Eclipse Workspace Patch 1.0
#P p2j
Index: rules/adm/adm_windows.xml
===================================================================
--- rules/adm/adm_windows.xml    (revision 1480)
+++ rules/adm/adm_windows.xml    (working copy)
@@ -659,7 +659,7 @@
                        <action>refId = ref.getAnnotation("refid")</action>

                        <rule>not pageDetails.containsKey("handle-ref-ids")
-                           <action>handleRefs = create("java.util.HashMap")</action>
+                           <action>handleRefs = create("java.util.LinkedHashMap")</action>
                            <action>pageDetails.put("handle-ref-ids", handleRefs)</action>

                            <action on="false">
```

#57 - 04/17/2018 11:03 AM - Greg Shah

See #3498-21 for a list of some ADM2 reference materials.

#59 - 04/26/2018 04:11 PM - Constantin Asofiei

- % Done changed from 0 to 70

3469a was merged to trunk rev 11252 and archived.

#60 - 07/02/2018 01:41 PM - Constantin Asofiei

Greg, I've been trying to find a smart/automated way to handle the z-order issues, when a third-party window (opened via i.e. toolbar) not belonging to the current ADM module gets shown (and hides everything else), but I can't find something feasible.

The solution I see at this time is to expose all top-level 4GL windows to the embedded mode (via a kind of mini-taskbar/breadcrumbs/etc), to allow the user to switch between them and/or close it, if the window doesn't have an explicit "Close" button (as the caption button to close it is not visible, because the decorations are not shown). The idea here is that the windows will be shown (their titles and a 'x' button to close it) in z-order, from left-to-right, with the visible window last.

Let me know if I should continue this way.

#61 - 07/03/2018 11:51 AM - Greg Shah

I've been thinking about this problem. If I understand correctly, the problem only occurs for non-dialog main windows that are not associated with a tab. Is that right?

Is there also an issue where a toolbar selection loads a different module? If this happens, do we switch the tabs properly?

The biggest issue here is that there is conceptual breakage between having a "window list" and having tabs.

Still, something does need to be done. Although breadcrumbs are natural for the web, they also normally imply a persistent URL which we won't have.

How about using chips? And these chips could be organized into a drawer or in a "sidebar" card. This drawer or card could be titled "Window List". Notice how the chips can have an X button and the currently shown window can be highlighted.

#62 - 07/03/2018 11:59 AM - Eric Faulhaber

Constantin Asofiei wrote:

> The idea here is that the windows will be shown (their titles and a 'x' button to close it) in z-order, from left-to-right, with the visible window last.

Do you intend to use the virtual desktop switcher as a basis for this, or something else? Where would it be located?

If multiple, non-modal windows can be opened by an application simultaneously, how will these be represented and share real estate in embedded mode? Is the idea to show the top one in the iframe, with no direct indication of any underlying windows (other than their representation in the switcher)? I'm just trying to visualize this...

#63 - 07/03/2018 01:53 PM - Constantin Asofiei

Greg Shah wrote:

> I've been thinking about this problem. If I understand correctly, the problem only occurs for non-dialog main windows that are not associated with a tab. Is that right?

Yes. But note that tabs are just one or more frames in the same ADM window (usually); there are also cases where selecting a 'page/tab', it will open another window.

> Is there also an issue where a toolbar selection loads a different module? If this happens, do we switch the tabs properly?

Didn't find one, but I'll try to code for this.

> How about using chips?

This is exactly how I visualized this. I'll try to use them.

#64 - 07/03/2018 01:56 PM - Constantin Asofiei

Eric Faulhaber wrote:

> Do you intend to use the virtual desktop switcher as a basis for this, or something else? Where would it be located?

See above. The idea here is that the embedded app will be notified by the FWD iframe of the existing FWD windows will build chips with each window title, allowing the user to switch between them and/or close it.

> If multiple, non-modal windows can be opened by an application simultaneously, how will these be represented and share real estate in embedded mode?

Only the last opened one will be viewed - embedded app currently has no way of discarding this non-ADM window explicitly (unless the window has an explicit 'Close' button).

> Is the idea to show the top one in the iframe, with no direct indication of any underlying windows (other than their representation in the switcher)? I'm just trying to visualize this...

