Subject: Docking package (plus examples)
Posted by mrjt on Tue, 15 Jan 2008 13:24:03 GMT
View Forum Message <> Reply to Message

This is a window docking framework designed to give the end-user and developer complete control over their workspace by floating/docking/hiding windows. The aim is to emulate and improve upon the a combination feature set offered by Qt and Visual Studio, while retaining Upp style and ease-or-use for the developer.

Currently the package is stable, almost feature complete and quite polished. There have been no major bugs for a while now and I use it myself in a large and complex application.

Source code and examples can be found in this thread (work back from the end until you find it) or the bazaar section of the SVN. It is also now distributed in Upp releases, under bazaar, but that may not be up-to-date. An example win32 exe is attached.

Full feature list:
- API documentaion included
- X11 support with native windows
- Animated Autohide
- Animated Docking/Undocking
- Full Serialization support
- Tabbed windows with nesting (one tabbed window inside another)
- Support for popup docking hints (sort of like Visual Studio). Inherit from PopUpDockWindow.
- Fully themed tabs (includes grouping/tab arrangemen with CTRL key)
- Vertical or Horizontal docking handles, chameleonized
- Transparent docking hints (when dragging over tabs)
- The ability for the user (or API) to create/save/load layouts)
- Window grouping support (organise windows by user or developer designated group)
- A window manager that allows the user to manage (including create/delete) groups and layouts (see screenshot)
- Predefined menus that can be added directly to an application
- Many configurable options

last update SVN Revision 268.

James

File Attachments
1) DockingExample2.zip, downloaded 901 times

Subject: Re: Docking / DockTest packages
Posted by forlano on Tue, 15 Jan 2008 18:19:14 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 15 January 2008 14:24I've recently refactored and improved the Docking package in the SVN (Docking is the main source, DockTest is an example). Current features

include:

- Normal (Simple) or Nested (Complex) docking
- Docking to/from tabs (tabs to be upgraded to the new style QuickTabs when available)
- Animated Autohide
- Animated Docking/Undocking (New)
- Vertical or Horizontal docking handles, chameleonized (New)
- Allow/Deny options for docking on a per window or all window basis, plus more configurable options (New)
- Partial support for size hints (to be improved)
- Partial X11 support (tearing/docking of windows works but there are problems with the behaviour of floating windows) (New)

This release should be very stable on XP but flaky on Linux. Unfortunately I'm unable to test it on Vista so if there are any problems please let me know. Any other bug reports or suggestions are also welcome.

I've attached the compiled DockTest package as a quick demo (Edit: re-uploaded with a minor improvement).

Cheers,
James

Very nice test case with animation.
I've not yet looked at the code. I wonder if it is possible to align the tabs vertically? I mean the text is horizontal but the tabs are on the left o right side of the ctrl.

Luigi

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Tue, 15 Jan 2008 19:10:44 GMT
View Forum Message <> Reply to Message

Hi!

Very nice indeed. And really useful.

I tested it a little under Vista, and I managed to break it a little. While moving around the tabs, I docked one on the top, near another one on the top, and it disappeared. Further tries to dock something on top behaved as if there where tree panels, one of them invisible.

Apart from this little bug which I couldn't reproduce yet, it all works great. There are some aesthetic problems, like those counter intuitive tab placements (but AFAIK you said you are going to use QuickTabs in the future), and the text is way to small and not really centered.

But these details are quite trivial and I hope in no time we'll have full docking support, both as you

implemented it and for menus and toolbars also.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 16 Jan 2008 10:50:13 GMT
View Forum Message <> Reply to Message

forlanoI wonder if it is possible to align the tabs vertically? I mean the text is horizontal but the tabs are on the left o right side of the ctrl.
cbpporterThere are some aesthetic problems, like those counter intuitive tab placements (but AFAIK you said you are going to use QuickTabs in the future), and the text is way to small and not really centered.
Tabs will indeed be improved, I haven't spent any development time on them recently since I've been waiting for an official tab-bar implementation. I'll hopefully add various tab alignment options once this is done. All the other visual elements are thoughly chameleonized and can be tinkered with very easily.

cbpporterApart from this little bug which I couldn't reproduce yet, it all works great.
It doesn't sound like a Vista bug, but is very similar to a bug I'm aware of with the animation system. As you say it's difficult to reproduce so I've had problems fixing it even though I know what the root cause is.

Thanks for your comments, I'm glad you like it. I've already started thinking about the toolbars, the principles are the same but the implementation if quite different. The trick is finding a clever and cohesive way of tying it all together, I believe Qt just adds it all to every main window by default and you have to disable it with compiler flags - a very ugly solution.

Cheers,
James

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Wed, 16 Jan 2008 11:16:23 GMT
View Forum Message <> Reply to Message

Quote:'ve already started thinking about the toolbars, the principles are the same but the implementation if quite different.
Well a simple solution could be to just change the skin and the grab bar, and insert the toolbar normally into the container. This should work for menus too. Problem is vertical/side-by-side stacking.

The ToolkitPro way, with visual hints is also interesting.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 16 Jan 2008 11:44:06 GMT

View Forum Message <> Reply to Message

My best idea so far is something like this:
1- A ToolBar is linked to a ctrl (the only one that it can be attached to), and then added as a frame normally.
2- Dragging the handle on the ToolBar tears it off and it is displayed as some sort of draggable popup (oddly, even Microsoft uses popups for this and I've already got all the necessary code).
3- When being dragged you have two options:
- if dragged to an edge of the owning ctrl you can add it as a new frame (this does stacking)
- if dragged over another toolbar it can be added to it as a child

This seems to solve most of the obvious problems, although I'm sure it won't be as easy as it sounds (frame ordering could be a problem, as will containing toolbars in other toolbars).

I've never (knowingly) seen ToolKit Pro, can you give me a rough idea what you mean?

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Wed, 16 Jan 2008 13:51:12 GMT
View Forum Message <> Reply to Message

Quote:I've never (knowingly) seen ToolKit Pro, can you give me a rough idea what you mean?
For some strange reason, Print Screen doesn't work when dragging those windows, so I can't paste a screenshot. But basically, when dragging a window over a component, over every border where a dock is possible, a little icon appears. When you move the cursor close to the icon, the animation kicks in and shows where the new position will be. This solves the problem of trying to position something somewhere and getting a lot of jumpy animations and stuff, but is a little more work for the user, because he has to bring the window near the icon, not near his desired position.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 16 Jan 2008 14:36:49 GMT
View Forum Message <> Reply to Message

That doesn't sounds very difficult to do. One of my original ideas was to allow different interfaces by inheriting from DockWindow (its possible, but not easy unless you have a good grasp of how it works internally), so I might have a go at implementing this. It actually sounds similar to the Visual Studio popups, only Microsoft's version is rather clumsy.

Subject: Re: Docking / DockTest packages
Posted by chickenk on Wed, 16 Jan 2008 15:25:12 GMT
View Forum Message <> Reply to Message

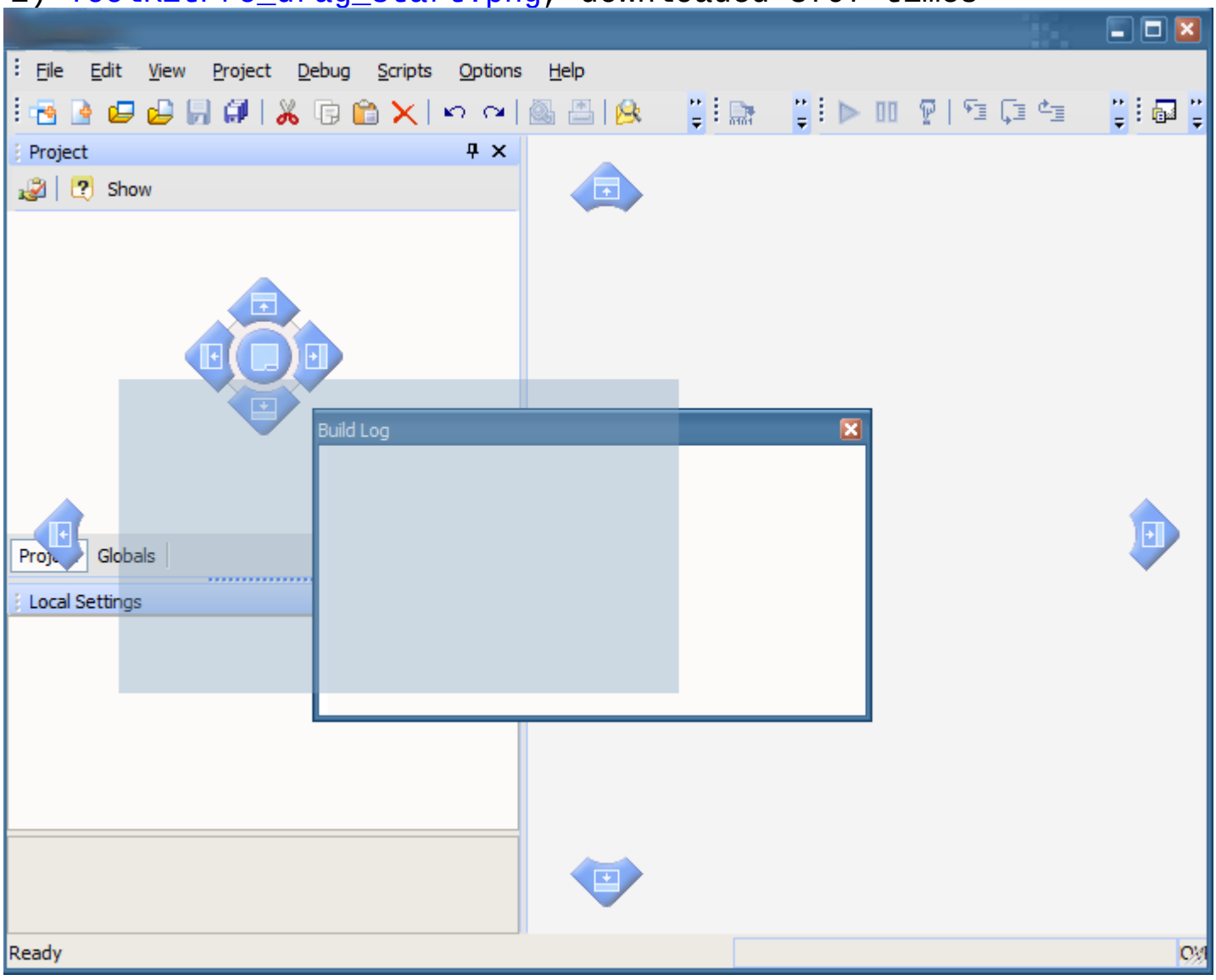
Here are screenshots from an app developed in my company, using ToolKitPro:

1. Start dragging the "Build log" window. We can see the hints appearing on the center of the docked window (the mouse cursor is in the docked window area so it is active) and on the borders of all its containers (in this case, the main container only).

Lionel

File Attachments

1) ToolKitPro_drag_start.png, downloaded 3797 times



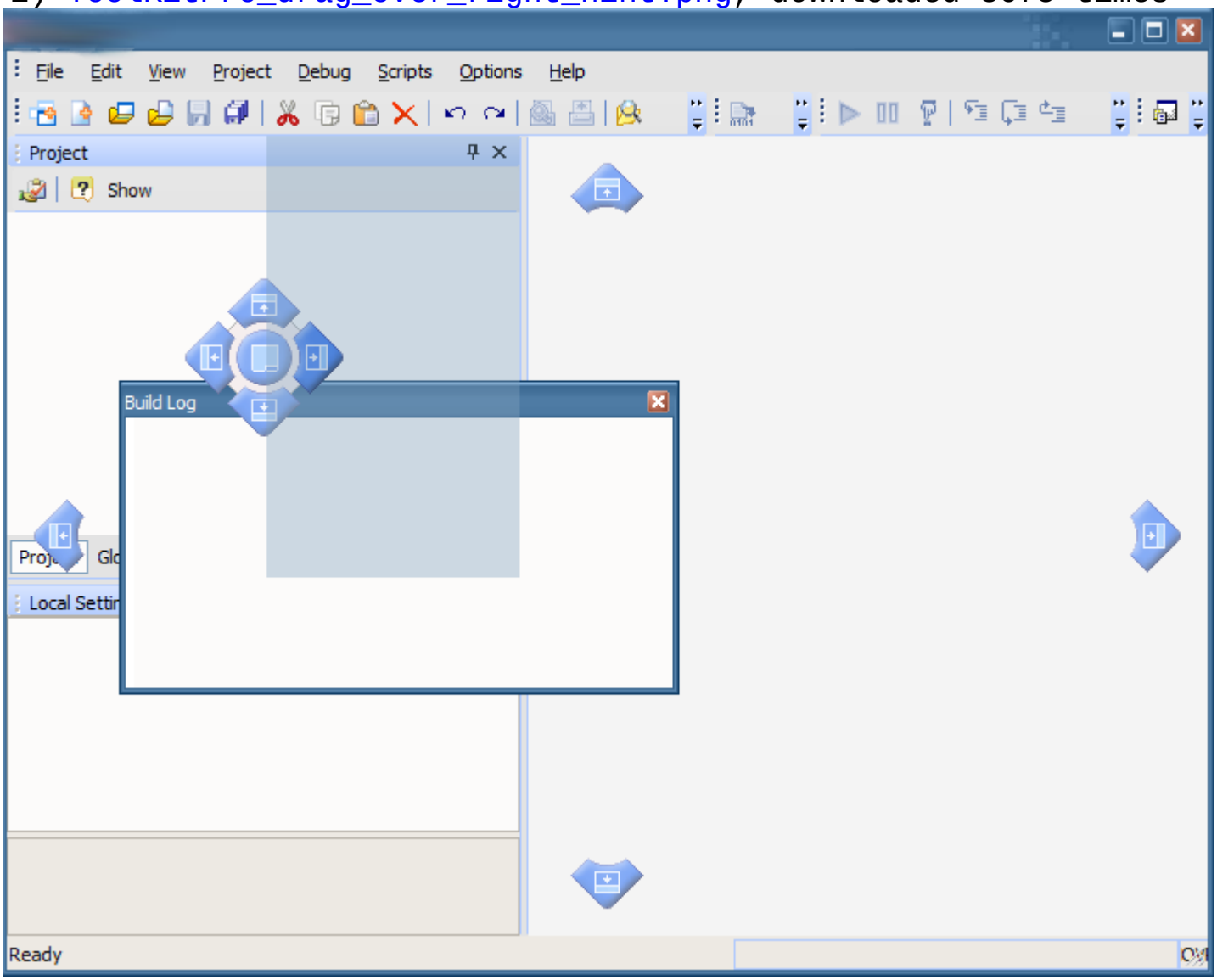
Subject: Re: Docking / DockTest packages
Posted by chickenk on Wed, 16 Jan 2008 15:27:46 GMT

2. Dragging the mouse cursor over the hint representing the right border of the docked pane. The hint color is changed, I think it would be nicer to change its opacity.

