Ideas for Koffice 2 GUI and Functionality

by Sandis Neilands
sandisn@gmail.com

What's Wrong With The Current Generation of Office Suites?

User Interface

Historically office suites contain word processor, spreadsheet and presentation program. Although they are separate programs, that deal with different types of documents, they share some functionality. Unfortunately quite often the user interface is slightly different in each of the applications. This is bad because user has to learn multiple interfaces to perform the same action. Another problem is that some tools differ in each application.

Long/Complex Documents

If you have ever created long or very complex document you already know the shortcomings of today's (or may I say yesterday's) office suites. For example, how do you select only paragraphs with red background? And how do you convert your presentations to text documents without loosing all the nice formatting and layout you have created? And of course, navigating in such documents with nothing but a couple of scrollbars and mouse wheel is a pain.

Multi User Environment

Situations where one document has multiple authors are becoming more and more common. Unfortunately most office suites doesn't respect that. In most cases it's impossible to see who did what with the document. Even worse – if you sent your thesis to all your student peers for a review you'll probably get back something like hundred heavily modified documents. Good luck comparing and merging proposed changes!

My Vision

One Program for All Kinds of Documents

Yes, I actually think that this is possible. Usually in a typical office document we have text frames (headings, paragraphs, lists, etc.), tables, a couple of images and maybe some other objects (like charts and formulas). The thing is that formatting and styling is pretty similar in all parts of the office suite. So why do we have three different programs then? Each of those programs contain some object specific tools that aren't normally accessible in other programs. However I think that it's possible to carefully port those tools to one central program. Think of a slide show menu in Kpresenter. Wouldn't it be great, if we could see all of our documents as slide shows? To have a grid (similar to one in Kspread) in text document wouldn't hurt too.

Media Independent Documents

With media I mean A4, screen presentation, web page, wiki page, relief embossing, screen reader, etc. Basically media is set of styles and capabilities (does this media support audio output, motion, colours, images?). Changing media is a matter of adjusting styles, object behaviour and doing object conversions (such as text -> relief embossing). All of this, of course, should be done automatically.

Program for Long/Complex Documents

Something has to be done to help those struggling with long and complex documents. Formatting and structuring them can be very difficult and time consuming, but it shouldn't be that way. Even long and complex documents should be structured and styled with ease. Thus I propose the browser – a tool that allows searches based on document's content and formatting.

Multi User/Version Documents

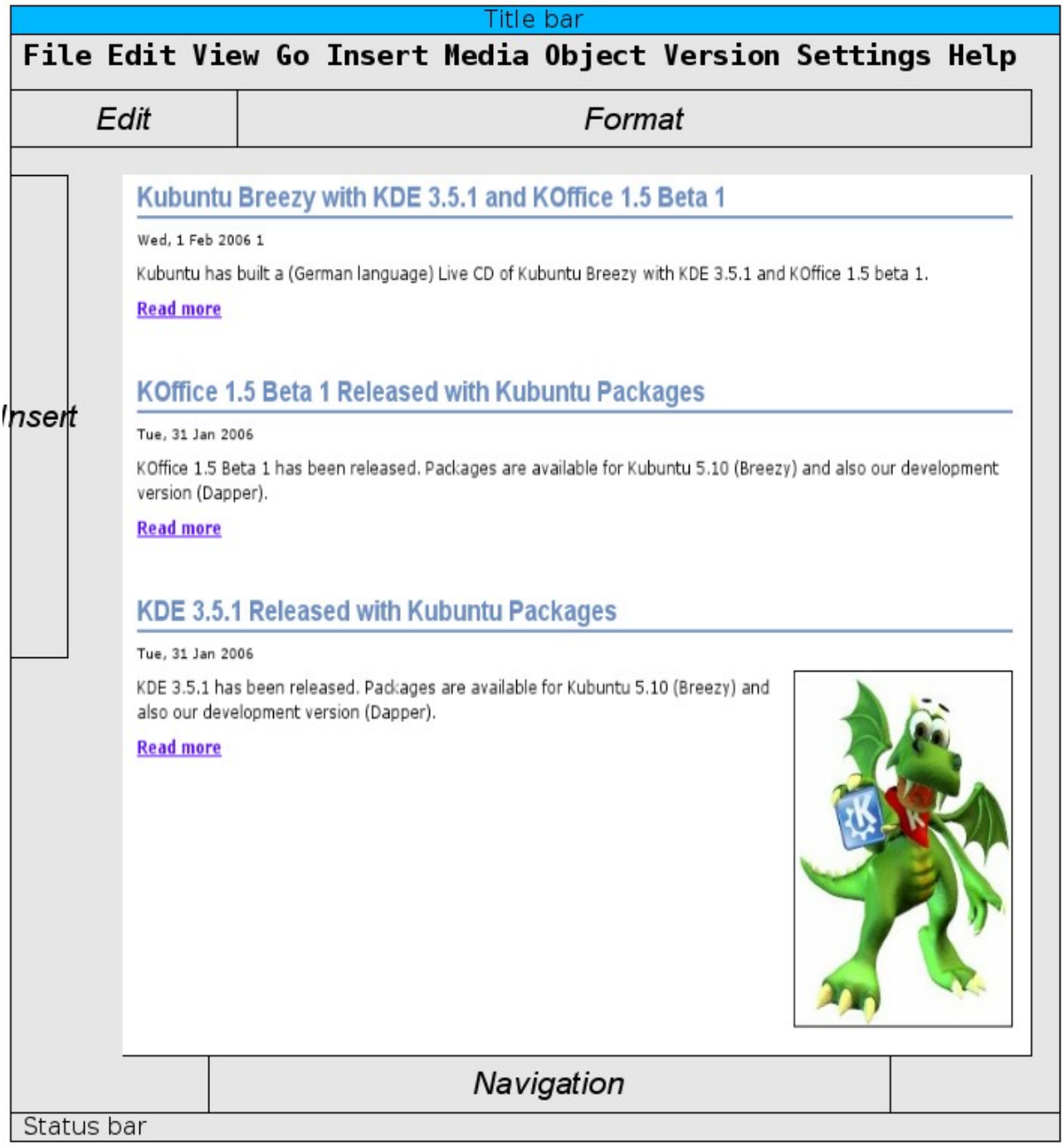
The office program should play nicely in multi user environments. It should provide ways to review and merge proposed changes in both – structure and formatting.

