QGIS Application - Bug report #5753
Menu items out of order on OSX

2012-06-06 03:04 PM - John Tull

Status:	Closed			
Priority:	Normal			
Assignee:				
Category:	GUI			
Affected QGIS version:	master	**Regression?:**	No
Operating System:	OS X	**Easy fix?:**	No
Pull Request or Patch supplied:	No	**Resolution:**	fixed
Crashes QGIS or corrupts data:	No	**Copied to github as #:**	15259

Description

On OS X, the menu items are mis-ordered. I've attached a screenshot, but the key is that Help and Window menus should be the last two items, respectively, but they are not. Some of the plugin menus are being appended, but they should be before the Window and Help menus to maintain standard GUI on the operating system.

History

#1 - 2012-06-06 03:12 PM - Jürgen Fischer

- Priority changed from Severe/Regression to Normal

#2 - 2012-06-06 10:55 PM - Paolo Cavallini

See #5516 (this seems a duplicate)

#3 - 2012-06-06 11:01 PM - Larry Shaffer

Paolo Cavallini wrote:

> *See #5516 (this seems a duplicate)*

Not exactly. What John is mentioning is a Mac OS convention. Other changes to the menu order could be settled on by QGIS devs, like in #5516. I have a fix almost done for putting Window and Help always to the right on Mac. The Window menu is also added as a Mac-only menu to meet OS convention.

#4 - 2012-06-07 12:24 AM - Giovanni Manghi

I have seen this issue also on Linux/Ubuntu.

#5 - 2012-06-07 05:34 AM - Larry Shaffer

- File mac-menus-patch-1.8_git.diff added

I have 'fixed' this in my recent builds. Apply attached patch to git repo to test.

Here what I did:

1) Adjusted the Mac-only addition of a Window menu, by placing it just before the Help menu, as per Mac OS convention.

2) Removed extraneous, old 'Raster' menu-adding from GdalTools.py. This secondary menu adding routine was hijacking the raster menu and always placing it just to the left of the rightmost menu. This is why it showed up between Help and Window (when they were in reversed order).

3) Adjusted the *Database, Web, Vector* menu inserts to be linearly ordered up against the Raster menu, which is the only one in src/ui/qgisapp.ui.I don't understand the need for the extra code for menu insertions, and deletions, relative to plugin load/unload.

Since these are now standardized menus, couldn't they be added to src/ui/qgisapp.ui (like Raster), then have their visibility turned off on initial app load? When plugins are first loaded/last unloaded into one of the menus, it could just be setVisible(true/false) relative to whether it had a plugin in it.

Note: this does not solve the problem of extra menus from third-party plugins coming between Window and Help menus, especially outdated plugins in the user's plugin folder, or ones that manually insert their menu. One such plugin is CadTools, which uses an apparently common menu-adding snippet:

```
menu_bar = self.iface.mainWindow().menuBar()
actions = menu_bar.actions()
lastAction = actions[ len( actions ) - 1 ]
menu_bar.insertMenu( lastAction, self.menu )
```

which causes insertion between Window and Help on Mac. The only solution to this, I see, is if QMenuBar had a signal for when an action was added (it doesn't). Then, the menus could be auto-shuffled according to some accepted scheme. Maybe QMenuBar and its changeEvent() could be reimplemented to add flexibility.

#6 - 2012-06-07 05:57 AM - Larry Shaffer
- File mac-menus-patch-1.8_2_git.diff added

Small change to patch, to include a version increment for GdalTools to trigger an update notice for users who might have an older, but same version, copy of GdalTools in their user plugin's folder.

This has already been fixed for fTools in master branch.

#7 - 2012-06-11 09:09 AM - John Tull

This is another good code update. I suppose this needs to remain open until a workaround for third-party plugins is made or a determination that plugins inserting their own menu item is not acceptable on OS X, thus making it a plugin bug rather than a QGIS bug. If the latter is preferable, and there is a reasonable code solution for plug managers to use to get around this problem, than this should be closed.

#8 - 2012-06-11 09:51 AM - Larry Shaffer

John, this has been mostly 'fixed' (good enough for release of 1.8, I think) in

commit:7f1a4e28d12e3fc2bd40d596a4f09138ed368f79

The problem, as you surmised, is now with third-party plugins, and any now-core plugins that you still have installed in your user plugins folder (remove those with plugin installer). In that commit I mention a solution that third-party plugins can use:

```
menu_bar.insertMenu( self.iface.firstRightStandardMenu().menuAction(), self.my_plugins_menu )
```

This works well when the plugin is first installed, installing it menu just before Window (on Mac) and Help for other platforms.