Lionel

File Attachments

1) ToolKitPro_drag_over_right_hint.png, downloaded 3675 times



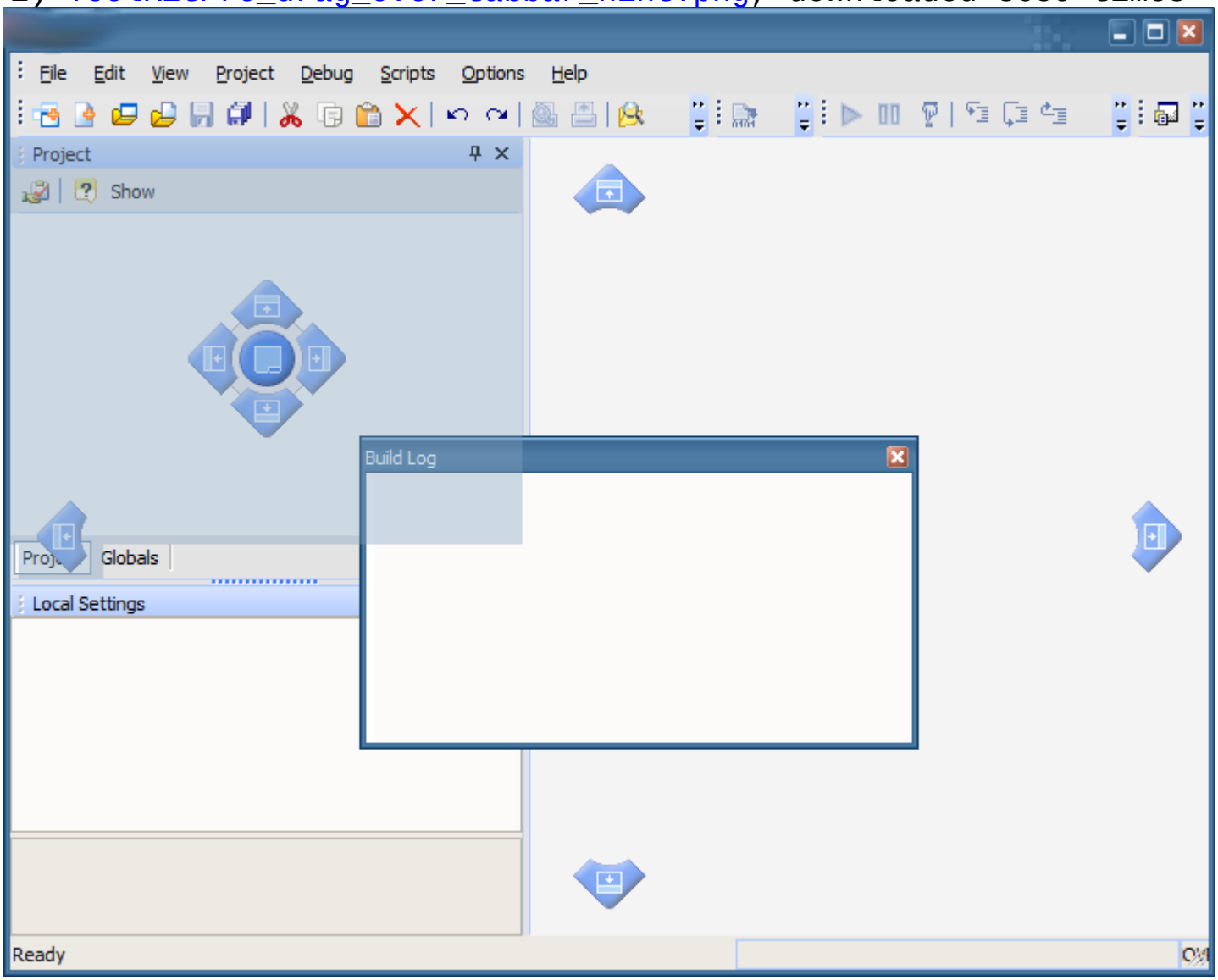
Subject: Re: Docking / DockTest packages
Posted by chickenk on Wed, 16 Jan 2008 15:30:20 GMT
View Forum Message <> Reply to Message

3. Dragging the mouse cursor over the hint representing a new tabbar item (note that there already is a tabbar: 'Project' and 'Globals' are 2 tabs). The anticipated area is not really optimal (the tab title rectangle is not at the place it will be displayed) but it's self-explanatory anyway.

Lionel

File Attachments

1) ToolKitPro_drag_over_tabbar_hint.png, downloaded 3659 times



Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 16 Jan 2008 17:51:40 GMT
View Forum Message <> Reply to Message

Thanks. The problem with reproducing that look is that AFAIK the irregularly shaped and partially

transparent popups/overlays are not possible with Upp at the moment. You can get close on Windows, but it causes all sorts of drawing problems, and IMO in V. Studio it doesn't work very well either. I believe it may not be possible at all on Linux, but I haven't verified this myself.

Having said that, it is possible to get something aproximating this. I've attached a quick demo, its got bugs and you'll have to excuse my icons as I'm shit at drawing . Note that this is implemented purely as inheritance, so both options (DnD/Popups) are available.

Subject: Re: Docking / DockTest packages
Posted by forlano on Wed, 16 Jan 2008 19:36:47 GMT
View Forum Message <> Reply to Message

mrjt wrote on Wed, 16 January 2008 18:51
Having said that, it is possible to get something aproximating this. I've attached a quick demo, its got bugs and you'll have to excuse my icons as I'm shit at drawing .

Hi James,

I've seen that feature in your code in the past, then disappeared in the next releases. Is there some reason? I found it very useful to help the user in positioning the tab... just the look need to be improved as you pointed out, but I like the overall effect.

Luigi

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Wed, 16 Jan 2008 22:30:21 GMT
View Forum Message <> Reply to Message

Quote:Having said that, it is possible to get something aproximating this. I've attached a quick demo, its got bugs and you'll have to excuse my icons as I'm shit at drawing . Note that this is implemented purely as inheritance, so both options (DnD/Popups) are available.
Very good . Who cares about transparency anyway? And I also like the icons. Just a little fine tuning and it will be great. I hope it makes it into CtrlLib or some other "official" package. But I laso like Oblivion's docker. He's looks a little better (even though the tabs could be smaller), but your's behaves a little bit nicer. Two implementations are better than one (sometimes).

Subject: Re: Docking / DockTest packages
Posted by mrjt on Thu, 17 Jan 2008 10:04:54 GMT
View Forum Message <> Reply to Message

forlano wrote on Wed, 16 January 2008 19:36I've seen that feature in your code in the past, then disappeared in the next releases. Is there some reason?
I ended up focusing mainly on the drag-drop approach since nobody said they wanted it done this

way, and when I had to rewrite it to add X11 support (that was a complete bitch!) I didn't bother with it. It was quick to add it back and I'd always planned on having different interfaces - in fact I've got a couple of less orthodox ones I'll try out once I've fixed some other things.

Quote:I hope it makes it into CtrlLib or some other "official" package. But I laso like Oblivion's docker. He's looks a little better (even though the tabs could be smaller), but your's behaves a little bit nicer.
The code comes in the bazaar section of the SVN, and I hope that once it's more polished it will be added the the officially numbered distributions (probably as a seperate package like GridCrl), but that's up to the U++ team.
Appearance-wise you could make it look identical to Oblivion's version (appart from the Tabs) just by changing the fontsize and handle images in the DockableCtrl Cham. style, and eventually I'll have to have more styles available. I personally quite like it with the much smaller handles.

James

Subject: Re: Docking / DockTest packages
Posted by mirek on Thu, 17 Jan 2008 15:36:32 GMT
View Forum Message <> Reply to Message

mrjt wrote on Thu, 17 January 2008 05:04
The code comes in the bazaar section of the SVN, and I hope that once it's more polished it will be added the the officially numbered distributions (probably as a seperate package like GridCrl), but that's up to the U++ team.

Actually, Bazaar was already shipped with last "dev", also somewhat hidden (no assemblies).

It is actually interesting question about integrating it:

- first, should it be placed as another nest in the MyApps assembly ("c:\MyApps;c:\upp\bazaar;c:\upp\uppsrc")?

- then, perhaps it should be divided into Bazaar and BazaarExamples, and BazaarExamples assembly precreated.

- also, I think it would be a good idea to write some description of package that would clearly designated it as Bazaar part, and with info about author(s). "Bazaar: Docking, by James Thomas"

Mirek

Subject: Re: Docking / DockTest packages
Posted by mrjt on Thu, 17 Jan 2008 18:27:47 GMT
View Forum Message <> Reply to Message

Seems sensible.

I've also now got a first version of dockable ToolBars/Menus, they were incredibly easy to implement, no more than a few hours coding . No changes to CtrlLib are required (yet) apart from the bugfix http://www.ultimatepp.org/forum/index.php?t=msg&th=3023& amp; amp;start=0&.

I've attached a version of modified version UWord as a demo. There are some display problems that may require CtrlLib changes to fix since MenuBar and ToolBar were never designed for some of this.

All thats missing is the ability to have multiple toolbars on one line.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Tue, 22 Jan 2008 12:34:17 GMT
View Forum Message <> Reply to Message

Docking, SVN Revision 99:
Complete X11 support using native windows
Fixed animation bug that prevented dockings to empty side docks.

Now that I have full X11 support I'm going to add docking to one of my applications, so now I won't have any choice but to support it

I've only tested properly on KDE (only briefly on Gnome), so if you have a different WM please test it and let me know of any problems.

James

Subject: Re: Docking / DockTest packages
Posted by Oblivion on Tue, 22 Jan 2008 20:06:46 GMT
View Forum Message <> Reply to Message

James, I'm going to upload the new version of DockCtrl to the SVN repository this week end. In case you need it, I Hope you'll find the "ripped" TabInterface (from Quicktabs) class useful for your version. It now draws left/right tabs (of course, with your hint for chameleon, thank you for it . I'm currently working on scrollbar of left/right tabs). As far as I know your code, it should be very easy to convert into it (probably only 4 or 5 lines of declaration, e.g. should change...)

Regards.

Subject: Re: Docking / DockTest packages

Posted by Oblivion on Fri, 25 Jan 2008 15:46:38 GMT
View Forum Message <> Reply to Message

James, I'm uploading the TabInterface Class source code with a basic example. It is now able to draw all four positions (left/right/top/bottom) and has scrollbar on all sides...

Also, as in the original Quicktabs, it supports icons, closebuttons, event handling, etc... I ripped out the unneeded part (file load/save and grouping) from the code, but it is quite easy if you want to put them back in (actually, If Daniel permits, I can add these changes to the current Quicktabs in the SVN too). It draws using chameleon so it is easy to skinnize...

I Hope You'll find it useful if you are going to use it in your Docking.

Regards...

File Attachments

1) TabInterfaceExample.rar, downloaded 739 times

Subject: Re: Docking / DockTest packages
Posted by unodgs on Sat, 26 Jan 2008 12:56:39 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Fri, 25 January 2008 10:46James, I'm uploading the TabInterface Class source code with a basic example. It is now able to draw all four positions (left/right/top/bottom) and has scrollbar on all sides...

Also, as in the original Quicktabs, it supports icons, closebuttons, event handling, etc... I ripped out the unneeded part (file load/save and grouping) from the code, but it is quite easy if you want to put them back in (actually, If Daniel permits, I can add these changes to the current Quicktabs in the SVN too). It draws using chameleon so it is easy to skinnize...

I Hope You'll find it useful if you are going to use it in your Docking.

I started to implement similar functionality in QuickTabs. You can see it in SVN bazaar from some time. However I think we could stay with your version as it seems to be completed. I only wonder if you could avoid somehow using IsVert() everywhere. I tried to "fix" Size an Point by swaping cx with cy and x with y where necessary. But it's up to you. IsVert() is not bad, but you must write a little bit more of code here and there. I also
wonder if TabInterface is a correct name. Couldn't we stay with QuickTabs?
My vision is to have one tabs system. So you'd have to integrate your code with current quicktabs code and restore grouping (frankly it can be a derived class if you want base tabs code to be clean).
Remember also that tabs in theide is a frame for editor. I mean editor is inside a frame

(FieldFrame exactely) which belogns to QuickTabs. That's why you could see r.xxx += 2. I saw you commented this. That's ok. But don't forget to solve this problem. Of course proper solution may be modifing theide code.
To finish this post: Please update quicktabs code. I'll be glad to see all this changes you've done in the main tree. Of course I'll keep my eye on this and help you if necessary.
PS: Default theming for tabs should be system theming like in TabCtrl.

Subject: Re: Docking / DockTest packages
Posted by Oblivion on Sat, 26 Jan 2008 14:25:08 GMT
View Forum Message <> Reply to Message

Quote:
I only wonder if you could avoid somehow using IsVert() everywhere.

Well, actually, as I stated before somewhere else, I first code then plan. It is hastily coded. I know It's not the best way of coding, but it gives some idea as to what not to do The "TabInterface" is a nested sub-class which is a part of DockCtrl (which uses only bottom tabbing -- so it has no IsVert() and neither it has std. tabctrl theming), so it is not intended to be a generic tab class or a Quicktabs replacement in the first place (it was just a sort of experiment); that's why I commented out those lines. And of course it is possible to remove all the silly IsVert(). I will find a better solution. Hopefully, this week.

Quote:
I also
wonder if TabInterface is a correct name. Couldn't we stay with QuickTabs?

Well, the coder of Quicktabs is you, so It's up to you to decide. Imo, Quicktabs is better too (since TabInterface is a DockCtrl sub-class)

And Thank you for your permission.

Regards.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Mon, 28 Jan 2008 09:44:49 GMT
View Forum Message <> Reply to Message

Quote:
I started to implement similar functionality in QuickTabs. You can see it in SVN bazaar from some time. However I think we could stay with your version as it seems to be completed. I only wonder if you could avoid somehow using IsVert() everywhere. I tried to "fix" Size an Point by swaping cx with cy and x with y where necessary. But it's up to you. IsVert() is not bad, but you must write a

little bit more of code here and there. I also
wonder if TabInterface is a correct name. Couldn't we stay with QuickTabs?
My vision is to have one tabs system. So you'd have to integrate your code with current quicktabs code and restore grouping (frankly it can be a derived class if you want base tabs code to be clean).
I have actually already solved all of these problems, but my code was horrible. I think the best control will actually be a merge between all three code bases, and I'm going to have a quick go at it this morning.

Quote:Remember also that tabs in theide is a frame for editor. I mean editor is inside a frame (FieldFrame exactely) which belogns to QuickTabs. That's why you could see r.xxx += 2.
Wouldn't the correct solution to this be to add the inner frame seperately? As you can see from the SVN example, when you have multiple TabFrames it looks horrible to have the frame drawn by the TabBar control.

Subject: Re: Docking / DockTest packages
Posted by unodgs on Mon, 28 Jan 2008 16:22:21 GMT
View Forum Message <> Reply to Message

[quote title=mrjt wrote on Mon, 28 January 2008 04:44]Quote:
I have actually already solved all of these problems, but my code was horrible. I think the best control will actually be a merge between all three code bases, and I'm going to have a quick go at it this morning.

That's great, but how to merge it? Oblivion is probably trying to modify quicktabs now as I gave him permission. I think we should wait for oblivion first then you could "fix" things you think you have better implemented/designed (after some discuss of course). Having this done we could finally have one codebase in svn repo.
Quote:
Wouldn't the correct solution to this be to add the inner frame seperately? As you can see from the SVN example, when you have multiple TabFrames it looks horrible to have the frame drawn by the TabBar control.
I know that. Separate frame seems to be the best solution.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Mon, 28 Jan 2008 17:52:41 GMT
View Forum Message <> Reply to Message

Fair enough. I've attached my latest version, please use any of the code if it helps (the rotated style stuff may be useful). File icons don't work properly, and there is a bug in the jump scrolling (in SetCursor), but otherwise it works quite well.

Edir: Fixed both of the above now. I hate leaving buggy code

File Attachments
1) QuickTabsTest.zip, downloaded 656 times

Subject: Re: Docking / DockTest packages
Posted by mrjt on Thu, 14 Feb 2008 16:30:27 GMT
View Forum Message <> Reply to Message

New version of Docking package now in the SVN. Automatic resizing is much improved, and it now uses themed tabs (these are a general version of QuickTabs).

Next on my list: Refactoring, documentation.

Subject: Re: Docking / DockTest packages
Posted by Oblivion on Wed, 27 Feb 2008 00:03:56 GMT
View Forum Message <> Reply to Message

hi James,

Last night I've compiled your Docking package on Linux (Kubuntu and Pardus). It seems that your current code does not posit the window correctly when it is dragged (so was mine). But I've managed to fix that issue very easily, so I would like to share my findings with you (for we are "unofficially" supporting each other). the trick lies in two places:

1. Add DockWindow::NoCenter() to the constructor of your dockable window's base class.

2. Open() the dockable window (or was it a container?) with no-owner.

This second one may not be valid for your code or brake something, I'm not sure about that. But I saw that you open the window by passing the "this" pointer to the methods(e.g, Open(this)). In case the second step fails, try to find a way to use it as such, I can confirm that It works perfectly

And another bug in you code (the latest one with transparent tab) I've encountered is regarding to your autohide code. Autohide a dock and try the autohide button or menu item when it is shown (in autohide mode). The framework will crash.

Hope these will help you.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 27 Feb 2008 09:25:39 GMT
View Forum Message <> Reply to Message

Hmm. It used to work, but then I foolishly didn't test the current SVN version on Linux before uploading it.

Oblivion wrote on Wed, 27 February 2008 00:031. Add DockWindow::NoCenter() to the constructor of your dockable window's base class.

2. Open() the dockable window (or was it a container?) with no-owner.

Thanks! I've been trying to avoid the Window auto-centering for ages, I can't believed I missed NoCenter!

Quote:And another bug in you code (the latest one with transparent tab) I've encountered is regarding to your autohide code. Autohide a dock and try the autohide button or menu item when it is shown (in autohide mode). The framework will crash.
Fixed.

Cheers!