Easy Navigation

Navigating in document shouldn't be harder than navigating in website.

GUI Mock-ups and Comments

Main Window



Menu Bar

As you can see the menu bar contains some previously unseen items - “Go”, “Media”, “Object”, “Version”. The “Go” and “Media” are pretty straightforward. “Object” item contains tree sections – tools, format and notes. This menu changes it's content depending on what kind of object is selected. For example if you select a table cell then “Goal seek” becomes active. Basically this menu is what you get by right-clicking on some object.

“Versions” has items related to recording and merging changes, setting document's version, etc. Of course, if you need it, “Versions” tool bar is few click away.

Tool Bars

The most interesting from these is navigation tool bar. It has “<<”, “<”, “>”, “>>” buttons and a drop-down menu with following members – page (or slide depending on media), object of the same type, object with the same formatting (or style), proposed change, “search box” (you can enter your search phrase there). The behaviour of arrow buttons changes depending on selected drop-down menu's item. For instance, if you have selected “object of the same type” in drop down menu and you have selected image in the document canvas then the “>” button will take you to the next image.

There is also a zoom slider (very similar to volume adjustment tool) and “Preview” button. When user hits “Preview” button current page zooms out far enough so that other pages are visible and can be focused.

Browser – Objects Tab

Perhaps the most interesting part of my GUI mock-up is the browser. It is designed to help users structure and format their documents. It could as well be a detachable sidebar, but I like separate windows. I will try to illustrate what you can do with browser by example.

Imagine you wanted to change colour and decoration of every link that says “Read more”. Traditionally you would search for that string, then apply the new style. The browser provides a way to do it faster and more convenient. It has filter which filters as you type. Entering “Read” would show all objects containing that string and hide everything else. Now you either Ctrl + click the objects that you want or just select them by mouse. Selecting object in browser selects it in document canvas too. After you have selected all you want you either select Styles tab in the browser or do your formatting within main window. If you chose Styles then you have another filter at your fingertips to find that one little link which is red. Of course you can do formatting from there too. You can also edit content in Objects tab.

The browser remembers, what's selected in the other tabs so if you select some particular style in the Styles tab, only objects with that style are shown in Objects and Events tabs.

Why all the hierarchy? Simple – if you suddenly realize that KDE 3.5.1 article in fact has to be closer to the top of the page than KOffice 1.5 Beta 1 article, you just select it with all of it's contents and drag it in the correct place. If you added a caption to the Konqi image, but later decided that you no longer want to keep Konqi and deleted it, the caption would be deleted as well.