Yes. The iframe will always see the top non-modal FWD window, unless there are modal windows opened - these will be 'stacked' on top of this FWD window owner. The switcher will allow you to navigate through other opened FWD windows.

#65 - 07/06/2018 10:11 AM - Greg Shah

- Related to Bug #3582: Add missing support for dynamic menu server updates added

#66 - 07/06/2018 10:36 AM - Greg Shah

FYI, I suspect that Hynek is not going to get to #3582 soon. Is this something you can take? I'm sure he can give you a head start.

Can you please summarize what is left open in #3469 and in #3218?

#67 - 07/06/2018 06:48 PM - Constantin Asofiei

Greg Shah wrote:

FYI, I suspect that Hynek is not going to get to [#3582](#) soon. Is this something you can take? I'm sure he can give you a head start.

Yes,I'll work on it, need a break from embedded mode...

Can you please summarize what is left open in [#3469](#) and in [#3218](#)?

What I have left is this:

1. **backed out, not usable** iframe scrollbars when the opened window is larger than then iframe (I'm trying to automate resizing the iframe, but a maximum width/height for the iframe is required to not mess up the embedded UI)
2. whatever else I find during testing the modules.
3. footer control ([#3469-1](#))

What is working at this time:

1. chips for window list, to resolve the z-index issue. You can close/switch between FWD windows - closing is done via emitting a APPLY window-close from the embedded app (I'm using the window ID as returned by the HWND attribute, as I don't want to expose all widget handles to the client-side, at this time).
2. send notification to embedded app if a window is closed by business logic (i.e. via a Cancel button), but only for ADM windows. Need to finish the case for non-ADM windows.
3. [#3218-30](#) - removing chrome and/or moving the window to top-left corner without changing the business logic attributes causes the coordinates to be out-of-sync for i.e. mouse processing - ~~reactivate 3218b~~
4. do not allow chrome'd windows (i.e. dialog boxes) to be moved outside iframe (to avoid iframe scrollbars)
5. embedded/JS-side notification when business logic changes a page on its own, in this module
6. window's message/status area outside of window - is working only for windows associated with a module.
7. open-mime-resource in embedded mode (not an issue, browser was blocking popups)

#68 - 07/07/2018 09:07 AM - Greg Shah

do not allow chrome'd windows (i.e. dialog boxes) to be moved outside iframe (to avoid iframe scrollbars)

What about a situation where the dialog is sized larger than the iframe in one or both dimensions? I think such a thing might be common.

Other questions:

- Didn't we have some deadlocks with certain window usage?
- Are the toolbars fully working?

- Is the module list fully working?

#69 - 07/07/2018 10:32 AM - Constantin Asofiei

Greg Shah wrote:

> What about a situation where the dialog is sized larger than the iframe in one or both dimensions? I think such a thing might be common.

Yes, I'll consider this and allow a dialog larger than the iframe to be moved outide of it.

- Didn't we have some deadlocks with certain window usage?

I don't recall this...

- Are the toolbars fully working?

I need a scenario to test resource downloading, via i.e. open-mime-resource or something else. Eric, can you point me to a window which has this? Otherwise, the toolbar and emulated menubar components are working.

- Is the module list fully working?

Yes.

#70 - 07/07/2018 12:38 PM - Eric Faulhaber

Constantin Asofiei wrote:

> Eric, can you point me to a window which has this?

Look at #3613, #3615, and #3617.

#71 - 07/07/2018 03:17 PM - Constantin Asofiei

Eric Faulhaber wrote:

> Look at #3613, #3615, and #3617.

For #3613 nothing is happening.

#3615 works only for Web Virtual Desktop - lots of PDFs open. For Web Embedded client, is not working, I assume because p2j.webRoot is blank - shouldn't this be something like https://localhost:7449/? Anyway, I've tested a window.open("https://www.google.com", "_blank") from within p2j.socket.js's buildOpenResourceUrl, and is not working on embedded mode. Some googling shows that it should be possible, but I don't know if this is a browser restriction or something else. If I can't find a way to make it work directly from the FWD iframe, I think what remains is to send a message to the embedded app window and let it handle the window.open.