Subject: Re: Docking / DockTest packages
Posted by mrjt on Fri, 29 Feb 2008 16:14:57 GMT
View Forum Message <> Reply to Message

I've just uploaded a massively improved version of the Docking package to the SVN (and attached to the first post)

What I've added:
Full Serialization support
The ability for the user (or API) to create/save/load layouts (my favourite)
Grouping support (being able to organise windows by group)
A window manager that allows the user to manage (including create/delete) groups and layouts (see screenshot)
Nested tabs windows
Transparent highlights (when dragging over tabs)
A slight delay before animation starts
Predefined menus that can be added directly to an application
Various TabBar improvements
Many, many configuration options and API improvements
2 new examples

The new Window Manager:

There are some known issues, mainly minor:
1 crash bug on Linux (seen once, never reproduced)
The 'intelligent' splitter layout needs improvement. It gets it wrong when it doesn't have enough space.
Some of the context menus need improvement
Rename group/layout needs to be added.
Group floating should layer the windows

General improvement of the Group tree neede (some of the problems are limitations with TreeCtrl)

It should be stable (jinx!), including on Linux (I double checked). The serialization stuff is new and complex but seems reliable.

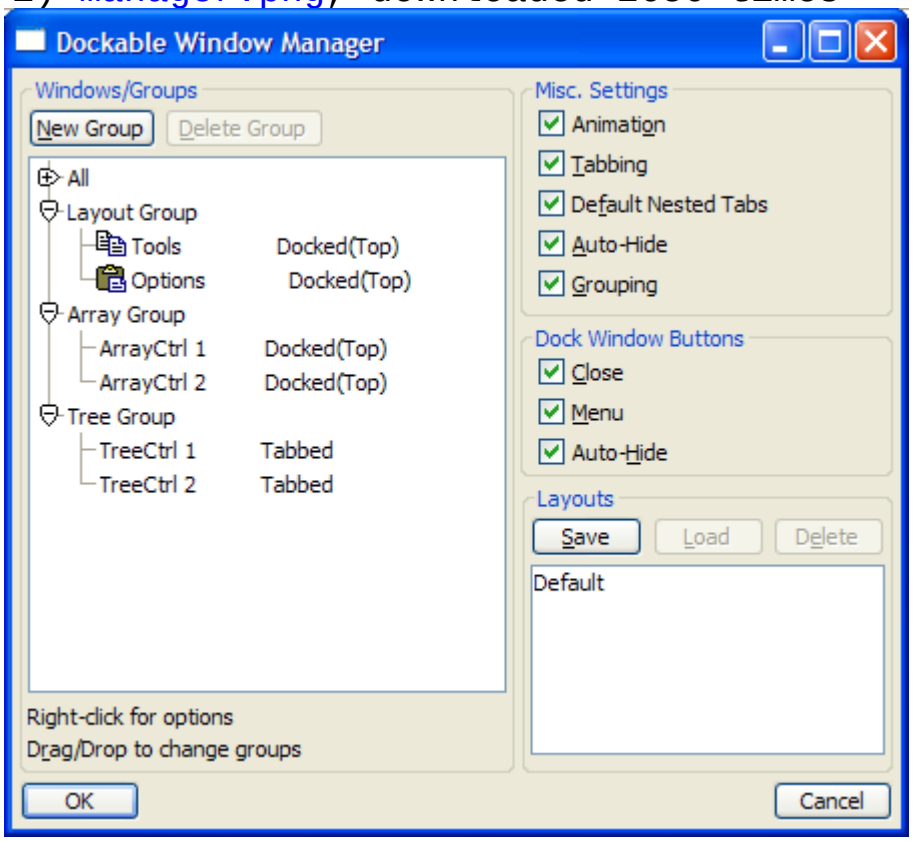
The API/class interfaces are all stable now too, there will be no further changes, only additions. I've dropped the complex docking mode (too much effort and it was preventing Serialization development).

The generalised TabBar ctrl I use is also now in the SVN, and I'm hoping to add source documentation to both packages soon.

James

File Attachments

1) manager.png, downloaded 2650 times



Subject: Re: Docking / DockTest packages
Posted by masu on Fri, 29 Feb 2008 20:31:05 GMT
View Forum Message <> Reply to Message

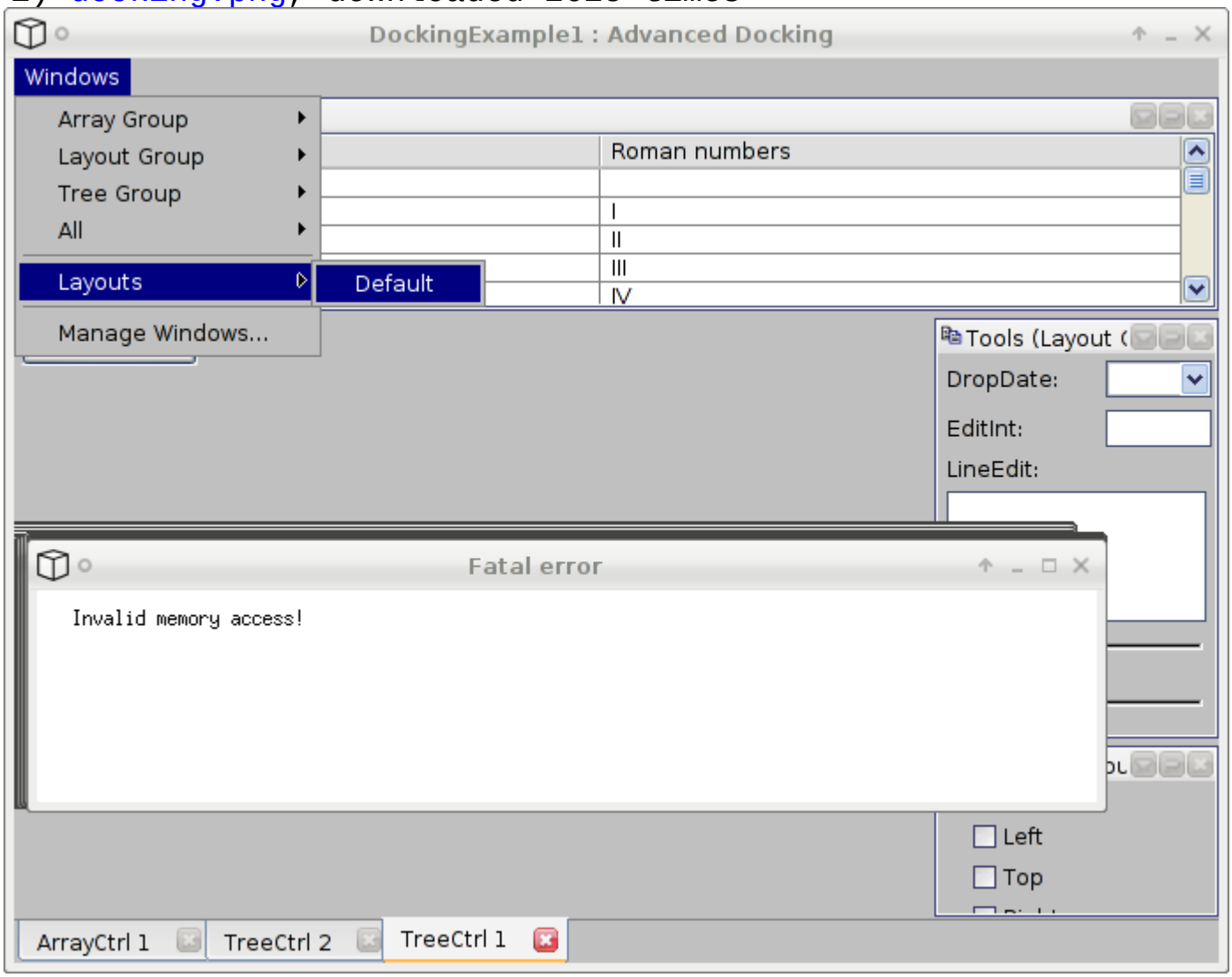
James, I got an error:

You can see, after I have organized a few windows in nested configuration and then want to go back to default layout, I get this error.

Matthias

File Attachments

1) docking.png, downloaded 2625 times



Subject: Re: Docking / DockTest packages
Posted by mrjt on Sat, 01 Mar 2008 01:15:25 GMT
View Forum Message <> Reply to Message

Thanks. I found the same one myself after uploading it.

I know exactly what the problem is, it's caused by trying to remove the nested windows in the

wrong order (before re-layout). Might take me a little while to fix, it's a bit of a tricky one.

I knew I'd jinxed it

Subject: Re: Docking / DockTest packages
Posted by mrjt on Sat, 01 Mar 2008 11:01:59 GMT
View Forum Message <> Reply to Message

Fixed. New code is in the SVN and attached. I've also updated the executable in the first post.

The moral of the story: Don't do clever things in ChildRemoved. It always causes more problems than it solves

James

Subject: Re: Docking / DockTest packages
Posted by mirek on Sat, 15 Mar 2008 13:44:07 GMT
View Forum Message <> Reply to Message

I feel pity that I have not tested this sooner

Well, mrjt, I like this. This is pretty good stuff, and I like the style of code too

Do you think it is ready enough to be included in 2008.1? (As a new addition, it does not have to be perfect yet, just useful

Mirek

Subject: Re: Docking / DockTest packages
Posted by mrjt on Mon, 17 Mar 2008 09:47:57 GMT
View Forum Message <> Reply to Message

Quote:Well, mrjt, I like this. This is pretty good stuff, and I like the style of code too
Such a compliment!

Quote:Do you think it is ready enough to be included in 2008.1?Almost. One thing I need to do first is to finalize the text so I can include a .t file, but I'll get this done today or tomorrow.

The long list of other improvements can be done later.

James

Subject: Re: Docking / DockTest packages
Posted by mirek on Mon, 17 Mar 2008 11:25:42 GMT
View Forum Message <> Reply to Message

mrjt wrote on Mon, 17 March 2008 05:47Quote:Well, mrjt, I like this. This is pretty good stuff, and I like the style of code too
Such a compliment!

Quote:Do you think it is ready enough to be included in 2008.1?Almost. One thing I need to do first is to finalize the text so I can include a .t file, but I'll get this done today or tomorrow.

The long list of other improvements can be done later.

James

BTW, have you checked with Gnome? I was not able to get it run there...

Mirek

Subject: Re: Docking / DockTest packages
Posted by mrjt on Mon, 17 Mar 2008 11:32:22 GMT
View Forum Message <> Reply to Message

No, but I'm installing it this afternoon so I'll check. I have a nasty feeling that the code I use to detect window moves is Window Manager specific, but we shall see.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Tue, 18 Mar 2008 15:32:26 GMT
View Forum Message <> Reply to Message

I've updated the SVN with a verison that includes a .t file. Additional changes:
Improved context menus
Various general bugfixes
Fixed major Gnome problems:
- Gnome seems to treat ToolWindows similarly to pop-ups, and this was breaking the dragging code.
- Upp::Animate was crashing the app when applied in reverse. This may be because the popup was getting set to 0 size. Temporarily disabled for Linux, but this doesn't work very well on Windows either.

The Theming is not very good on the TabBar and in general it doesn't work quite as well on Linux as Win32, but at least it *should* work now. Please let me know if you have any problems.

Next priorities:
Improving the 'smart resize' algorithm for better animation/docking results
TabBar theming

Also, I notice that in the translation file common words like 'OK ' and 'Cancel' are listed, when these are obviously defined in CtrlLib. Is it not possible to exclude already translated words?

James

Subject: Re: Docking / DockTest packages
Posted by mirek on Tue, 18 Mar 2008 21:05:50 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 18 March 2008 11:32
Also, I notice that in the translation file common words like 'OK ' and 'Cancel' are listed, when these are obviously defined in CtrlLib. Is it not possible to exclude already translated words?
James

Frankly, not sure about this

I see this as little problematic: Some words can be in CtrlLib now, but then be removed.

So I think each package should carry on its translations...

Mirek

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Tue, 18 Mar 2008 21:24:30 GMT
View Forum Message <> Reply to Message

I would like to integrate docking into an app both to test the packages in the long run and to see if docking really gives an improvement in the usability of it. I'll try to integrate it tomorrow.

I'll start with your version, and then try Oblivion's. They both seem to offer a common general feature set? Does your version still include somewhere support for visual aids? I find it very hard to position windows without them in general, not just in your ctrls). Also, where can I get the sources? I don't know the secret svn IP address.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Tue, 18 Mar 2008 22:10:28 GMT
View Forum Message <> Reply to Message

There is no secret SVN address Just go to the SourceForge page and choose SVN from the menu. It gives the necessary command there, although you will have to install Subversion. DockingExample1 & 2 under bazaar are what you are looking for once you've checked it out.

The two versions are similar (we share the same vision, but differ in style), though this version is

further along in development. The visual aids are currently unsupported as I keep on changing things and not propagating the changes, but they will be shortly since you like them.

It is certainly not finished yet, so please share you're thoughts.

Subject: Re: Docking / DockTest packages
Posted by mr_ped on Wed, 19 Mar 2008 08:25:42 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 18 March 2008 23:10although you will have to install Subversion.

Well, you need SVN client, not full SVN itself.
On Windows I can recommend TortoiseSVN for it's rich set of supported SVN functions and generally good stability (although many changelogs from new versions made me shiver with "how was it working before those bug fixes?", but almost never ran into actual issue).
On linux I have still no favorite, actually anything I tried is lacking something, kdesvn works for me in many cases ok, but far from complete solution like Tortoise is.

Subject: Re: Docking / DockTest packages
Posted by cbpporter on Wed, 19 Mar 2008 08:47:14 GMT
View Forum Message <> Reply to Message

Thanks for the info. I know how to use SVN and I have Tortoise (even though it doesn't give me such a big advantage since I use TotalCmd).

I downloaded SVN version, but unfortunately I managed to loose some modifications to U++ source code. Have to reproduce them and this time isolate them properly.

Subject: Re: Docking / DockTest packages
Posted by mdelfede on Wed, 19 Mar 2008 13:12:59 GMT
View Forum Message <> Reply to Message

IMO on linux the best is svn command line + meld for diff.... And, if you don't want to loose some local changes, keep a svn folder separate from you devel, and sync when you need with meld.

Max

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Sat, 22 Mar 2008 12:25:59 GMT
View Forum Message <> Reply to Message

Hi Mrjt,

Here is the new docking.t with Turkish translation.

ps: I had to change the char. encoding to UTF8.

Regards.

File Attachments

1) Docking.t, downloaded 619 times

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Thu, 27 Mar 2008 19:25:57 GMT
View Forum Message <> Reply to Message

Hi!

I finally got to the point where I can use your tabbing system. It was not that easy, because if you try to design the "perfect" interface without taking into consideration dockable windows, you can hardly insert this concept in the interface so that it feels natural.

But I found a way to improve my interface, but it seems that I have special needs . I wonder if you system could be adapted so that a layout is formed only by docked windows, without a main layout? First docked window takes full surface, second takes half and there is only one splitter. It would be nice if certain windows can be configured to tile vertically, and others horizontally. I believe this is some work and I'm not asking you to implement this for me. It is just an idea of what I would like to have, and I'm certain I can get it to work after some hacking by inserting a dummy header into the main layout, and an option to dynamically switch the context of it. But this is something for the future.

