

# Some thoughts on toolbars in KDE

## 1.1 A brief introduction

This document describes some of my thoughts and suggestions regarding the toolbars in KDE. The first section (which equals to the first and second page) only gives a introduction. If you don't want to read this part, feel free to skip it.

First a few words about myself: I've used KDE (and Linux/GNU) for about 8 years, and would classify myself as a pre-advanced user; I like to tweak different things without touching the actual source code. One of the things I liked (and still like) about KDE was how much power it gave the user. The way you can customize the toolbars is a good example, you can move them around, change the size of the icons, add/remove icons and much more. However, the way to configure them was not always pleasant.

Here, I've tried to gather some of the problems I've experienced and also give some suggestions on how to tackle them. Please don't see this as a wishlist for KDE 4.0, but rather my vision on toolbars in future KDE versions.

Before I start, there are a few things I want to clarify:

- As I'm just a regular user, not a developer, usability expert or such, everything I write is from my perspective as a user. I don't know if my requests are possible to archive, but it doesn't hurt to share my ideas, right?
- English isn't my native language, which you probably have figured out by now. If there's anything you don't understand, please don't hesitate to ask.
- Sometimes I describe users as “basic” and “advanced”. It's a really bad way of grouping people, but this time it is for my convenience; I hope you understand what I mean.
- Most images are screenshot (some slightly modified). However, most of the figures in Section 6 are mockups.
- All graphics are copyright to their rightful owners.

## **1.2 Overview**

The different topics are categorized in the following way:

### **Section 1**

- 1.1 A brief introduction p. 1
- 1.2 Overview p. 2

### **Section 2: The toolbar handler**

- 2.1 Problems and current solution p. 3
  - Fig 1
  - Fig 2
- 2.2 Normal/customize state p. 4

### **Section 3: Right click context menu**

- 3.1 Show or not show, that's the question p. 4

### **Section 4: Add/remove items**

- 4.1 Problems p. 5
  - Fig. 3

### **Section 5: Multiple toolbars**

- 5.1 Problems p. 6
  - Fig. 4

### **Section 6: Conclusion**

- 6.1 Putting everything together p. 7
- 6.2 Getting inspiration p. 7
  - Fig. 5
- 6.3 Make it KDEish p. 8
  - Fig. 6
  - Fig. 7
- 6.4 Important details p. 10
- 6.5 Unsolved problems p. 10
- 6.6 The end p. 10

## Section 2: The toolbar handler

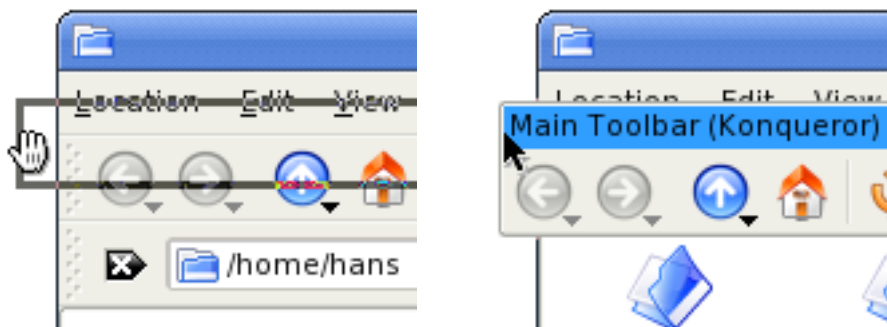
### 2.1 Problems and current solution

The `Back` action is probably one of the most used action in web browsers like Konqueror. The icon is often the first toolbar item, counting from left. However, it's not on the left edge - there's a small “handler” to the left of the icon, which is used to move the toolbar using drag and drop.



*Fig 1. Mouse pointer hovered over the toolbar handler*

While making it inconvenient for the user (Fitts' law?), it's not unlikely that someone accidentally hits the handler instead of the intended toolbar icon. S/he might even drag the toolbar to another place or undock it, which in many cases causes confusion, specially for new users.



*Fig. 2. User accidentally makes the toolbar floating using drag and drop*

In the current SVN version of KDE, I've found two things that manage this issue: Qt4 shows a nice animation when moving the toolbars, reducing the risk of an unpredictable action from the user, and there's also a `locked/unlocked` state (like the one Kicker features) which hides/shows the handlers. This is a good approach to solve the problem described before.

In my opinion, the toolbars should be locked by default. The “basic” user is often not interested in changing the toolbars at all, while the more “advanced” user probably know how to unlock them.