However, on re-launch of QGIS, third-party and core plugins are added alphabetically, enabling additional menus (e.g. Vector, Database, Web) relative to whether those menus have plugins. For example, CadTools updated with the firstRightStandardMenu() code will show up correctly on first install, but after relaunch, will have its menu show up before the Database menu because **db_manager** loads after **cadtools** package. This is because the firstRightStandardMenu() menu placement was correct, *before* Database menu was filled with a plugin and became enabled.

One further solution exists from here (short of asking all plugin devs to change their affected plugins):

Edit plugin_installer, or whatever parses and adds plugins on launch, to parse all plugins to load, queuing plugins that have a category set (i.e. will make one of the standardized menus show up) and push those plugins to the front of the list before loading. Then, all standard plugin menus will be filled and become visible before any plugin that uses firstRightStandardMenu() will be added, which would then make firstRightStandardMenu() work properly.

Lastly, once that's fixed, IMHO, plugin developers should be recommended to add their plugins to the standard menus, use firstRightStandardMenu(), add their plugin menu after all QGIS menus, or just **please don't** add their own menus.

Yes, it's going to still be a bit perplexing to users until fixed. This last issue affects **all platforms**, but is not a regression, as it existed in 1.7.4.

Hope all that made sense.

#9 - 2012-06-11 09:54 AM - Larry Shaffer

Yes, I think the issue should stay open (but not hold up 1.8 release in any way), and the title changed to 'Menu items out of order due to third-party Python plugins'.

#10 - 2012-06-13 05:06 PM - Larry Shaffer

I believe I have finally fixed this, for all platforms, in this pull request:
https://github.com/qgis/Quantum-GIS/pull/174

This fix can also be adapted for new dynamically added/removed plugin category menus, like maybe Analysis, in the near future.

It takes a different approach to dynamically building standard plugin category menus, by putting them after or before the Raster menu, which is already enabled in /src/ui/qgisapp.ui.

If third-party plugins use firstRightStandardMenu() as a key for placing their plugin menus before it, this fix should now keep the menubar properly ordered, with dynamically added/removed standard plugin category menus, as such:

File, Edit, View, Layer, Settings, Plugins, **, Raster, [Database], [Web], [Plugin-added-menus]**, Window (on Mac), Help

You can now add/remove, in the Plugin Manager, *any number of combinations* of plugins that use category menus or their own menu. You can do this repeatedly, without the menus losing order, even after re-launching QGIS.

Sometimes, the order of third-party plugin menus will be different between initial adding of plugin and re-launching of QGIS, but this is a difference between manually adding plugins via Manager, and the packagename-alphabetical manner of loading plugins from qgis.utils on launch. I don't see this as a bug, just a bit of a nuisance.

This is how it works:

- Plugin category menus are built relative to each other, not *any other* menu to their right:

  [Vector] -> Raster <-> [Database] <-> [Web]

- Third-party plugins can now load their own menus using:

    menuBar = self.iface.mainWindow().menuBar()
    menuBar.insertMenu(self.iface.firstRightStandardMenu().menuAction(),
            self.my_plugins_menu)

> *so they will always show up in this fashion, not interfering with plugin category menus, or rightmost OS-specific conventional menus:*
>   *[Plugin-added-menu, Plugin-added-menu] -> Window (on Mac), Help*

This paves the way for adding a convenience public slot in QgisInterface (to a function in qgisapp.cpp) that offers a means for plugin authors to always correctly add their own plugin menus between the category menus and the OS-convention menus, maybe worded like:

    addPluginMenuToMenuBar(QMenu) / removePluginMenuFromMenuBar(QMenu)

Sometimes, there may be too many menus in the menubar for the user's screen size and some menus become obscured. Having third-party plugin menus before any firstRightStandardMenu (instead of after) will allow the functional menu (the plugin's) to possibly be visible, while Window or Help might be obscured. I think this is a fair trade-off, given the user wants to use the plugin, probably more than those menu actions.

#11 - 2012-06-13 05:09 PM - Larry Shaffer

In the post above the order is:

File, Edit, View, Layer, Settings, Plugins, [Vector], Raster, [Database], [Web], [Plugin-added-menus], Window (on Mac), Help

(something happened to 'Vector' when posting)

#12 - 2012-06-21 01:36 AM - Jürgen Fischer

- Resolution set to fixed
- Status changed from Open to Closed

applied in commit:de6f64215

Files

qgis_osx_menus.png	61.3 KB	2012-06-06	John Tull
mac-menus-patch-1.8_git.diff	2.83 KB	2012-06-07	Larry Shaffer
mac-menus-patch-1.8_2_git.diff	3.22 KB	2012-06-07	Larry Shaffer