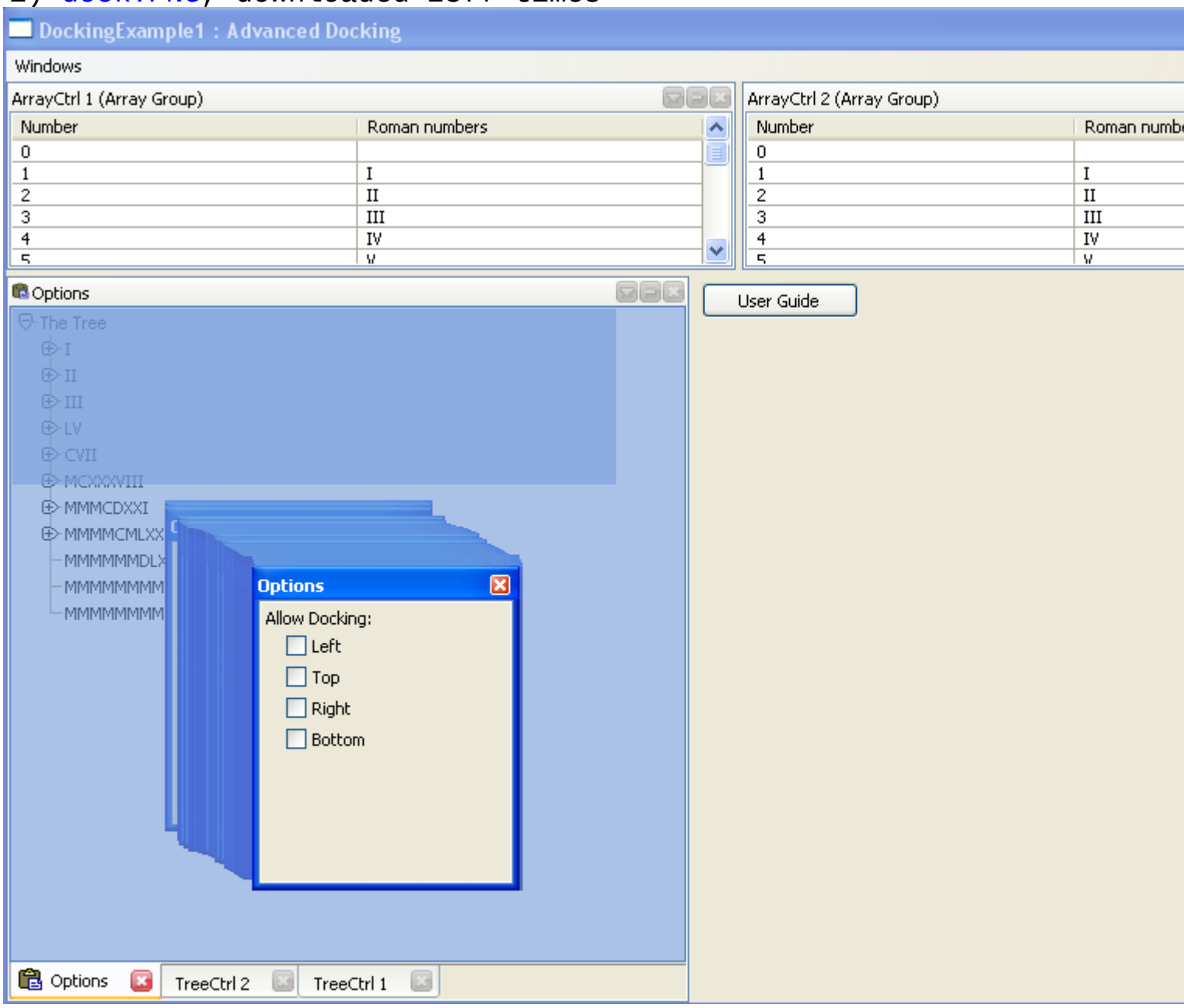
Something that is for now is a bug. I found that if you slowly drag a dock over a blueish highlighted surface you can leave a trail. To reproduce, open your test example, dock any window on the left for example. Take a second floating window, and drag it over the first one which is docked now. A blue highlight will appear. Move it around until the highlight covers the whole window (not just top or bottom). After the highlight appears, without releasing the mouse, move the window arround and you will see a trail.

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Thu, 27 Mar 2008 19:28:58 GMT
View Forum Message <> Reply to Message

I just reread my post and the direction are not as clear as they could have been. Here is a picture to help:

File Attachments

1) dock.PNG, downloaded 2377 times



Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 28 Mar 2008 13:40:05 GMT
View Forum Message <> Reply to Message

I've just uploaded a new version to the repository:
-Animation and auto resize now work perfectly, I still need to add some more intelligence to the

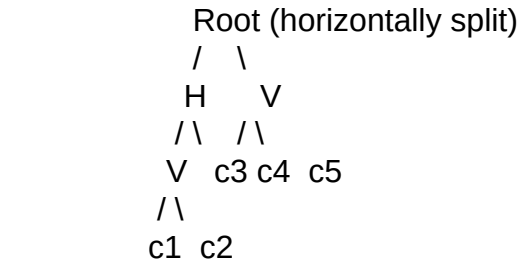
resize algorithm but for now it works very well (no more problems with not having enough space).
-Pop-up docking hints are supported again (and should be forever now). Just inherit from PopUpDockWindow instead of DockWindow.
-Some other minor stuff.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 28 Mar 2008 13:45:19 GMT
View Forum Message <> Reply to Message

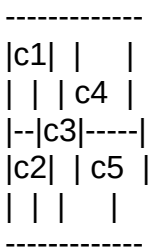
Something that is for now is a bug.
Thanks, I was trying to do it with a transparent Ctrl, which doesn't work very well in Upp . Fixed now.

But I found a way to improve my interface, but it seems that I have special needs Smile.
If I understand correctly this actually doesn't sound too difficult to do, but is quite different to how the docking works at the moment. What you actually need is a single ctrl like a tree of splitters:

Internal structure:
H- horizontal splitter node
V- vertical splitter node
cN- ctrls (leaves)



Appearance:



and then add the abitility to drag-drop the ctrls around. It's an interesting concept, if you can make it work right. I've already got a SplitterTree ctrl so I might have a quick play with it. You can also just implement this by adding splitters to splitters to form the tree, but this gets unwieldy quite quickly.

James

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 28 Mar 2008 16:07:35 GMT
View Forum Message <> Reply to Message

Well, that was easy. Quite a bit of deleting, about 20 lines of code and we have a new interface.

It's a modified version of the Docking package, almost all of the features work except for Serialization/Animation. These are essentially the same problem, and because the tree structure is more complex than a normal splitter it's more difficult to do.

The example attached use the popup interface, although the drag-drop one also works.

Is this something like you were suggesting? I thinks it's quite cool in any case

File Attachments
1) Example exe.zip, downloaded 656 times

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 28 Mar 2008 16:08:50 GMT
View Forum Message <> Reply to Message

Source code

File Attachments
1) Example Source.zip, downloaded 630 times

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Fri, 28 Mar 2008 19:55:43 GMT
View Forum Message <> Reply to Message

Quote:Well, that was easy. Quite a bit of deleting, about 20 lines of code and we have a new interface.
Wow! That even sounds easy! I was aware of an approach with multiple splitters, but before I would have produced something with the fraction of the functionality of you framework it would have certainly taken me quite some time.

And I really like it. It was exactly what I was hopping for. You see, I have o TotalCmd like interface, with the left side different from the right side. Each side has for choseable layouts. With the previous version of the docking package I would have had to select one of them and make them the main layout and second with a dock and the rest of them as pre-hidden ones. This is all OK, but the design goals was to have both windows on the screen at the same time and they both have the same importance.

But with this new layout scheme, I can definitely get the ultimate (no pun intended) end user configuration possibility. I certainly think this has nice potential, and could possibly be used to emulate MDI/split screen document structure too.

And now for the bugs:
1. If you undock all windows, none can be docked again.
2. The ability to choose from the interface a position for docking is gone. As far as I can tell, this was dropped because position are no longer that straightforward. But still, a nice left and right at left could be achievable.
3. If I put a window to autohide, then I wait for it to appear and click on it's option button, a menu appears, but the window will automatically hide again. It should stay until the menu is closed.
4. Windows set to autohide can not be dragged away anymore.
5. Let an autohide open and quickly resize the window. You will see the pop-up borderless window stay in place.

Items 3 (only noticed for top position autohide docks) and 5 are also in the old code.

Anyway, thank you for giving my idea a shot .

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 31 Mar 2008 10:32:59 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 28 March 2008 19:55But with this new layout scheme, I can definitely get the ultimate (no pun intended) end user configuration possibility. I certainly think this has nice potential, and could possibly be used to emulate MDI/split screen document structure too.

And now for the bugs:
1. If you undock all windows, none can be docked again.
2. The ability to choose from the interface a position for docking is gone. As far as I can tell, this was dropped because position are no longer that straightforward. But still, a nice left and right at left could be achievable.
3. If I put a window to autohide, then I wait for it to appear and click on it's option button, a menu appears, but the window will automatically hide again. It should stay until the menu is closed.
4. Windows set to autohide can not be dragged away anymore.
5. Let an autohide open and quickly resize the window. You will see the pop-up borderless window stay in place.

Items 3 (only noticed for top position autohide docks) and 5 are also in the old code.

Anyway, thank you for giving my idea a shot .

No problem. As I said it wasn't difficult to do and I agree that it's a nice interface for certain apps.

Item 4 on your list I haven't been able to reproduce. I've tested it on XP, Gnome and KDE and that portion of the code is identical to the original package (and hasn't changed for some time). Can you give more details? All the other items are fixable, though 3 is going to be difficult. The problem

is that I'm not sure what to do with the code now.

If you'd like to use it I can fix 1, 2 & 4 and add serialization by the end of the week, which makes it useable in a real-world app. But unless there is interest from other parties I can't really justify maintaining it and adding Animation, which is a real pain. Unfortunately although 90% of the code is the same as the original docking package it's different enough that I can't see how to merge the two.

So let me know if you'd like me to make it useable and I'll have to think about what to do about maintaining it.

James

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Mon, 31 Mar 2008 13:22:46 GMT
View Forum Message <> Reply to Message

Quote:If you'd like to use it I can fix 1, 2 & 4 and add serialization by the end of the week, which makes it useable in a real-world app. But unless there is interest from other parties I can't really justify maintaining it and adding Animation, which is a real pain. Unfortunately although 90% of the code is the same as the original docking package it's different enough that I can't see how to merge the two.

So let me know if you'd like me to make it useable and I'll have to think about what to do about maintaining it.

James
I don't know how to proceed. Give a day or tho to look over the code, and if I can make head or tails out of it, maybe I can strip out all that I don't need and provide a package which I maintain until it reaches maturity (read no obvious bugs). But I can't promise anything, since there is so much code and so little time.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 31 Mar 2008 14:47:13 GMT
View Forum Message <> Reply to Message

Good luck That sounds like a reasonable plan, let me know if you need any help. The code can be slighly unruly in places but there is a structure and design to it (mainly an attempt to create a small public interface and maintain loose-coupling between the internal classes). The DockCont class is the worst bit but that handles the tabbing/nesting and you shouldn't have to worry about it.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 09 Apr 2008 14:34:17 GMT

I've updated the Docking package in the SVN.

SVN Revision 211 Changes:
- Major improvement to the Smart-Resize algorithm. Works very well now.
- Additional window animation (floating windows now 'slide' into place when docked).
- Improved group tree in config window
- Some internal refactoring and various little fixes. No major bugs (none found).

James

File Attachments

1) Docking Package rv.111.zip, downloaded 604 times

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Mon, 14 Apr 2008 22:21:53 GMT

mrjt wrote on Mon, 31 March 2008 17:47Good luck That sounds like a reasonable plan, let me know if you need any help. The code can be slighly unruly in places but there is a structure and design to it (mainly an attempt to create a small public interface and maintain loose-coupling between the internal classes). The DockCont class is the worst bit but that handles the tabbing/nesting and you shouldn't have to worry about it.
Wow, that took me a lot longer than expected! I have now a basic grasp of the inner workings of your code. This is good, but right now I don't know what to do with it. The next update of my app got published with the same TC like interface, and so far there were no complains regarding it. So I don't know if maintaining a fork of your docking system would be a wise decision, especially since the next update is planned for August. But I would like to investigate a way to make any control pop-upable (is that a word) with minimal impact on core packages.

But I really love your tree code. I wonder what the impact of a college course (if there are still C++ courses, with all the Java ones)which teaches such tree data would have. Unfortunately, here in my local city at it's relatively high regarded Computer Science course, tree structures (and other data structures) as well are thought with nasty plain C style. This is fine for an introductory level, but taking into account the shear number of hours dedicated to the subject, some higher level stuff would be welcome. And I'm sure that 75% of students would break their teeth on that code .

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 15 Apr 2008 08:56:13 GMT

cbpporter wrote on Mon, 14 April 2008 23:21So I don't know if maintaining a fork of your docking system would be a wise decision
Cool. I'd like to come back to this later though, perhaps after I've finished with the main package and can think of integrating them somehow. I think it's an interesting paradigm that I haven't seen

anywhere else so it's a fun experiment.

Quote:But I would like to investigate a way to make any control pop-upable (is that a word Smile) with minimal impact on core packages.
I'm not quite sure what you mean (presumably not just calling Popup on a ctrl), but if you can give me a more detailed description I'm sure I can help. I consider myself a bit of an expert of that subject now

Quote:But I really love your tree code. I wonder what the impact of a college course (if there are still C++ courses, with all the Java ones)which teaches such tree data would have. Unfortunately, here in my local city at it's relatively high regarded Computer Science course, tree structures (and other data structures) as well are thought with nasty plain C style. This is fine for an introductory level, but taking into account the shear number of hours dedicated to the subject, some higher level stuff would be welcome. And I'm sure that 75% of students would break their teeth on that code Twisted Evil.
No doubt I'm still not sure myself whether the Tree class was a stroke of brilliance or absolute madness. And I couldn't write code like that before encountering Upp and reading lots of Mirek's code, but I did at least get taught lots of C++ data structures on my CS course (and Computational Geometry).

Some/all of the traverses don't work correctly I'm afraid, I didn't notice until I tried to do serialization and I haven't got round to fixing them yet.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 23 May 2008 16:21:18 GMT
View Forum Message <> Reply to Message

Just a quick post so that it doesn't look like the package is dead really. I haven't been posting updates since there aren't really any new features, just incremental improvements.

X11 support has been improved (including performance).
TabBar theming has been much improved thanks to unodgs.
API documentation is now included.
Many bugs squashed.

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Fri, 23 May 2008 23:05:45 GMT
View Forum Message <> Reply to Message

Quote:
Just a quick post so that it doesn't look like the package is dead really. I haven't been posting updates since there aren't really any new features, just incremental improvements.

X11 support has been improved (including performance).
TabBar theming has been much improved thanks to unodgs.
API documentation is now included.
Many bugs squashed.

Well, mrjt congratulations. It seems that you have won the "competition" (If there was one) I was away for a long while and I had to pause the development of my DockCtrl package, since I was very busy with my other projects and my ph.d. I'm sorry for that, but I'm glad to see that, at least it encouraged you to finalize your solution (or I hope so) And one more thing, if you would like to use my packages control panel style, etc. and "DockCtrl" label for your docking package, please do it.

Well, for me, probably, I'll come up with something totally new (a new ctrl maybe) and/or some additons/ideas to your package (if you ask for any) this summer

And I would like to thank the U++ community for their support to the DockCtrl package.

Regards.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Sat, 24 May 2008 16:49:52 GMT
View Forum Message <> Reply to Message

Well, perhaps a bit of friendly competion, it's good for motivation But I definitely wouldn't have got it working this well without your ideas and designs, so thanks!

And if you have any more of those good ideas please let me know, or try them out yourself . Otherwise, if you're looking for a project I believe some people are desiring a port of WebKit, and I'd happily collaborate on it if you wanted help.

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Sat, 24 May 2008 19:22:09 GMT
View Forum Message <> Reply to Message

Quote:
And if you have any more of those good ideas please let me know, or try them out yourself.

Actually I have an idea.
As you know, I had planned to add a dockable bar control to the DockCtrl framework. Now that

the development of DockCtrl is stopped, I would like to share what I have in my mind. I have studied your framework and I've come to the the conclusion that there is no reason that it can't be done (since, it's similar to mine). Here is the idea: A "dockable" DockBar . What I mean is, a dockbar which will have both user defined standard buttons (as usual) and dockwindows which can be "docked" to the dockbar as buttons. This is both theoretically and pratically possible. Since the dockwindows have both a label and an icon, we can use them as buttons (or, we can supply a default icon if they don't have one) and by using a simple overlay icon (something like on Windows desktop shortcut) on the dockwindow-buttons, user can easily seperate between standard button and a dockwindow-button. As for the behaviour of this dockwindow-buttons, when the user presses the the button the dockwindow can pop-up (just like the autohidebar behaviour) and when the user drags the buttoned dockwindow, it can float in its standard dockwindow form. I know that we already have AutoHideBar for similar behaviour, but I think this will save a lot of space in professional projecs. Think about having, say, a dockable "virtual functions" or other windows in TheIDE which can be buttonized and docked. Now, I didn't see something like this in any other docking framework (porbably this is going to be a unique feature). Imho, it would be both useful and eye-candy. It shoudn't be so hard to implement this (actually, I had started to implement this, but then abandoned due to the reasons I've mentioned in my last post). Since, I think we both have learned a hell lot about docking mechanisms and U++ behaviour throughout our "competition" we can add this feature easily. If it seems reasonable to you, we can collaborate on it or you can do it all by yourself. What do you think about it?

Quote:
Otherwise, if you're looking for a project I believe some people are desiring a port of WebKit, and I'd happily collaborate on it if you wanted help.

Yes, I myself was considering that since Mirek has suggested a plugin/port for the webkit. I've examined webkit and now I'm examining the WxWidgets port of Webkit. Their solution (and interface) seems good. I can plunge into this "adventure" by myself, but probably it would be better to do it as a "party". Well, after June 3, I'm free. And, probably "I'm in!".

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 09 Jun 2008 11:22:20 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Sat, 24 May 2008 20:22
Actually I have an idea.

That does sound very interesting! As you say it would duplicate some of the features of Autohide so I would probably not want both. I'll give it a go when I get the chance (if you wish to try it should be quite straightforward to test, just replace the autohide bars with the new Dockbar. The other changes should be small).

And for the WebKit port have a look and see what needs doing, then perhaps start a Bazaar

thread and a repository. If you don't mind being in charge of the overview I can happily work on individual sections and perhaps others will to.