#72 - 07/09/2018 11:24 AM - Greg Shah

One other improvement that I think is important is to implement a default embedded mode application into the FWD server. The idea is that instead of running a separate embedded mode application server, we should have one built in and available using the same jetty environment as the rest of the system.

For most cases, this will be the optimal way to deploy since:

- It greatly reduces the amount of configuration needed as well as the complexity of the solution.
- Custom code and configuration can still be utilized if needed, but most customers will have little of that unless and until they start doing pure web development (and upcalls). At first, most customers probably won't do this.
- We should not make any changes that would stop a customer from implementing their own custom web application as a separate process.

#73 - 07/18/2018 06:01 PM - Constantin Asofiei

Greg, the default embedded web server is mostly working (allowing all the resources to be handled from the app jar, not FWD), but:

- do we want this to be ran on different port or under /embedded context, with adminPort (i.e. https://localhost:7443/embedded)
- for example, the Hotel GUI - we can make this to automatically start the embedded server, and the Hotel-specific resources be copied into com/goldencode/hotel/embedded) - like jquery, d3, images, fonts, etc. The standalone embedded web server will work, but to bring the JS code to work with the built-in FWD embedded server, there must be:
  - /custom context to handle any app-specific resources (likes images, css, custom JS scripts)
  - /fwd/ context to handle any common FWD resources (like p2j.module.js and other scripts)
  - index.html will be searched first in the {package-root}/embedded/index.html location and after that in the FWD jar - otherwise I don't see a easy way to let this be customized; although I don't know yet how a 'default' index.html would look.
- I'm still working on maximizing/standardizing the JS code from the embedded app, to be brought into FWD. Something I need to find is the best way to handle common JS libraries like jquery, Google Material Components, fonts - do we want separate zips committed to FWD? Because at least for Material Components stuff and fonts, there is no maven repo from which these can be retrieved via gradle. The alternative is to use web URLs for Google Material Components resources (from the official CDN at unkpg.com), but this will not allow offline usage.

I hope to finish this 'merger' this week/early next week.

I've edited note #3469-67 - what's remaining I think will be done next week.

#74 - 07/18/2018 06:33 PM - Greg Shah

The /embedded context makes sense.

> for example, the Hotel GUI - we can make this to automatically start the embedded server

I'm not sure what you mean here. The FWD server itself would handle the embedded mode as well as the virtual desktop mode.

Overall, I agree that taking things from multiple locations makes sense so that we can keep common code in FWD etc...

> Something I need to find is the best way to handle common JS libraries like jquery, Google Material Components, fonts - do we want separate zips committed to FWD?

I thought that gradle pulls JS zips from https://proj.goldencode.com/artifacts/javascript/ and that we created these zips ourselves. That would be the same here.

> The alternative is to use web URLs for Google Material Components resources (from the official CDN at unkpg.com), but this will not allow offline usage.

No, we need the offline usage.

#75 - 07/18/2018 06:38 PM - Constantin Asofiei

Greg Shah wrote:

> The /embedded context makes sense.
>
>> for example, the Hotel GUI - we can make this to automatically start the embedded server
>
> I'm not sure what you mean here. The FWD server itself would handle the embedded mode as well as the virtual desktop mode.

I should have worded it better, I meant that currently is working in a way that embedded server starts with the FWD server, with all Hotel GUI embedded resources in com/goldencode/hotel/embedded.

I thought that gradle pulls JS zips from https://proj.goldencode.com/artifacts/javascript/ and that we created these zips ourselves. That would be the same here.

Forgot about this, I was thinking of official repositories; will prepare zips for all resources.

#76 - 07/19/2018 04:34 AM - Greg Shah

I meant that currently is working in a way that embedded server starts with the FWD server, with all Hotel GUI embedded resources in com/goldencode/hotel/embedded.

This seems correct. Any generic resources should come from FWD but any application-specific resources should come from the application jar.

#77 - 07/19/2018 04:35 AM - Greg Shah

business code notification for page switch (I see no way other than ADM sources to do a PUBLISH that the page changed)

This seems reasonable. It is OK for us to modify the Possenet-based ADM code.