This raises a question: To configure the toolbar, the user have to unlock the toolbar. So if someone unlocks the toolbars, s/he probably wants to customize them, right? If that's the case, is there really a need for a locked/unlocked state?

## 2.2 Normal/customize state

How many times do you usually change the toolbars of an application? I usually do it the first time I use the application. Sometimes I want to adjust something after using the applications for a while, but in general it happens very rarely. As I said before, a basic user probably doesn't customize the toolbars at all.

My suggestion is to change the `locked/unlocked` state to `normal/customize`. The `normal` state is the same as the current `locked` in SVN. As the name implies, it's also the toolbars' default state. The real difference, apart from the new names, is the `customize` state. Not only does it make the handlers visible, it also enables you to drag the items in the toolbars and change other things such as text position, icon size etc. In other words: you use toolbar in normal state and customize it in customize state. Makes sense, doesn't it? A more detailed explanation will follow in Section 6.

## Section 3: Right click context menu

### 3.1 Show or not show, that's the question

When right clicking on a toolbar in KDE3, you were presented with the following options:

- Orientation >
- Text Position >
- Icon Size >
- Toolbars >
- Configure Toolbars...

I rarely change settings such as `Icon Size`, and when I do I only do it once; should the context menu show these settings in the normal state mentioned in 2.2? In SVN, it seems to hide the most of the options when in locked state, only showing:

- Lock Toolbars ×
- Configure Toolbars...

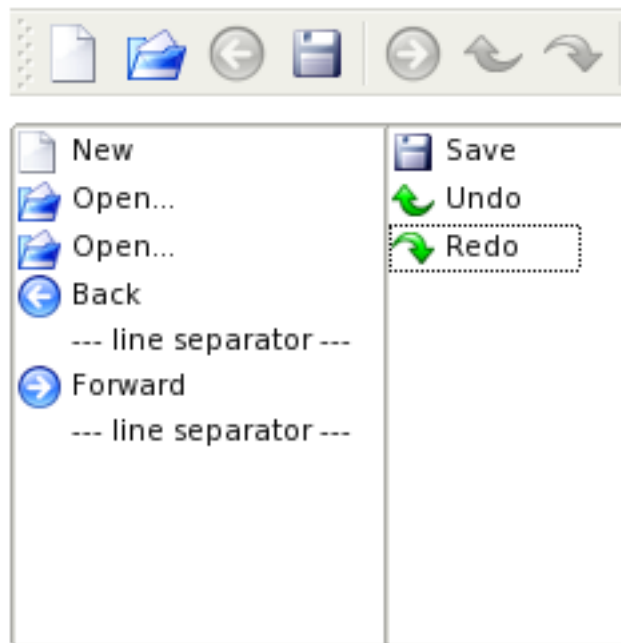
To show all the options you have to unlock the toolbars first. How should it be handled if we change the states to `normal/customize`? Read more in 6.5.

## Section 4: Add/remove items

### 4.1 Problems

This was the part I found most frustrating when I first tried to configure the toolbars in KDE applications. Not only was it a pain to add/remove/move icons (at that time drag and drop wasn't even supported), I also encountered other annoying issues.

For example, Items did not always end up where you expect them, a problem many applications still have. Please see the screenshot below:



*Fig. 3. Kate's toolbar and its current actions*

At the top you see the toolbar. The left list underneath contains the actions for Main Toolbar <kate> while the right shows Main Toolbar <KatePartView>'s actions. Now take a look at the resulting toolbar; can you spot something odd?

Another thing that made me annoyed was when I accidentally deleted an item and didn't know how to restore it. <Merge> in Konqueror is a such an item - the only way I've found to restore it is by resetting the toolbar to its default settings.