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Sun, 22 Jun 2008 11:04:03 GMT
View Forum Message <> Reply to Message

Quote:
And for the WebKit port have a look and see what needs doing, then perhaps start a Bazaar thread and a repository. If you don't mind being in charge of the overview I can happily work on individual sections and perhaps others will to.

Yep, that I will do. And nope, I won't mind being in charge.
See: http://www.ultimatepp.org/forum/index.php?t=msg&goto=165 22&#msg_16522

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 01 Sep 2008 15:21:40 GMT
View Forum Message <> Reply to Message

I've just updated the SVN, however, I'm no longer 100% sure whether I'm using the correct repository. Is https://upp.svn.sourceforge.net/svnroot/upp/trunk still correct?

Various bug fixes + some new features:
- Greatly improved animation performance (I think this was caused by an image resizing bug in chameleon that has since been fixed, but I now cache the image regardless).
- 'Activate' functions that bring a window to the foreground if it's hidden or open it if it's closed
- Layout 'locking'. When the locked all frames and buttons (except for tabs) are hidden and the user cannot modify the windows layout.
- some addition customisation options.

James

Subject: Re: Docking package (plus examples)
Posted by mirek on Mon, 01 Sep 2008 16:31:01 GMT
View Forum Message <> Reply to Message

mrjt wrote on Mon, 01 September 2008 11:21I've just updated the SVN, however, I'm no longer 100% sure whether I'm using the correct repository. Is https://upp.svn.sourceforge.net/svnroot/upp/trunk still correct?

Incorrect.

Please read PM from Thu, 14 August 2008 17:49.

Mirek

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 02 Sep 2008 17:17:49 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 01 September 2008 17:31mrjt wrote on Mon, 01 September 2008 11:21I've just updated the SVN, however, I'm no longer 100% sure whether I'm using the correct repository. Is https://upp.svn.sourceforge.net/svnroot/upp/trunk still correct?

Incorrect.

Please read PM from Thu, 14 August 2008 17:49.

Mirek
Thanks. Not sure how I missed that, but the update is now committed to correct one.

Subject: Re: Docking / DockTest packages
Posted by mirek on Wed, 03 Sep 2008 08:21:29 GMT
View Forum Message <> Reply to Message

Well, I think this is almost ready to be moved to uppsrc.

Just small comments:

I really like the code, the only thing that makes me a little bit nervous is "unstable placement" of '*' and '&'.

This is not really critical. Anyway, MY coding style mandates to place '&' alongside type and '*' with variable, I believe that makes it visually more readable. I have no problem if you have different style, it is just funny that you seem to mix several styles in single package

Also, with class definition, I put '{' on the same line.

Now important: If your preferences are different, please keep them! Over years, I have found this can have some advantage too, as just seeing the code I can say how is responsible (the style forms sort of fingerprint

A little bit more important: Maybe we should discuss the button sizes. So far, I tend to prefer 64x24 - while not correct on any platform, looks almost correct everywhere.

Mirek

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 03 Sep 2008 09:10:45 GMT
View Forum Message <> Reply to Message

Well, the style problems are just because I'm inconsistent I've gradually been adopting Upp style since I started using it but I'd honestly never conciously noticed the different placement of '*' and '&' and I have no problem fixing it. Likewise with button sizes.

I'll let you know when I've fixed it.

Subject: Re: Docking / DockTest packages
Posted by unodgs on Wed, 03 Sep 2008 11:14:54 GMT
View Forum Message <> Reply to Message

luzr wrote on Wed, 03 September 2008 04:21Well, I think this is almost ready to be moved to uppsrc
We have a nv branch of theide which uses Docking package. I think it'd be better to work on it a little bit more before moving. I'd like to finish tabbar as well. It's all connected.

Subject: Re: Docking / DockTest packages
Posted by mrjt on Wed, 03 Sep 2008 14:47:28 GMT
View Forum Message <> Reply to Message

I agree, there is also one major addition I need to make which may introduce some new bugs.

Subject: Re: Docking / DockTest packages
Posted by mirek on Wed, 03 Sep 2008 21:05:51 GMT
View Forum Message <> Reply to Message

OK, but let us do it soon.

IMO, there is no code that would be broken by adding this package to uppsrc. So the only real reason not to do so would be if you feel that the interface design is faulty.

Mirek

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Fri, 19 Sep 2008 12:32:32 GMT

View Forum Message <> Reply to Message

Hi Mrjt,

I have noticed something about the "autohide" bar. When a dock goes into autohide state, it's tab (or the last activated one's tab) stays active even if the mouse is not hovering on it. Is this behaviour intentional, or is it just a small bug?

Regards.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 19 Sep 2008 14:29:13 GMT
View Forum Message <> Reply to Message

Bug

Subject: Re: Docking package (plus examples)
Posted by Novo on Sun, 09 Nov 2008 20:08:56 GMT
View Forum Message <> Reply to Message

I couldn't build Docking package from trunk (http://upp-mirror.googlecode.com/svn/trunk).

c:\local\home\sergey\cpp\code\gui\upp\svn\bazaar\Docking\DockCont.cpp(766) : error C2248: 'Upp::CtrlFrame::CtrlFrame' : cannot access private member declared in class 'Upp::CtrlFrame'
c:\local\home\sergey\cpp\code\gui\upp\svn\uppsrc\CtrlCore\CtrlCore.h(75) : see declaration of 'Upp::CtrlFrame::CtrlFrame'
c:\local\home\sergey\cpp\code\gui\upp\svn\uppsrc\CtrlCore\CtrlCore.h(57) : see declaration of 'Upp::CtrlFrame'

Am I doing something wrong? As far as I understand, there is an ongoing development in a branch. If this is correct, could you please promote changes from a branch to the trunk?

TIA

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 10 Nov 2008 10:56:04 GMT
View Forum Message <> Reply to Message

Novo wrote on Sun, 09 November 2008 20:08I couldn't build Docking package from trunk

(http://upp-mirror.googlecode.com/svn/trunk).

c:\local\home\sergey\cpp\code\gui\upp\svn\bazaar\Docking\DockCont.cpp(766) : error C2248: 'Upp::CtrlFrame::CtrlFrame' : cannot access private member declared in class 'Upp::CtrlFrame'
c:\local\home\sergey\cpp\code\gui\upp\svn\uppsrc\CtrlCore\CtrlCore.h(75) : see declaration of 'Upp::CtrlFrame::CtrlFrame'
c:\local\home\sergey\cpp\code\gui\upp\svn\uppsrc\CtrlCore\CtrlCore.h(57) : see declaration of 'Upp::CtrlFrame'

Am I doing something wrong? As far as I understand, there is an ongoing development in a branch. If this is correct, could you please promote changes from a branch to the trunk?

TIA

That odd. I've just compiled it against the svn and 2008.1 versions using MSC8 and MingW without any problems. I can see a potential problem on line 766, but it doesn't match the error message. Are you using GCC on linux by any chance? If so which vesion?

I've changed the offending line but I can't guarantee it'll fix the problem since I can't reproduce it. It's in revision 628.

All updates go into the main trunk, the version you were using was the most recent.

Hope that helps. If you intend on using the Docking package I would love to hear feedback so that I can improve it further, particularly on how easy the API is to use.

James

Subject: Re: Docking package (plus examples)
Posted by Novo on Tue, 11 Nov 2008 05:11:45 GMT
View Forum Message <> Reply to Message

mrjt wrote on Mon, 10 November 2008 05:56
That odd. I've just compiled it against the svn and 2008.1 versions using MSC8 and MingW without any problems. I can see a potential problem on line 766, but it doesn't match the error message. Are you using GCC on linux by any chance? If so which vesion?

I've changed the offending line but I can't guarantee it'll fix the problem since I can't reproduce it. It's in revision 628.

All updates go into the main trunk, the version you were using was the most recent.

Hope that helps. If you intend on using the Docking package I would love to hear feedback so that I can improve it further, particularly on how easy the API is to use.

James

Thanks. Your change fixed the compilation problem.

Sorry for forgetting to report compiler name and version. That was MSVC9.

I've checked compilation with GCC 4.1.3 on Ubuntu 7.10 + GTK. No problem with compilation. Unfortunately, there is a problem with DockingExample1. Basically, docking is not working with this example. When I'm trying to drag any control (TreeCtrl 2, for example), several other controls just disappear. Sometimes they disappear all, sometimes one or two of them stay visible. That seems to depend on dragging direction.

And that is even more strange because DockingExample2 seems to work correctly.

Another problem. Docking is not working on Vista compiled with MSVC9 + disabled "Show window contents while dragging".
It is not possible to dock a control using mouse.

I've also checked docking on Windows XP + MSVC8 + enabled "Show window contents while dragging". Everything was working fine.

Sorry for making troubles

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 11 Nov 2008 09:52:49 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 11 November 2008 05:11
Sorry for making troubles

No problem, I did say I wanted feedback after all!

Novo wrote on Tue, 11 November 2008 05:11
Unfortunately, there is a problem with DockingExample1. Basically, docking is not working with this example.Thanks, I haven't tested DockingExample1 for a while so I didn't notice. I'm currently re-installing everything on my machine but I'll have a look at it as soon as I'm able.

Novo wrote on Tue, 11 November 2008 05:11Another problem. Docking is not working on Vista compiled with MSVC9 + disabled "Show window contents while dragging".
It is not possible to dock a control using mouse.
Bugger. You may be the first person to test this with MSC9, although I'll be installing it shortly.

The dragging problem is a real pain in the arse! The dragging works by tracking window movement, so Vista must not be sending the usual WM_MOVE/WM_SIZE messages when the contents aren't being drawn. If Vista only sends move events to the window after it has completed

the move then I can make it dock, but the animation/highlighting will never work .

Subject: Re: Docking package (plus examples)
Posted by Novo on Thu, 13 Nov 2008 03:42:47 GMT
View Forum Message <> Reply to Message

I've found two more problems with DockingExample1. I was testing two configurations:

1) Vista + disabled "Show window contents while dragging" + MSVC71;
2) The same as above, but compiled with MSVC9;

In case 2 dragging of TreeCtrl 2 causes disappearing of ArrayCtrl 1 and TreeCtrl 1. This is similar to what happens on Linux + GCC 4.1.3.

In both case 1 and case 2 docking and undocking of "single" controls using mouse causes they to shrink in size. Try to dock-undock ArrayCtrl 2 or TreeCtrl 2 several times, and see what happens.

Subject: Re: Docking package (plus examples)
Posted by Novo on Sat, 22 Nov 2008 17:34:49 GMT
View Forum Message <> Reply to Message

I checked DockCtrl (by Ismail Yilmaz) + Vista + disabled "Show window contents while dragging" + MSVC71/MSVC9.

Docking with mouse is working fine. I couldn't find any difference between MSVC71 and MSVC9.

There are currently only two problems with DockCtrl:

1) DockCtrlExample crashes during docking a control as a tab;
2) Closing a tab (by pressing a close/cross button) causes assert in DockCtrlExample;

Problem (2) happens in Docking package too. But in this case application just crashes.

I wonder if it is possible to make at least one stable docking control/package?

Sorry for being annoying.

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Sun, 23 Nov 2008 21:38:38 GMT
View Forum Message <> Reply to Message

Novo wrote on Sat, 22 November 2008 19:34I checked DockCtrl (by Ismail Yilmaz) + Vista + disabled "Show window contents while dragging" + MSVC71/MSVC9.

Docking with mouse is working fine. I couldn't find any difference between MSVC71 and MSVC9.

There are currently only two problems with DockCtrl:

1) DockCtrlExample crashes during docking a control as a tab;
2) Closing a tab (by pressing a close/cross button) causes assert in DockCtrlExample;

Problem (2) happens in Docking package too. But in this case application just crashes.

I wonder if it is possible to make at least one stable docking control/package?

Sorry for being annoying.

Hello Novo,

Unfortunately, the development of the DockCtrl is stopped (Recently I have marked it as "Depreceated") since I was very occupied with other projects. I'm sorry for any inconvenience caused with the package. If possible, could you give some details about the assert?

Regards.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 24 Nov 2008 12:27:09 GMT
View Forum Message <> Reply to Message

Sorry for the delay.

I think I've fixed the X11 issues. it was an uninitialised variable, as usual. I missed this only testing with example 2 because it serializes and initialises the variable.

I haven't looked at the Vista 'no contents' problem, this will probably take some time to fix and I'm pretty busy at the mo. It should be noted that this problem will occur anytime the WM behaviour is not what I'm expecting (some Linux WMs do this too) and each instance just has to be fixed on it's own. I don't think there is a universal solution.

I cannot reproduce any problems with closing tabs (after the X11 fix), could you provide more information?

Novo
I wonder if it is possible to make at least one stable docking control/package?
Yes. The problems you've encountered are the result of:
1- Not enough prople having used it to report all of the bugs yet
2- System specific support being required for certain OSs
Both of these will be fixed with time and your feedback will help, response time is just a little slow at the moment because I'm very busy.

Quote:Sorry for being annoying.
Nonsense Thank you for the feedback!

James

Subject: Re: Docking package (plus examples)
Posted by mirek on Mon, 24 Nov 2008 17:14:19 GMT
View Forum Message <> Reply to Message

mrjt wrote on Mon, 24 November 2008 07:27
1- Not enough prople having used it to report all of the bugs yet

That can be improved by moving it to uppsrc...

Mirek

Subject: Re: Docking package (plus examples)
Posted by Novo on Mon, 24 Nov 2008 17:48:47 GMT
View Forum Message <> Reply to Message

Thanks.

mrjt wrote on Mon, 24 November 2008 07:27
I cannot reproduce any problems with closing tabs (after the X11 fix), could you provide more information?

James

I checked in Windows XP Professional, no administrative privileges, svn rev. 671, MSVC 71. Closing tabs crashes in Debug configuration in DockCont.cpp line 383 with memory access violation. This is a call to Layout(). TheIDE debugger is not very helpful in this situation. And the Visual Studio itself cannot open executable.

There is no crash in Optimal configuration. In this case controls just get hidden.

Behavior is different when DockingExample1 runs under TheIDE debugger or standalone. Under debugger ArrayCtrl1 and TreeCtrl1 disappear when I start dragging TreeCtrl2. Looks like uninitialized variable to me.

I ran DockingExample1 under Purify and got only a report about two memory leaks and nothing about reading uninitialized memory. That is strange.

Subject: Re: Docking package (plus examples)
Posted by unodgs on Mon, 24 Nov 2008 18:13:26 GMT
View Forum Message <> Reply to Message

luzr wrote on Mon, 24 November 2008 12:14mrjt wrote on Mon, 24 November 2008 07:27
1- Not enough prople having used it to report all of the bugs yet

That can be improved by moving it to uppsrc...

Mirek

I already moved it in my local repo. Maybe I'll commit it. I waited for complete integration of it with ide (50%) but recently I was simply lack of time. So I'll move it there now if noone raises an objection

Subject: Re: Docking package (plus examples)
Posted by Novo on Mon, 24 Nov 2008 19:20:02 GMT
View Forum Message <> Reply to Message

unodgs wrote on Mon, 24 November 2008 13:13So I'll move it there now if noone raises an objection

IMHO Docking package is not usable at the moment.

Subject: Re: Docking package (plus examples)
Posted by Novo on Tue, 25 Nov 2008 05:00:07 GMT
View Forum Message <> Reply to Message

Oblivion wrote on Sun, 23 November 2008 16:38If possible, could you give some details about the assert?

Hi Oblivion,

DockCtrlExample + Vista + disabled "Show window contents while dragging" + MSVC71/MSVC9 + SVN rev. 671 + Debug configuration, of course.

Just launch DockCtrlExample and try to close "Dock 4".

DockTabs.cpp line 436. This is a call to WhenActive() in method TabInterface::LeftDown(). Assert happens in Vector::Get(int i). Variable "active" is zero.

I'm also getting memory access violation when trying to dock "Dock 5" control into "Dock 2"

control. It happens in Ctrl.cpp line 258, method Ctrl::IsVisible(). It is called from CtrlDraw.cpp line 43 (Ctrl::Refresh()), which is called from DockableCtrl.cpp line 192, DockableCtrl::WindowProc().

I hope that helps.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 25 Nov 2008 12:39:36 GMT
View Forum Message <> Reply to Message

NovoI checked in Windows XP Professional, no administrative privileges, svn rev. 671, MSVC 71. Closing tabs crashes in Debug configuration in DockCont.cpp line 383 with memory access violation. This is a call to Layout(). TheIDE debugger is not very helpful in this situation. And the Visual Studio itself cannot open executable.

There is no crash in Optimal configuration. In this case controls just get hidden.