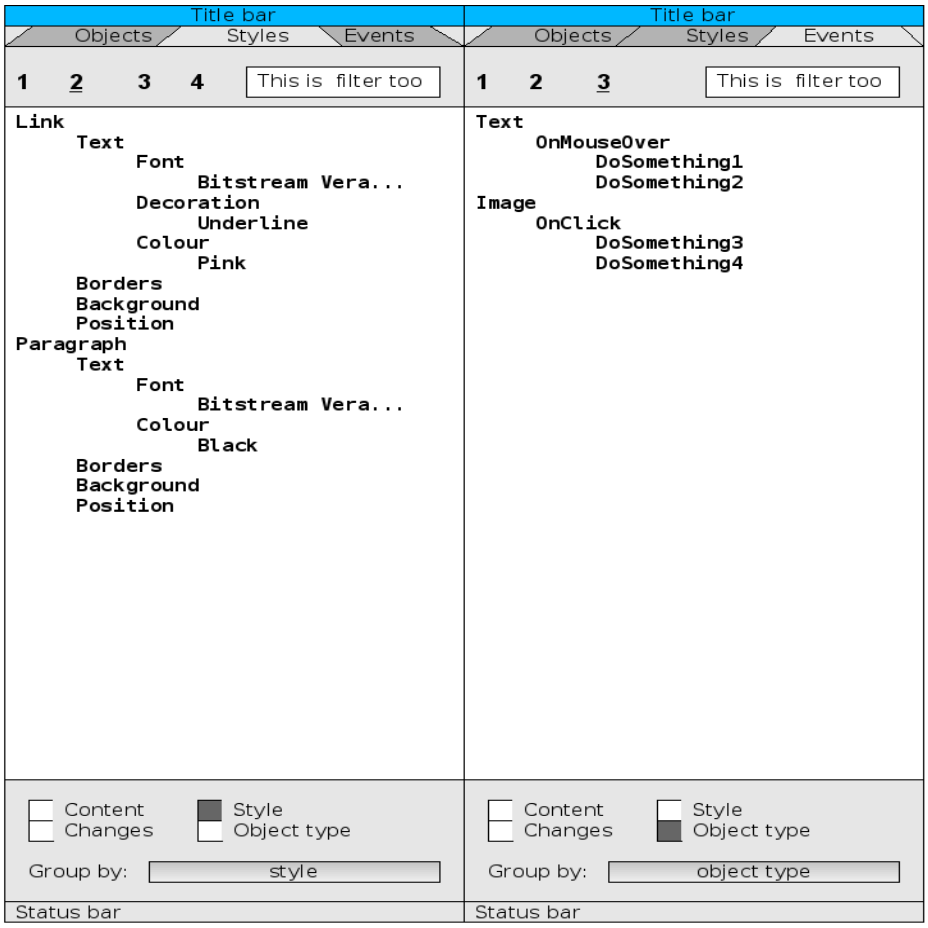
The numbers (could be implemented as horizontal slider) represent your browsing level. If you want to see just headlines, you slide it to 1. If you are interested in some particular article you can right click it -> expand. 0 could be reserved for media (pages, slides), but since web page has no limitations, you can't see it now.

At the bottom there are some optional check boxes and drop down menu. If you are looking for images, you group by object type (drop down menu), then check “Object type”. In filter type “Image” and every other object type disappears. Then you either use the slider or right click -> expand to find the image you are looking for.

Browser – Styles and Events Tabs

Don't let the boring fonts fool you. Browser could be implemented with very nice buttons, drop down menus, etc. The events tab should be accessible only if the media supports it.

Having thought a bit I think that the browser actually needs another tab – Notes. If it's possible to attach notes to the objects, then it should also be possible to easily find those objects and perform searches on the content of those notes.



Use Case

Book (ABC)

John is writer. Currently he works on ABC book for very small children. To make sure that everything is understandable and that his book is accessible for disabled children too John plays a lot with formatting. In the past it has been very difficult, but now he uses Koffice 2 with browser and everything is fine. For example, when he writes a new page for letter K he can easily select all K's in that page by filtering with “k AND K” at level 5 and then just selecting and formatting the results.

He has sent the version 0.92 of his book to editor and his mother for content review (only changes in content are recorded). Since then he is recording his own changes too. When he got the documents back he merged them with his own, clicked “proposed changes” in the navigation tool bar's drop down menu, then “<<”. The first proposed change is a correction of spelling error. John happily presses “Approve change” button and goes on (“>”)...

When John was asked to produce a version in relief embossing because there were some blind parents who otherwise couldn't help their children to learn reading, he added the new media via Media ->Add. Then Media ->Change ->Relief Embossing. Text was nicely converted and all the shiny and colourful images had gone (leaving alt'ed text).

Expense report

Joe is preparing an expense report. He enables “Show grid” in View menu. From now on his work flow is very similar to one he had developed while working with the previous versions of Kspread. The only exception is that cells expand as he writes, they resize themselves to fit inserted images, etc. When he hides grid, program automatically splits objects into text frames, tables and images. If the report is complex Joe can use browser to develop object hierarchy and format his document.

Typical Work Flow

Short document

For those who use office software to create X-mas cards there wouldn't be much of a change. This is not necessarily a bad thing – forcing a new way of working upon them wouldn't be very nice at all. On the other hand, even they could squeeze some functionality out of the browser, object integration and media independence.

Long documents

1. Work on content, structure it.
2. Format using browser.
3. Review changes made by other authors. Then merge back.