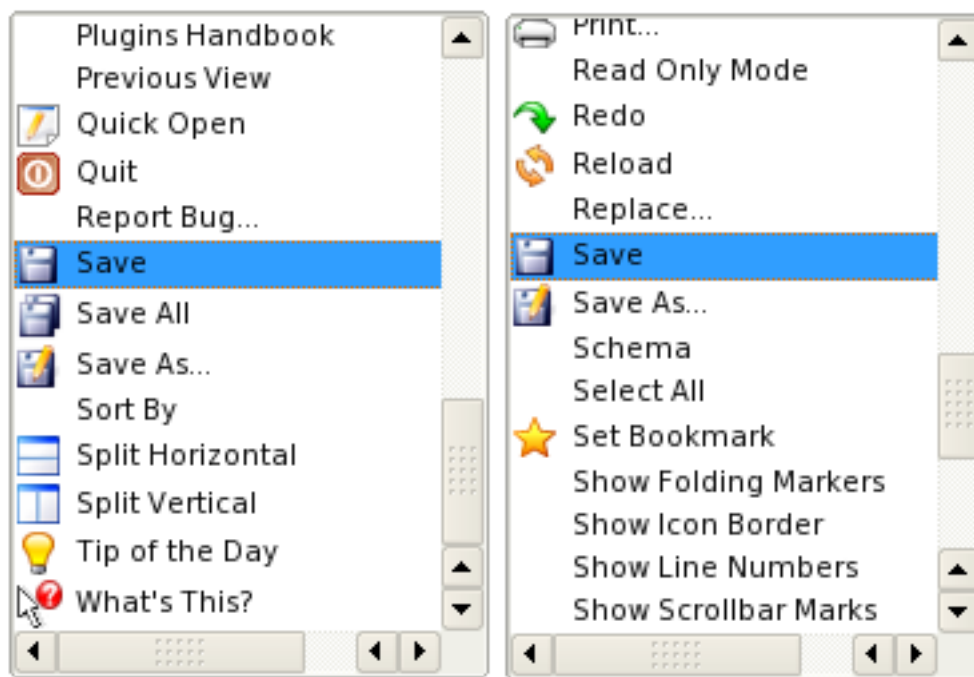
## Section 5: Multiple toolbars

### 5.1 Problems

Many applications have more than one toolbar. However, the number of toolbars you can use is limited by the developer. I don't know if this is a good or bad thing, as I rarely need any more toolbars than provided by the application. However, a related issue I've encountered is when you want to add a certain actions which is only available on a specific toolbar.

Sometimes you can't add an action to a toolbar, if you already have added it to another; it doesn't matter if the latter toolbar is visible or not. (For example the location bar in Konqueror and edit global identity widget in Kopete).

In 4.1, I wrote about a problem with the order of items in Kate's toolbar. In the `Configure Toolbars` dialog, they appear as two “toolbars”. However, the items are added to the same toolbar in the application, thus causing confusing. See the image below:



*Fig. 4. Available actions for Main Toolbar <kate> and Main Toolbar <KatePartView>*

There are two `Save` actions, and they are added to the same toolbar. However, they don't do the same thing – one saves the current file, while the other saves the current session. In this case the save actions could be renamed to solve the problem. But as a user I still wonder why these actions has to appear as different toolbars when they in fact end up on the same.

This is probably more complex from a developer's point of view, but my vision is:

- Categorize the actions in way that makes more sense.
- Don't limit certain actions to certain toolbars (think: Kicker and its panels and applets).

## Section 6: Conclusion

### 6.1 Putting everything together

How can we make a toolbar that is easy to use and easy to customize, without reducing the current options? Here are some of the problems we want to solve:

- A user who “just want to use the application” shouldn't be able to accidentally change the appearance of the toolbar.
- The right-click context menu shouldn't be filled with options you rarely change – instead, keep it simple when in normal state (see 2.2).
- It should be easy and natural to add/remove/move items. They should end up where the user expect them to, without any hocus pocus (like the *Animated Logo* in Konqueror).
- A better handling of multiple toolbars and their actions.

### 6.2 Getting inspiration

Why reinvent the wheel when there already are solutions available? There is an application I like in many ways, its way of handling the toolbars is one aspect. It is easy and “just works”, making the user satisfied.

To add items you simply drag them to the preferred place from a dialog window, and vice verse if you wish to delete it. See the screenshot below.