Thanks, I've found and fixed the problem. It was introduced with a recent change, and a ctrl was deleting itself by accident. It was the change I made after saying:
Novo there is also one major addition I need to make which may introduce some new bugs.So I'm not that suprised

NovoLooks like uninitialized variable to me.
I didn't get around to updating the svn yesterday, so I would guess that this is the problem I fixed.

NovoIMHO Docking package is not usable at the moment.I've actually been using an older stable version of it for a few months without issues. I do have the advantage of a fixed target platform however.

I've now updated the SVN (rev. 672) with both fixes and changed the code style to better match uppsrc. I don't plan on making any major additions to the code now (unless someone comes up with some amazing new features), so I have no objections to moving it to uppsrc. I'll update and improve the documentation when I get the chance, perhaps later this week.

Subject: Re: Docking package (plus examples)
Posted by Novo on Tue, 25 Nov 2008 17:09:20 GMT
View Forum Message <> Reply to Message

Hi mrjt,

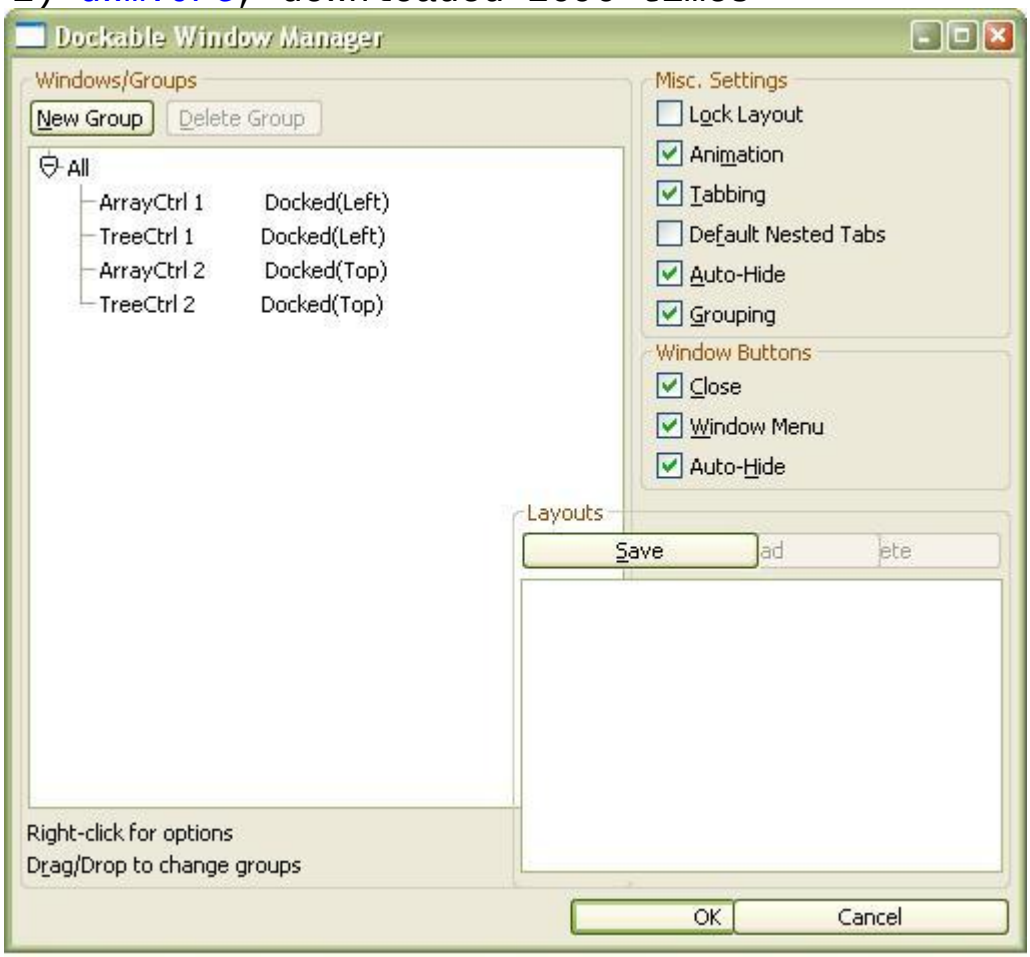
Thanks for taking care of the problems!

Docking package got much more stable, although there is a couple of minor issues.

1) Undocking-docking of a single control still shrinks its size. To reproduce that try to undock-dock

File Attachments

1) dwm.JPG, downloaded 2696 times



Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 25 Nov 2008 17:52:35 GMT
View Forum Message <> Reply to Message

Thanks.

I've fixed the layout issue.

The size issue is more complicated, it a combinatiojn of two factors:
- This is actually deliberate. The docking code is attempting to keep all of the other ctrls that are

docked in the same pane above their minimum sizes. The newly docked ctrl is guaranteed to get it's minimum size but anything more than that is dependent on how much spare space there is.
If you make the window larger (I've updated DockingExample1 so that you can now do this) you should see the ctrl getting more space.
There is still rom for improvement in the sizing algorithms, if you have any ideas please let me know
- There was a minor bug that may have prevented the ctrl using as much size as it was supposed to, but in this case I don't think it was the limiting factor

New update is revision 674

James

Subject: Re: Docking package (plus examples)
Posted by Novo on Wed, 26 Nov 2008 17:04:59 GMT
View Forum Message <> Reply to Message

Hi James,

mrjt wrote on Tue, 25 November 2008 12:52
I've fixed the layout issue.

James

Thanks a lot!

I checked Docking package with disabled "Show window contents while dragging" in both Windows XP and Vista. Docking is still not working in both these configurations. In opposite, DockCtrlExample (by Ismail Yilmaz) is working correctly in these configurations. I suspect Ismail

Thanks.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Thu, 27 Nov 2008 09:28:55 GMT
View Forum Message <> Reply to Message

Novo wrote on Wed, 26 November 2008 17:04Hi James,

mrjt wrote on Tue, 25 November 2008 12:52
I've fixed the layout issue.

James

Thanks a lot!

I checked Docking package with disabled "Show window contents while dragging" in both Windows XP and Vista. Docking is still not working in both these configurations. In opposite, DockCtrlExample (by Ismail Yilmaz) is working correctly in these configurations. I suspect Ismail

Thanks.

I didn't realise you could do it in XP as well, I would have fixed it sooner

Turned out I just needed to change the event I was listening for to WM_MOVING. The fix is in revision 678.

Thanks for all your testing!

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Thu, 27 Nov 2008 19:46:08 GMT
View Forum Message <> Reply to Message

Novo wrote on Wed, 26 November 2008 19:04Hi James,

mrjt wrote on Tue, 25 November 2008 12:52
I've fixed the layout issue.

James

Thanks a lot!

I checked Docking package with disabled "Show window contents while dragging" in both Windows XP and Vista. Docking is still not working in both these configurations. In opposite, DockCtrlExample (by Ismail Yilmaz) is working correctly in these configurations. I suspect Ismail

Thanks.

Hi Novo,

Of course I won't mind. DockCtrl and it's example codes belong to U++ community. Please feel free to use/modify/change and "steal" it

Hi Mrjt,

Don't worry about "beta" testers. Currently, I am testing your Docking package on several OS configurations (ranging from latest (k)ubuntu distros (8.10/KDE 4+ and Gnome 2.24) to XP(SP3)/Vista(SP1)/Win98(SE). I have encountered some "small" bugs, but didn't have the time

to make a list of them (by the way, it seems that most of them are already exposed). I will come up with an up-to-date bug list asap.

Regards.

Subject: Re: Docking package (plus examples)
Posted by Novo on Fri, 28 Nov 2008 06:28:24 GMT
View Forum Message <> Reply to Message

Hi James,

Thank you very much! I didn't mean to push you. I just wanted to use Docking package and met all these problems. They all used to be major problems for me.

Thanks again!

Subject: Re: Docking package (plus examples)
Posted by Novo on Sun, 18 Jan 2009 19:18:09 GMT
View Forum Message <> Reply to Message

Is it possible to create hierarchical docking structure?

For example, I'm developing a database application. In database world you have master tables and detail tables (one-to-many relationship). Detail tables can have their detail tables, e.t.c.

It would be nice to be able to represent such hierarchical system as a set of docking controls.

The only limitation I see right now is that the Docking system seems to be a flat system. You have one main DockWindow, and you can dock DockingCtrl everywhere you want. The only hierarchy allowed is nested tabs via NestedTabs() call.

You can doc your controls everywhere in DockWindow.

In my database world I'd like to have a set of DockWindow (or docking containers), which is organized hierarchically, and DockingCtrl linked (and limited) to a particular docking container, so, I may dock detail controls to a particular main control only.

TIA

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 21 Jan 2009 11:15:25 GMT

Novo wrote on Sun, 18 January 2009 19:18Is it possible to create hierarchical docking structure?

For example, I'm developing a database application. In database world you have master tables and detail tables (one-to-many relationship). Detail tables can have their detail tables, e.t.c.

It would be nice to be able to represent such hierarchical system as a set of docking controls.

The only limitation I see right now is that the Docking system seems to be a flat system. You have one main DockWindow, and you can dock DockingCtrl everywhere you want. The only hierarchy allowed is nested tabs via NestedTabs() call.

You can doc your controls everywhere in DockWindow.

In my database world I'd like to have a set of DockWindow (or docking containers), which is organized hierarchically, and DockingCtrl linked (and limited) to a particular docking container, so, I may dock detail controls to a particular main control only.

TIA

Holy crap, it works! I never tried that because it just seemed too insane to work . You can have a DockWindow as a child of another DockWindow, and you can even have a Dockable DockWindow.

I've attached a modified version of DockingExample1 that shows this (see the DockInit function in main), but it's very simple:

```
 // Add our child docking window!
 DockTop(Dockable(subdock, "Child DockWindow").SizeHint(Size(300, 400)));
 subdock.DockTop(Dockable(arrayctrl2, "ArrayCtrl 2").SizeHint(Size(300, 200)));
 subdock.DockRight(Dockable(treectrl2, "TreeCtrl 2").SizeHint(Size(300, 200)));
```

File Attachments

1) DockingExample1.zip, downloaded 578 times

Subject: Re: Docking package (plus examples)
Posted by Novo on Thu, 22 Jan 2009 05:47:44 GMT

mrjt wrote on Wed, 21 January 2009 06:15
Holy crap, it works! I never tried that because it just seemed too insane to work . You can have a DockWindow as a child of another DockWindow, and you can even have a Dockable DockWindow.

I've attached a modified version of DockingExample1 that shows this (see the DockInit function in main), but it's very simple:

```
// Add our child docking window!
DockTop(Dockable(subdock, "Child DockWindow").SizeHint(Size(300, 400)));
subdock.DockTop(Dockable(arrayctrl2, "ArrayCtrl 2").SizeHint(Size(300, 200)));
subdock.DockRight(Dockable(treectrl2, "TreeCtrl 2").SizeHint(Size(300, 200)));
```

It is just cool !!! Thanks a lot !!! I thought I may have only one TopWindow or DocWindow in my application.

One small problem. When I hide a control, which belongs to subdock (treectrl2, for example), and then make it float using DocManager, treectrl2 takes main DocWindow as a container. So, I cannot dock it back into "Child DockWindow".

Subject: Re: Docking package (plus examples)
Posted by mrjt on Thu, 22 Jan 2009 10:12:40 GMT
View Forum Message <> Reply to Message

Ah, I made a mistake. The code should be:

```
// Add our child docking window!
DockTop(Dockable(subdock, "Child DockWindow").SizeHint(Size(300, 400)));
subdock.DockTop(subdock.Dockable(arrayctrl2, "ArrayCtrl 2").SizeHint(Size(300, 200)));
subdock.DockRight(subdock.Dockable(treectrl2, "TreeCtrl 2").SizeHint(Size(300, 200)));
```

Because the Dockable function registers the ctrl with the window it must be called using on the DockWindow you wish to own it, otherwise at some point there will be problems

The Docking configration window only applies to a single DockWindow instance.

Subject: Re: Docking package (plus examples)
Posted by Novo on Fri, 23 Jan 2009 04:14:52 GMT
View Forum Message <> Reply to Message

Thanks.

mrjt wrote on Thu, 22 January 2009 05:12
The Docking configration window only applies to a single DockWindow instance.

In such case how can I prevent closing of DockableCtrl? Once closed it cannot be accessed any more ...

Subject: Re: Docking package (plus examples)

Posted by mrjt on Fri, 23 Jan 2009 12:20:28 GMT
View Forum Message <> Reply to Message

You have several options here:
- Use the DockWindow::WindowButtons function to remove the close button from all it's child windows
- You can use the Docking config window to manage all of the DockWindows, but only one at a time. Just call DockWindow::DockManager on the window you want to manage.
- Provide some custom method for controlling the windows. The DockMenu class is provided to support custom menus. Or perhaps you could have a toolbar with window icons, and when the user clicking on them would call DockWindow::ActiveDockable to restore it to it's last position. There are many options here.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 23 Jan 2009 13:39:45 GMT
View Forum Message <> Reply to Message

After fiddling with it for a bit I found a few relatively minor bugs with Activate that I've fixed.

Also calling ActivateDockable on an already visible window now highlights it's position for the user. Useful on a crowded display.

Revision 797.

Also heres the code to fill a toolbar with DockableCtrl icons:

```
void DockingExample::WindowBar(Bar &bar)
{
 const Vector<DockableCtrl *> &ctrl = GetDockableCtrls();
 for (int i = 0; i < ctrl.GetCount(); i++)
  bar.Add(AsString(ctrl[i]->GetTitle()), ctrl[i]->GetIcon(), THISBACK1(ActivateWindow, ctrl[i]));
}

void DockingExample::ActivateWindow(DockableCtrl *ctrl)
{
 ActivateDockable(*ctrl);
}
```

It would be nice to limit the icons only to invisble windows, but I can't find a reliable way of triggering a Callback notification when ctrls close. There are just too many code paths that control it, but I might have a another go in the future.

Subject: Re: Docking package (plus examples)
Posted by Novo on Sat, 24 Jan 2009 05:35:36 GMT
View Forum Message <> Reply to Message

Thanks.

mrjt wrote on Fri, 23 January 2009 07:20You have several options here:
- Use the DockWindow::WindowButtons function to remove the close button from all it's child windows

prefer to completely prevent closing/hiding of controls.

Another scenario, which I'd like to see in the docking package is the ability to automatically close/hide all floating child controls when a parent control gets deactivated (by switching tabs, for example), and making them float back when parent control gets activated again. That should look like closing and reopening of a bunch of windows.

Thanks again.

Subject: Re: Docking package (plus examples)
Posted by Novo on Sun, 25 Jan 2009 03:59:02 GMT
View Forum Message <> Reply to Message

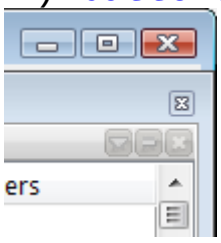
It would be nice to fix size of the "close" button on a floating window. It seems to be very small.

Windows Vista.

TIA

File Attachments

1) buttons.png, downloaded 1933 times



Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 26 Jan 2009 09:31:44 GMT
View Forum Message <> Reply to Message

Novo wrote on Sun, 25 January 2009 03:59It would be nice to fix size of the "close" button on a floating window. It seems to be very small.
I don't have a Vista machine at the moment, but I guess this is because I'm calling ToolWindow on the child container windows. On XP I think this looks much better, on X11 and perhaps Vista not so much.

I've added a function DockWindow::ChildToolWindows so that you can specify this yourself. Defaults to true on Win32, false on X11. Revision 812.

Subject: Re: Docking package (plus examples)
Posted by Novo on Tue, 27 Jan 2009 05:15:52 GMT
View Forum Message <> Reply to Message

Thanks!

Minor problem:

"autohide" is not initialized within DockWindow::DockWindow()

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 27 Jan 2009 11:15:00 GMT
View Forum Message <> Reply to Message

Novo wrote on Tue, 27 January 2009 05:15Minor problem:

"autohide" is not initialized within DockWindow::DockWindow()

It is, but I call DockWindow::AutoHide() rather than directly initialising the member variable. The initialisation could be cleaned up though.

Subject: Re: Docking package (plus examples)
Posted by Novo on Wed, 28 Jan 2009 04:53:52 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 27 January 2009 06:15Novo wrote on Tue, 27 January 2009 05:15Minor problem:

"autohide" is not initialized within DockWindow::DockWindow()

It is, but I call DockWindow::AutoHide() rather than directly initialising the member variable. The initialisation could be cleaned up though.

```
DockWindow&  DockWindow::AutoHide(bool v)
{
 if (v != autohide) {
  autohide = v;
  SyncAll();
 }
 return *this;
}
```

You call AutoHide(true) in the constructor with uninitialized "autohide".