#78 - 07/23/2018 02:52 AM - Constantin Asofiei

Constantin Asofiei wrote:

I thought that gradle pulls JS zips from https://proj.goldencode.com/artifacts/javascript/ and that we created these zips ourselves. That would be the same here.

FYI, the zips in this folder have another zip inside with the same file content, which doubles the final file size. Was this by mistake?

#79 - 07/23/2018 08:54 AM - Greg Shah

I think that must be a mistake.

Hynek/Sergey: please see [#3469-78](#) and comment.

#80 - 07/23/2018 09:09 AM - Hynek Cihlar

Constantin Asofiei wrote:

> FYI, the zips in this folder have another zip inside with the same file content, which doubles the final file size. Was this by mistake?

I think this is a mistake, but I will let Sergey to confirm.

#81 - 07/23/2018 09:15 AM - Sergey Ivanovskiy

Yes, it seems that it was my mistake.

#82 - 07/26/2018 07:28 PM - Constantin Asofiei

3469b rev 11275 contains almost final changes. What is left is:

1. p2j.embedded.js cleanup
2. finish the JS 'merger' issue to run the embedded web app from FWD

I've updated [#3469-67](#), too.

#83 - 07/27/2018 08:10 AM - Constantin Asofiei

Greg, if you want to review 3469b 11275, please go ahead.

#84 - 07/27/2018 12:58 PM - Constantin Asofiei

I've rebased and updated 3469b to 11277 - Greg, please review.

#85 - 07/27/2018 03:25 PM - Greg Shah

Code Review Task Branch 3469b Revision 11277

Sorry it took a while for the review. I think the changes are really good.

Have you already tested Hotel GUI (swing and virtual desktop mode)? If so, then I think this can be merged to trunk.

#86 - 07/27/2018 04:39 PM - Constantin Asofiei

Greg Shah wrote:

> Code Review Task Branch 3469b Revision 11274

I assume you reviewed 11277?

#87 - 07/27/2018 05:36 PM - Eric Faulhaber

Constantin Asofiei wrote:

> Greg Shah wrote:
>
>> Code Review Task Branch 3469b Revision 11274
>
> I assume you reviewed 11277?

Strange, I could not bzr up from 11274 to 11277. I had to check out the branch from scratch instead.

#88 - 07/27/2018 06:57 PM - Greg Shah

Constantin Asofiei wrote:

> Greg Shah wrote:
>
>> Code Review Task Branch 3469b Revision 11274
>
> I assume you reviewed 11277?

Yes, sorry.

#89 - 07/27/2018 07:01 PM - Constantin Asofiei

OK, I have Hotel GUI up-to-date with 3469b rev 11279 (added p2j.emain.js) and is working properly. I can merge this to trunk if that's OK.

What is left is Hotel GUI configs to work with embedded web server from within FWD main app.

#90 - 07/27/2018 07:15 PM - Greg Shah

Yes, please merge to trunk.

#91 - 07/27/2018 07:27 PM - Constantin Asofiei

- Status changed from New to WIP

3469b was merged to trunk rev 11275 and archived.

#92 - 08/10/2018 01:23 PM - Constantin Asofiei

From the initial list in #3469-1, only this is left:

- footer control - provide a mechanism for the application to provide static content that can be placed in the footer links, copyright notice...)

#93 - 08/10/2018 05:33 PM - Constantin Asofiei

Constantin Asofiei wrote:

From the initial list in #3469-1, only this is left:

- footer control - provide a mechanism for the application to provide static content that can be placed in the footer links, copyright notice...)

A thought about this: via P2J-REMOTE-CALL, the 4GL business logic can potentially drive any of the embedded UI - it can call remote APIs to i.e. change the logo, footer text, etc. So I don't think we need something special here.

#94 - 08/23/2018 03:09 PM - Constantin Asofiei

- Related to Feature #3700: in embedded mode, show a loading overlay when the server-side has control (of long-running tasks?) added

#95 - 03/18/2019 03:16 PM - Greg Shah

- % Done changed from 70 to 100

- Status changed from WIP to Closed

Files

material_components_toolbar_example.png	2.62 KB	03/07/2018	Greg Shah
index.html	8.36 KB	03/12/2018	Constantin Asofiei