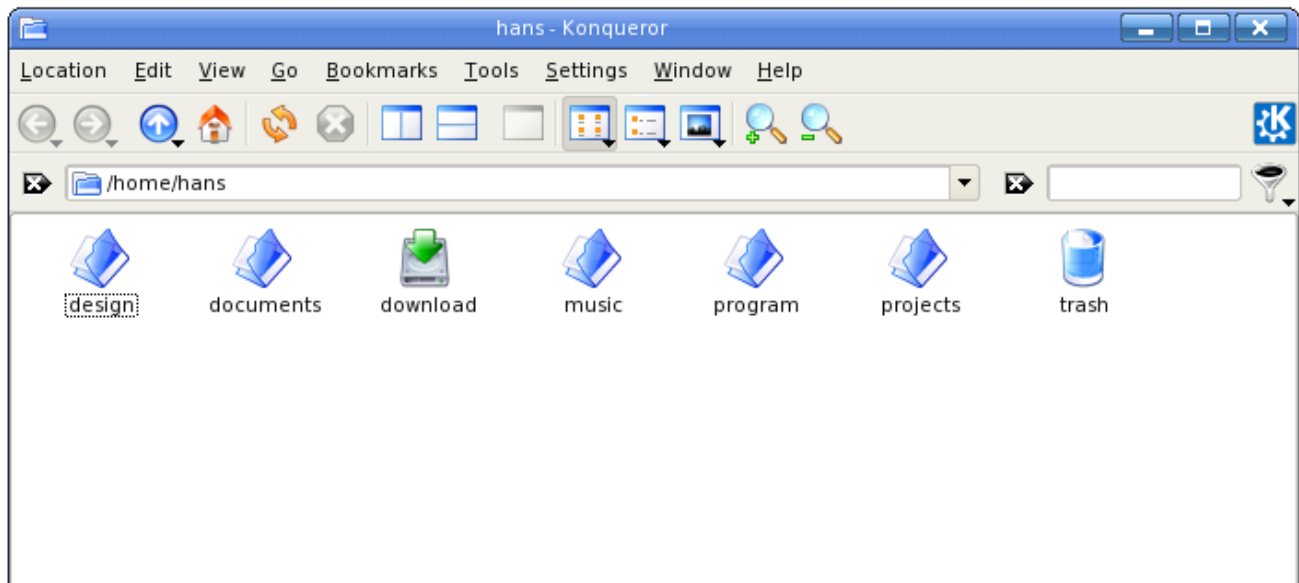


*Fig. 5. Another way of customizing the toolbars*

That's right, the application I'm talking about is none less than the web browser Firefox. As seen in the picture, you add separators the same way as actions with drag and drop. There is also a “Flexible Space”, which stretch as much as possible (like spacers in Qt Designer). This explains why Firefox' equivalent to the animated logo in Konqueror is pushed to the right edge.

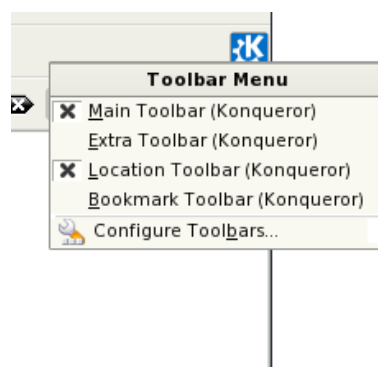
### 6.3 Make it KDEish

As an image says more than thousand words, instead of trying to explain using my poor English, I've drawn some mockups and added comments to them.



*Fig. 6. Konqueror in normal state*

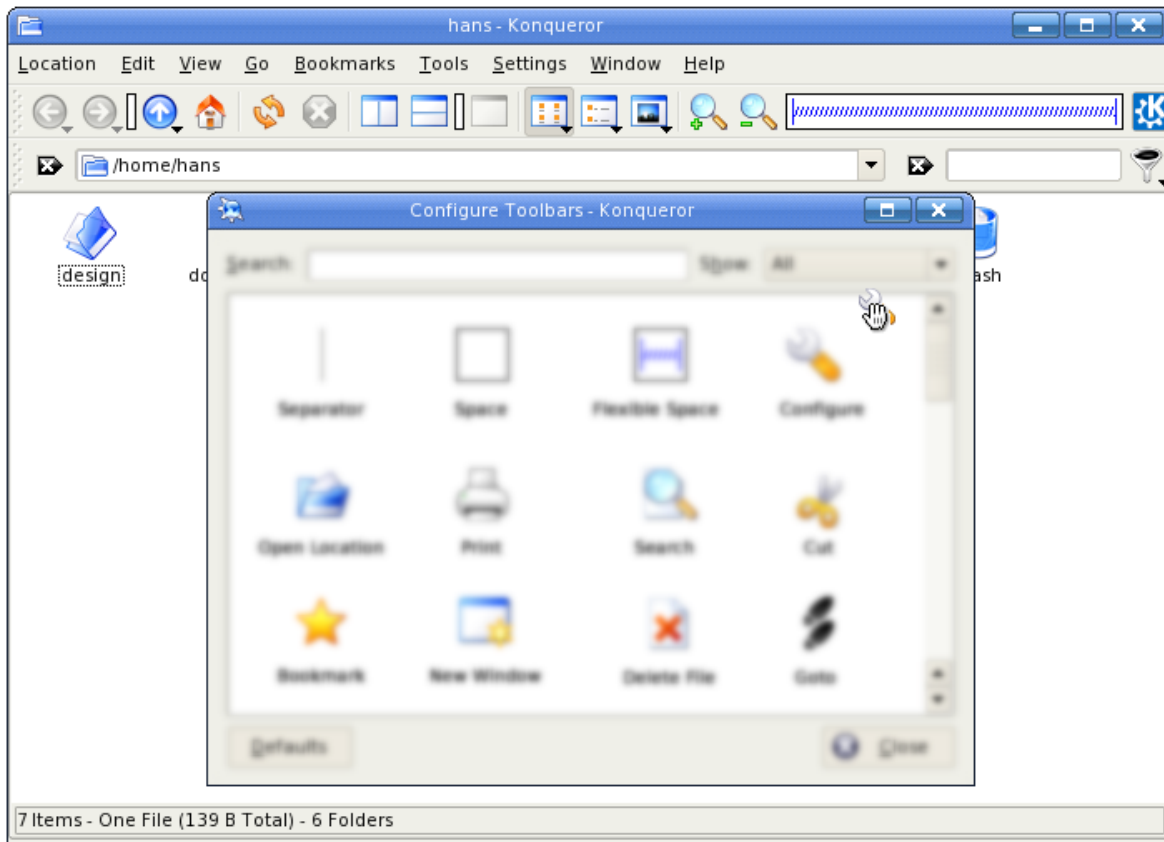
First, a screenshot that shows Konqueror in its default state. We can't change the toolbars' appearance here; first we have to enter the customizing mode.