It is hard to fool valgring

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 28 Jan 2009 08:32:45 GMT
View Forum Message <> Reply to Message

Technically true, but it will always work .

It is sloppy and I will fix it. I just didn't want to do a whole revision for something so minor.

Subject: Re: Docking package (plus examples)
Posted by White_Owl on Thu, 29 Jan 2009 20:16:22 GMT
View Forum Message <> Reply to Message

I am using MinGW with -Wall and packages Bazzaar/Docking and Bazaar/TabBar (which is called from Docking) give me a lot of warnings...
Here is a zip with all modified files, old files and diffs for both packages.
Mostly it is a unused variables, functions without returns and warnings about nested constructors (I guess GNU C++ just does not like them).

File Attachments
1) Changes_to_Docking_and_TabBar.zip, downloaded 562 times

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 30 Jan 2009 09:57:27 GMT
View Forum Message <> Reply to Message

Thanks. I've incorporated most of those changes.

There was one case where your fix was incorrect though:

25c25
< AlignedFrame() : border(0), framesize(0), layout(TOP) {}

> AlignedFrame() {border=0; framesize=0; layout=TOP;}
129,130c129,130
< Tab() : visible(true), id(-1)
< {}

> Tab() {visible=true; id =-1; }
What MingW is actually complaining about here is not the use of the initialiser list, but that the list is in a different order than the member variables are declared. Frankly I cannot see the purpose of such a warning, so I'm not going to spend the time fixing it. The same goes for all of the unnecessary parenthesis warnings, because I was aware of operator precedence when I wrote the code thank you very much MingW

All of those missing returns and unused variables (usually from debugging) were very good to fix though.

Revision 826.

Subject: Re: Docking package (plus examples)
Posted by cbpporter on Fri, 30 Jan 2009 10:06:39 GMT
View Forum Message <> Reply to Message

mrjt wrote on Fri, 30 January 2009 11:57
What MingW is actually complaining about here is not the use of the initialiser list, but that the list is in a different order than the member variables are declared. Frankly I cannot see the purpose of such a warning, so I'm not going to spend the time fixing it.
MinGW is probably complaining because when using initializer lists the members are initialized in the order they were declared in the class, not the order from the initialized list. This can lead to ugly bugs if one member depends on the value of another. But it is not the case right here, so I guess MinGW thinks it's better to always nag, as is the case with those parenthesis.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 30 Jan 2009 11:11:25 GMT
View Forum Message <> Reply to Message

cbpporter wrote on Fri, 30 January 2009 10:06MinGW is probably complaining because when using initializer lists the members are initialized in the order they were declared in the class, not the order from the initialized list. This can lead to ugly bugs if one member depends on the value of another. But it is not the case right here, so I guess MinGW thinks it's better to always nag, as is the case with those parenthesis.

Interesting. I didn't realise the initialise order was different, not that I've every depended on it.

There's always something new to learn with C++.

And always nag is certainly the safer option. I did have -Wall, so I was pretty much asking for a nagging

---

Subject: Re: Docking package (plus examples)
Posted by White_Owl on Fri, 30 Jan 2009 15:46:25 GMT
View Forum Message <> Reply to Message

mrjt wrote on Fri, 30 January 2009 12:11Interesting. I didn't realise the initialise order was different, not that I've every depended on it. There's always something new to learn with C++.
So can you fix it for a next revision? Pleeese... So many warnings...

Subject: Re: Docking package (plus examples)
Posted by vasil on Mon, 16 Mar 2009 13:06:24 GMT
View Forum Message <> Reply to Message

Hi mrjt,

First of all good work with the docking stuff.

Can I propose a feature or two!?

1. Is it possible to add maximize ability to a docked window? I think that this is useful because I worked with both MSVS and Eclipse and one feature that I like from the eclipse docking system is the ability to maximize the docked windows. Of course this is possible in MSVS too but you have to make several mouse click to make all the windows in auto hide mode except the one you want maximized.

2. I do understood (or may be I am wrong) that making a window transparent is not amongst the current features of UPP, so it is not possible making the window that is being dragged transparent so it is more clearly visible where you want to drop it. As long as understand with the UPP you can make drag&drop operations and you can have an transparent image that moves along with the mouse cursor. So is it possible once somebody starts dragging a window to make a scaled image of it then hide the window and use that image for the drag&drop operation and once it is dropped onto new docking location window is then shown again?

--Vasil

P.S. Excuse my English it is not my native language

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 16 Mar 2009 14:23:47 GMT
View Forum Message <> Reply to Message

vasil wrote on Mon, 16 March 2009 13:06Hi mrjt,

First of all good work with the docking stuff.

Can I propose a feature or two!?

1. Is it possible to add maximize ability to a docked window? I think that this is useful because I worked with both MSVS and Eclipse and one feature that I like from the eclipse docking system is the ability to maximize the docked windows. Of course this is possible in MSVS too but you have to make several mouse click to make all the windows in auto hide mode except the one you want maximized.

The only major problem with this feature is with managing the toggling between maximised and non-maximised states. This is further complicated because DockableCtrls can have a maximum size, so won't necessarily fill up the whole parent window.

I think I know how this could work, but it may require lots of complicated additions and modifications, in which case I'm reluctant to add it. I will need to think about it further.

Quote:
2. I do understood (or may be I am wrong) that making a window transparent is not amongst the current features of UPP, so it is not possible making the window that is being dragged transparent so it is more clearly visible where you want to drop it. As long as understand with the UPP you can make drag&drop operations and you can have an transparent image that moves along with the mouse cursor. So is it possible once somebody starts dragging a window to make a scaled image of it then hide the window and use that image for the drag&drop operation and once it is dropped onto new docking location window is then shown again?
I don't think the cursor idea would work, but I will look at adding this on Win32 with platform specific code and the possibility of adding X11 support in the future (but in any case it would require a correctly configured Window Manager). I'll test it and get back to you, but I don't have time today.

Quote:P.S. Excuse my English it is not my native languageYour English is excellent

Subject: Re: Docking package (plus examples)
Posted by cocob on Wed, 08 Apr 2009 10:53:42 GMT
View Forum Message <> Reply to Message

Is there any plans for improvements of PopUpDockWindow ?

Thanks

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 08 Apr 2009 14:07:34 GMT
View Forum Message <> Reply to Message

I had kind-of forgotten about it to be honest, it hasn't been updated to match changes in the base DockWindow for while so it might not work.

But strangely this is the second request I've had for this today, so I'll be having a look at it soon. Expect an update next week.

Subject: Re: Docking package (plus examples)
Posted by cocob on Wed, 08 Apr 2009 14:54:14 GMT
View Forum Message <> Reply to Message

I can confirm that it does'nt works very well.
thanks for your work

Subject: Re: Docking package (plus examples)
Posted by Didier on Sat, 25 Apr 2009 22:47:09 GMT
View Forum Message <> Reply to Message

Great job on docking

I have looked at the code in order to find solutions to my current problem and I have the following question:

1 Can we remove a docked ctrl (I do not mean hide)?
2 The 'ctrls' member in DockWindow is only accessed to 'add' never to remove ?!?.
3 I am searching for a hook-point in order to launch the delete on dock close and I haven't found any yet

In fact I want to enhance the package in order to be able to write code like in the following example:

```
// MyControlType  is a derived from TopWindow
DockCtrl& dc = CreateDockable<MyControlType>(param1, param2, ...);
dc.setTitle("xxxxxx");
Float(dc);
```

The point is to be able dynamically create dialogs that can be docked/floated while delegating the 'new()' statement and associated management to dockWindow.

The instanciated control would get deleted when the dialog gets closed.

Maybe I'm searching the wrong way so if it's the case and some solution exists please tell me.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 27 Apr 2009 12:54:10 GMT
View Forum Message <> Reply to Message

I was going to say that I wasn't going to add this, but I guess someone else will probably want it in future anyway.

Some drawbacks:
You will not be able to Serialize the layout.
When the user closes and reopens a window it will not remember where it last was.
You will not be able to use the in-built menus to open windows.

You do this:

```
// MyControlType must be deriveed from DockableCtrl
void AddCtrl() {
 DockableCtrl &dc = DockWindow::CreateDockable<MyControlType>("Title");
 dc.WhenState = THISBACK1(OnDockerState, &dc);
 Float(dc);
}

void OnDockerState(DockableCtrl *dc) {
 if (dc->IsHidden())
  DockWindow::Deregister(*dc);
}
```

You will need the most recent SVN update. I'll upload it shortly (just fixing a last unrelated problem).

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 27 Apr 2009 13:33:19 GMT
View Forum Message <> Reply to Message

Revision committed (1104)

cocob wrote on Wed, 08 April 2009 15:54I can confirm that it does'nt works very well.
thanks for your work
I've checked this now, and it seems as it works how it's suppossed to. I've also added icons for autohiding (use ShowAutoHidePopUps to disable).

If you have any suggestions for improvements let me know.

Subject: Re: Docking package (plus examples)
Posted by Didier on Mon, 27 Apr 2009 19:56:36 GMT
View Forum Message <> Reply to Message

Fantastic !

I'm going to try it right now.

I needed this functionnality in order to open several identical dialogs to make comparison views: no need for the special menu features and of course even less for serialisation.

Thanks

Didier

Subject: Re: Docking package (plus examples)
Posted by Didier on Tue, 28 Apr 2009 19:32:53 GMT
View Forum Message <> Reply to Message

OK works fine

I added the following additional creator methods to enable constructors with parameters:

in 'class DockWindow'

```
 template<class T>
 DockableCtrl&   CreateDockable(WString title)      { return Register(ctrls.Create<T>().Title(title)); }
 template<class T, class PARAM1>
 DockableCtrl&   CreateDockable(WString title, PARAM1 p1)   { return
Register(ctrls.Create<T>(p1).Title(title)); }

 template<class T>
 DockableCtrl&   CreateDockable(const char *title = 0)          { return
CreateDockable<T>((WString)title); }
 template<class T, class PARAM1>
 DockableCtrl&   CreateDockable(const char *title, PARAM1 p1)       { return
CreateDockable<T>((WString)title, p1); }
```

in 'class Array' : Core/Vcont.h

```
 template<class TT> TT& Create()     { TT *q = new TT; Add(q); return *q; }
 template<class TT, class PARAM1> TT& Create(PARAM1 p1)     { TT *q = new TT(p1); Add(q);
return *q; }
```

Maybe it's a good idea to add such methods with 1,2,3,4 ... parameters since Array does not use Pick behaviour (copy is a move)

What do you think ?

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 29 Apr 2009 07:53:42 GMT
View Forum Message <> Reply to Message

Didier wrote on Tue, 28 April 2009 20:32
Maybe it's a good idea to add such methods with 1,2,3,4 ... parameters since Array does not use Pick behaviour (copy is a move)

What do you think ?
This should be avoided. The Upp way is to almost always use default constructors, as this solves many common problems. You should use some sort of initialisation function instead. The special cases are usually classes that are scoped as local variables, and never Ctrl derived classes since these are usually member variables.

I had made a mistake that made this more difficult though. I've just updated the SVN with the fix (CreateDockable now returns type T&). You can now do:
MyClassType &c = DockWindow::CreateDockable<MyClassType>("Title");
c.Init(param1);
or event better:
DockWindow::CreateDockable<MyClassType>("Title").Init(param1);

Subject: Re: Docking package (plus examples)
Posted by Novo on Mon, 08 Jun 2009 01:57:10 GMT
View Forum Message <> Reply to Message

Could you please fix Docking package? It is broken (rev. 1289) because of changes to TabBar.

TIA

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 16 Jun 2009 08:35:51 GMT
View Forum Message <> Reply to Message

Yes. I was on holiday all of last week but I'll have it done either today or tomorrow.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 16 Jun 2009 15:09:15 GMT
View Forum Message <> Reply to Message

Done. Revision 1315.

Subject: Re: Docking package (plus examples)
Posted by ptDev on Thu, 18 Jun 2009 11:29:09 GMT
View Forum Message <> Reply to Message

Hi,

Thanks for the Docking library and the update to make it work with the new version of TabBar.

I've been trying the Docking library, and noticed that a small bug. If you only have one docked window, and it is minimized, closing it causes the program to crash. Is this behaviour expected in any way?

Subject: Re: Docking package (plus examples)
Posted by mrjt on Thu, 18 Jun 2009 12:13:25 GMT
View Forum Message <> Reply to Message

Quote:Hi,

Thanks for the Docking library and the update to make it work with the new version of TabBar.

I've been trying the Docking library, and noticed that a small bug. If you only have one docked window, and it is minimized, closing it causes the program to crash. Is this behaviour expected in any way?
It's a bug, though quite and interestingly subtle one. I've just committed the fix, plus I've stopped the AutoHide bar from always having an active tab (changes to both TabBar and Docking).

Subject: Re: Docking package (plus examples)
Posted by Novo on Fri, 19 Jun 2009 01:54:44 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 16 June 2009 11:09Done. Revision 1315.

Thanks for the fix.

It looks like rev. 1320 doesn't work on Linux. I checked that with 32 and 64-bit versions of Ubuntu 9.04 with Compiz enabled.

Floating window cannot be docked back.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 19 Jun 2009 07:38:20 GMT
View Forum Message <> Reply to Message

Bugger. I've never tested it with Compiz and none of the floating/docking code has changed, so perhaps it's that.

This might be much more difficult to fix, what I need is a log of X events during window draging and dropping.

Subject: Re: Docking package (plus examples)
Posted by Oblivion on Fri, 19 Jun 2009 12:31:04 GMT
View Forum Message <> Reply to Message

Hi mrjt,

I can confirm that bug on Ubuntu + compiz too. OTOH, docking package (build 1320) works flawlessly on Kubuntu 9.04 + KDE 4.2.4 with/without window compositing enabled. Nice job! The only thing that is annoying me is the docking animation gets choppy on docks with 2 or more widgets.

Regards.

Subject: Re: Docking package (plus examples)
Posted by Novo on Sun, 21 Jun 2009 22:52:38 GMT
View Forum Message <> Reply to Message

mrjt wrote on Fri, 19 June 2009 03:38Bugger. I've never tested it with Compiz and none of the floating/docking code has changed, so perhaps it's that.

This might be much more difficult to fix, what I need is a log of X events during window draging and dropping.

Sorry for making you a problem again.

I attached a log of X11 events for DockingExample1. It is created using xtrace. I hope that it is what you need.

TIA

[Attachment is deleted ...]

Subject: Re: Docking package (plus examples)
Posted by mrjt on Tue, 23 Jun 2009 10:07:53 GMT

View Forum Message <> Reply to Message

I don't have much time to look at it now, but that might be too much information to get what I need. The best way is to add logging to the X11 window proc, i'll give you a patch to do this when I can.

Subject: Re: Docking package (plus examples)
Posted by cocob on Thu, 25 Jun 2009 20:51:00 GMT
View Forum Message <> Reply to Message

is it possible to remove this line in the begining of Docking.h

using namespace Upp;

Because it is very diffcult to resolve names when some "using" are set in header files.

Thanks
cocob

Subject: Re: Docking package (plus examples)
Posted by mrjt on Fri, 26 Jun 2009 09:04:38 GMT
View Forum Message <> Reply to Message

Would you prefer namespace Upp;?

Subject: Re: Docking package (plus examples)
Posted by ptDev on Fri, 26 Jun 2009 13:29:20 GMT
View Forum Message <> Reply to Message

I'm sorry for my confusion.

Is it that you wish to always type the additional "Upp::" scope prefix whenever you're referring to an U++ class?

Note that when you're making a new package, you can always omit the "using namespace Upp" statement and be explicit about the namespaces. In fact, you can even still refer to the namespaces for clarity, "using" or no "using".

Personally, I never saw any obvious advantage in this, but you do have the option to do so if you wish.

Subject: Re: Docking package (plus examples)
Posted by cocob on Fri, 26 Jun 2009 16:55:48 GMT

View Forum Message <> Reply to Message

I use "using namespace Upp" too ! but not in header files !
It is the right way to proceed.

Yes i prefer namespace Upp { }

Thanks
cocob

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 29 Jun 2009 08:49:48 GMT
View Forum Message <> Reply to Message

Ok, committed. I can see your point. It's just never an issue for me as I will always be using the Upp namespace.

Subject: Re: Docking package (plus examples)
Posted by Novo on Sun, 19 Jul 2009 16:43:48 GMT
View Forum Message <> Reply to Message

mrjt wrote on Tue, 23 June 2009 06:07I don't have much time to look at it now, but that might be too much information to get what I need. The best way is to add logging to the X11 window proc, i'll give you a patch to do this when I can.

I can try to enable LLOG ...
Is that what you mean?

Subject: Re: Docking package (plus examples)
Posted by mrjt on Mon, 20 Jul 2009 10:36:29 GMT
View Forum Message <> Reply to Message

I think that should do it, but it's better if you can exclude expose events and some other stuff.