*Fig. 7. Configure toolbars from right click context menu*

To customize the toolbar(s), right click on it and choose `Configure Toolbars...` from the context menu. You can also choose `Settings - Configure Toolbars...` from the menu of the application.





*Fig. 8. Enter customize state with full drag and drop support*

When you choose to configure the toolbars, there are many things that happens.

- The toolbars are now in customize state. You can see that the handlers now are visible, which allows you to drag the toolbars to other positions.
- The actions can't be accessed anymore. You can drag them to other positions and remove them by dropping them on the dialog or outside the toolbars.
- The separators/flexible separators (here I've renamed them to `space`/`flexible space`) are also visible. They're handled like actions.
- And finally, the dialog window. I've blurred it out because it's far from complete and I don't want you to focus too much on it. But you can see the main idea: the actions are shown as icons, and can be dragged and dropped on the toolbars.

When you're finished, and want to enter normal state again, just click the `Close/Done` (or something similar) button. (Read more about the button in 6.5).

## 6.4 Important details

As far as I know, the default text position in KDE4 will be `Text Under Icons`. While this makes the application easier to use the first time, I believe many will change it back to only show the icon to save space. However, even if it's an easy task, it'll get annoying if you have to change it for every KDE application. Instead, I would like to see an approach similar to icon size: a default value. If you want to change the text position and icon size for every KDE application, you just have to change the default value (although I don't know where it should be located in System Settings). You can of course still set specific settings for single toolbars too.

## 6.5 Unsolved problems

In Firefox, there is only a `Done` button; the changes are applied immediately. In KDE, however, you often find an `Apply` button, and also the ability to `Cancel` thus discarding the changes. In this case, I would prefer the first approach with a `Done/Close` button, but it wouldn't be a disaster with `Apply/OK/Cancel` either.

Firefox shows options such as `Text Position` and `Icon Size` in the dialog, and those are applied globally. I appreciate the ability to set individual settings for single toolbars in KDE, so I suggest to show these options in the right click context menu instead. However, that would mean that the content of the context menu would change when you enter the customize state (which it currently does when you lock/unlock the toolbar, as mentioned before); I don't know if that would be a good solution, hopefully someone else can give a better answer.

The ability to change the icon would remain the same as it is now. It just needs some polishing. For example, you shouldn't be able to change the icon of the location bar. Sometimes an action has more than one icon (`play/pause` for example). In this case I think it would be good if you could replace all icons of the action if you want.

I think a search/filter function could be handy, but the way of categorizing the action is in my opinion more important. However, I don't know how they should be categorized, so I leave this question to open for someone else to respond.

## 6.6 The end

While I've focused on an easy and powerful way of customizing the toolbars, I think the most important thing is that the default settings are good. This is something I hope to see in KDE4 – sensible default settings.

I want to thank you for reading this, and I hope you didn't find it a waste of time. If you have questions, ideas, comments etc. I'm open for discussion.