The problem is that there is no X11 event for 'this window is no longer being dragged'. In Windows this is quite easy, but for X11 I have to rely on unrelated events, and presumably Compiz is sending a slightly different sequence.

Sorry I don't have much time for this at the moment, but I will get to it eventually. If I new what events were needed it should be simple fix.

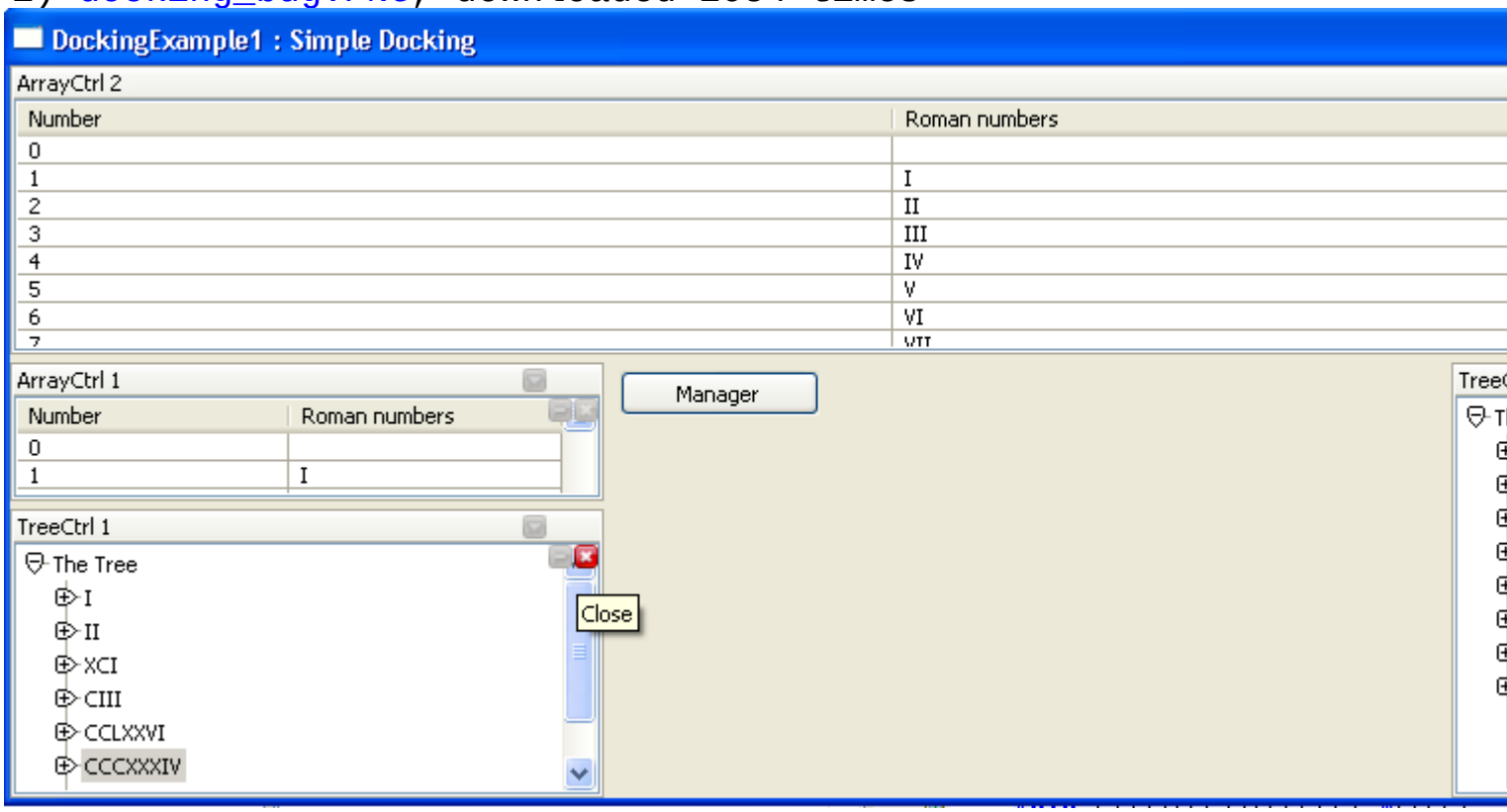
Subject: Re: Docking package (plus examples)
Posted by tojocky on Mon, 03 Aug 2009 08:43:27 GMT
View Forum Message <> Reply to Message

Hello All,
Very interesting project.. by I thing that have a little bag.

This bug i produced on win32 windows xp os with the latest svn repository. When I unmark and after then mark Windows Menu button from Docking window Manager form, The buttons of the docking windows was positioned wrong.

File Attachments

1) docking_bug.PNG, downloaded 2054 times



Subject: Re: Docking package (plus examples)
Posted by Didier on Sun, 30 Aug 2009 17:07:47 GMT
View Forum Message <> Reply to Message

Hi,

I have the latest svn version and there are two Docking pacakges:
one uppdev and one in bazaar.
The one in Bazaar works, the other one has unresolved conflicts and therefore does not compile.

Which one should be used ?

Subject: Re: Docking package (plus examples)
Posted by koldo on Tue, 01 Sep 2009 10:57:20 GMT
View Forum Message <> Reply to Message

Hello Didier

As far as I know in Bazaar there are the "stable" versions of packages submitted by the Upp community.

Therefore the right package to use would be the one in Bazaar.

Best regards
Koldo

Subject: Re: Docking package (plus examples)
Posted by koldo on Tue, 01 Sep 2009 10:59:19 GMT
View Forum Message <> Reply to Message

Hello James Thomas

Here I enclose you a translation to spanish.

Best regards
Koldo

File Attachments
1) Docking.t, downloaded 518 times

Subject: Re: Docking package (plus examples)
Posted by Didier on Tue, 01 Sep 2009 18:44:27 GMT
View Forum Message <> Reply to Message

Hi Koldo,

Thanks !

Subject: Re: Docking package (plus examples)
Posted by tojocky on Wed, 23 Sep 2009 11:38:55 GMT
View Forum Message <> Reply to Message

I see that this bug is not corrected.

It is persist in ubuntu 9.04 too!

tojocky wrote on Mon, 03 August 2009 11:43Hello All,
Very interesting project.. by I thing that have a little bag.

This bug i produced on win32 windows xp os with the latest svn repository. When I unmark and after then mark Windows Menu button from Docking window Manager form, The buttons of the docking windows was positioned wrong.

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 07 Oct 2009 16:55:51 GMT
View Forum Message <> Reply to Message

tojocky wrote on Wed, 23 September 2009 12:38I see that this bug is not corrected.

It is persist in ubuntu 9.04 too!

tojocky wrote on Mon, 03 August 2009 11:43Hello All,
Very interesting project.. by I thing that have a little bag.

This bug i produced on win32 windows xp os with the latest svn repository. When I unmark and after then mark Windows Menu button from Docking window Manager form, The buttons of the docking windows was positioned wrong.

Fixed. It was actually caused by hiding the buttons then showing them agan.

I haven't merged the translation files yet.

Subject: Re: Docking package (plus examples)
Posted by andrei_natanael on Tue, 27 Oct 2009 10:22:21 GMT
View Forum Message <> Reply to Message

The un-docked window doesn't get docked back, it doesn't get docking hints shown. That happen on Gnome 2.28 with Gnome effects (not compiz), Ubuntu 9.10.

Subject: Re: Docking package (plus examples)
Posted by softcoder on Wed, 11 Nov 2009 12:22:03 GMT
View Forum Message <> Reply to Message

At first: Thank you for this great cool thing!
But I think I've found a bug concerning styles in docking (I'm using WinXP).

To see what I mean, perform the following steps:
- Open DockingExample2 as project in UPP so you can compile it etc.
- Add this function (which is basing on BlueBar):

```
void DockingExample::ChangeStyle()
{
 TabCtrl::Style& ctrlTabBarStyle = DockTabBar::StyleDefault().Write();
 ctrlTabBarStyle.Standard();
 for(int i = 0; i < 4; i++)
 {
  ctrlTabBarStyle.text_color[i] = White();
 }
}
```

- In DockingExample::DockingExample() replace
button <<= THISBACK(OnUserGuide);

by

button <<= THISBACK(ChangeStyle);

- Compile and execute the program.
- Look at the tabs (TreeCtrl 1 and 2) at the left bottom of the window. The font color is black as it should be.
- Pull one of the tabs out of the window so it becomes an additional floating window.
- Drag'n'drop the floating window where it was before.
- Now push the "User Guide" button -> font doesn't become white although it should become white.
- Pull out the tab again and drop it again to its old place -> font is white.

My current workaround (or even the best possible bugfix?) is to add the following line directly before Refresh() in DockCont::GroupRefresh():

tabbar.SetStyle(tabbar.GetAlign(), DockTabBar::StyleDefault());

This problem even occurs in UPP1679 (the newest version availabe here, but mainly using a much older one for some reasons).

Now the question: Is the problem sitting in front of my screen or is that really a bug? If it's a bug, is it a good solution?

TIA

softcoder

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 11 Nov 2009 17:23:35 GMT
View Forum Message <> Reply to Message

The problem is that TabBar requires a different style for each orientation. These are generated on demand and you have to make sure you modify the correct one.

The Docking tabs use bottom alignment, so this should work:

```
void DockingExample::ChangeStyle()
{
 TabCtrl::Style& ctrlTabBarStyle = DockTabBar::StyleBottom().Write();
 ctrlTabBarStyle.Standard(); // Is this necessary?
 for(int i = 0; i < 4; i++)
 {
  ctrlTabBarStyle.text_color[i] = White();
 }
}
```

(untested though)

I'm suprised your solution works actually. This code:
tabbar.SetStyle(tabbar.GetAlign(), DockTabBar::StyleDefault());

is applying the style for top-aligned tabs to a bottom aligned TabBar. I would expect the tabs to be the wrong way up.

Subject: Re: Docking package (plus examples)
Posted by mdelfede on Sat, 12 Dec 2009 18:44:41 GMT
View Forum Message <> Reply to Message

Hi,

I'm testing your docking package, really impressive

I have just a small question : I'd need some sort of tabbed docking, but without having a dockable container... I mean, I'd need to tabify-untabify some windows inside a FIXED container, which shouldn't be dockable. When last tab is undocked it should stay empty in-place waiting for pages to be eventually re-tabified.
The docking bar shouldn't be present, either, just the tabs.
Is it possible ?

Another nice thing would be the ability to lock/unlock some parts or the whole interface.

Ciao

Max

Subject: Re: Docking package (plus examples)
Posted by Didier on Sun, 13 Dec 2009 09:33:59 GMT
View Forum Message <> Reply to Message

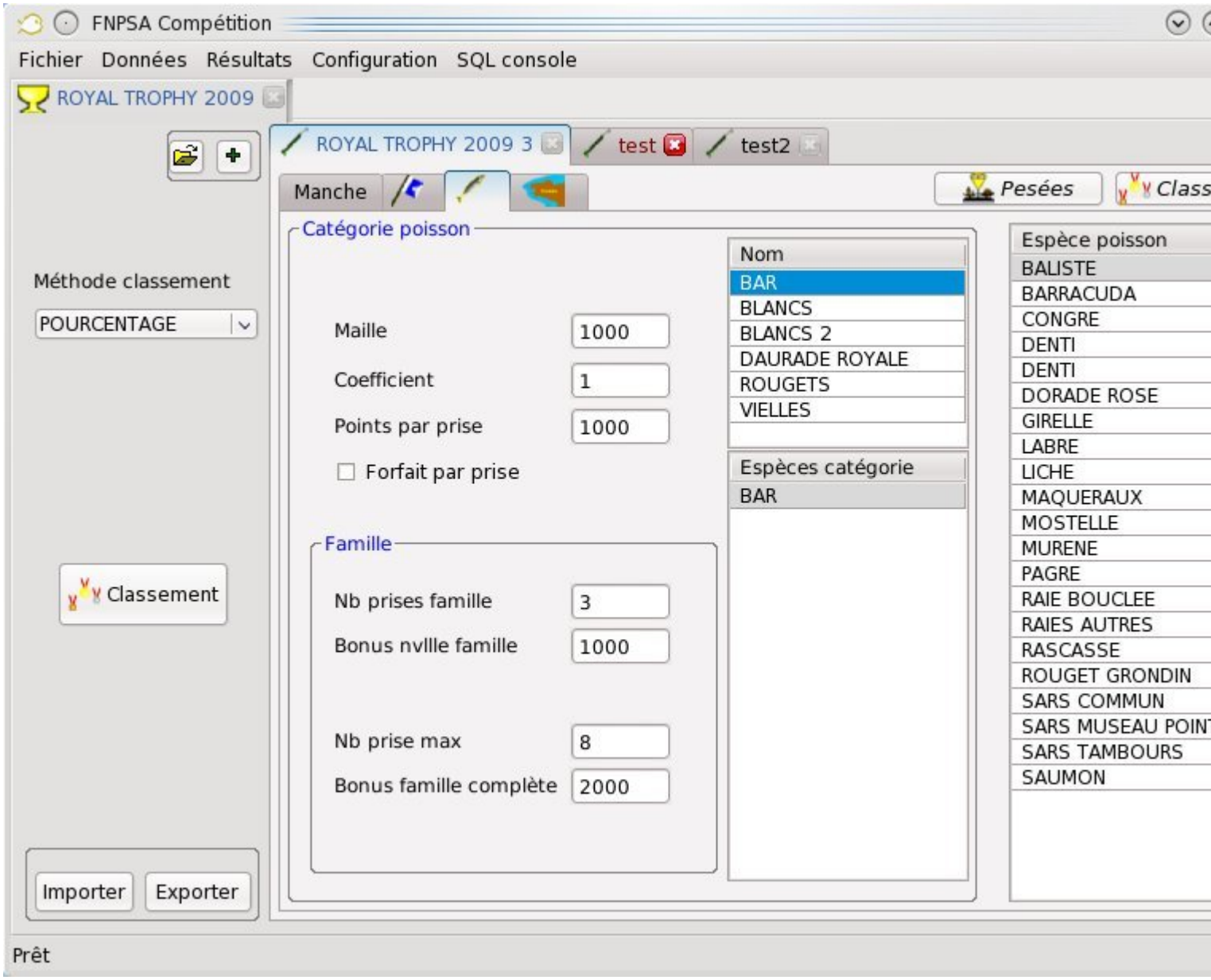
Hi,

Try TabCtrl: tab management without docking facility.
I'm not it's what you need but it's simple and works well, you can also put controls in the tab to do whatever you want

Here is a screen shot of where you can see the different possibilities

File Attachments

1) TabsExample.jpg, downloaded 1745 times



Subject: Re: Docking package (plus examples)
Posted by mdelfede on Sun, 13 Dec 2009 09:56:32 GMT
View Forum Message <> Reply to Message

Hi Didier,

That's not exactly what I meant, but it's indeed what I'm using
I'll wait to introduce docking in my app later on.
What I wanted to do was the ability to open more application files, show them inside a tabbed view and have the ability to undock each "job" to have it on a separate window, if needed.... handy to look, for example, at 2 jobs at once.
Just a little comfort more when copy/paste between different jobs whithout having to switch

between them with the tabs.
I like also the autohide function, for page selection, for example, but I'll wait to introduce it later, too. I think I'll have to extend a bit the docking package, but no time now.

Thank you for answer

Ciao

Massimo

Subject: Re: Docking package (plus examples)
Posted by mrjt on Wed, 16 Dec 2009 12:36:30 GMT
View Forum Message <> Reply to Message

I think everything you need is in the TabBar package, and you can add docking later. It's much more suited to a document tab system than TabCtrl is.

Just make a TopWindow derived document viewing window with a TabBar frame and add the callback behaviour. You may only need to use WhenClose and WhenAction, the others are there to support complex behaviour.

Unfortunately the Autohide behaviour is missing, I'm not sure why. Possibly I forgot to add it but it can also cause some layout problems so maybe it's left out for that reason. The DockTabBar class in the Docking package implements it and it's not much code so you could add it youself to a derived class.

If you want to overload the right-click menu you must create a derived class and overload ContextMenu.

There is a lot of additional stuff you can do with it if you want to (custom icons and Display, groups, key/value behaviour and stacking to name a few) but out-of-the box it should fit your requiredments fairly well. Let me know if you need some assistance, there isn't any documentation at the moment.

James

Subject: Re: Docking package (plus examples)
Posted by mdelfede on Sat, 19 Dec 2009 22:06:40 GMT
View Forum Message <> Reply to Message

Hi mrjt,

thank you for suggestions, I' already using tabctrl, but I'll try to add docking/autohide later on,

when app will be finished.

Ciao

Max
