



## **Heuristic evaluation of Kaffeine Player**

Brief study to learn about possible improvements in the user interface

**Author** Chi Shang Cheng

**Date** 11 July 2006

---

## Abstract

After a short heuristic evaluation of Kaffeine, a total of 40 usability issues were found. Only 5 issues of high severity were found, 20 were marked medium, and the remaining 15 issues considered lowly severe.

Most of the issues were related to unnecessary complexity of the user interface. The application was not tested for task-oriented usability. Even though the application functioned properly from a technical perspective, attention should be given to a more aesthetic user interface design in the future.

---



## 1. Introduction

Aside from digital music, there is another multimedia need that a lot of computer users require: digital video. Kaffeine Player fulfills the multimedia needs of the KDE users. Although this application is not registered on OpenUsability<sup>1</sup>, it was still reviewed (on behalf of the KDE Usability Project<sup>2</sup>) because of its presence in popular Linux distributions such as Mandriva Linux<sup>3</sup>, SuSe Linux<sup>4</sup> and Kubuntu<sup>5</sup>.

The purpose of this study was to find possible improvements in the graphical user interface of Kaffeine. A specific usability inspection method was chosen for this study: the heuristic evaluation, which will be discussed in detail later on.

The application has only been roughly reviewed. Most parts are only glanced, other parts weren't examined at all, such as the icons. This leaves material to be examined in the future.

Please note that during the evaluation no users were involved. Some issues are merely assumptions, but do have some ground based on the heuristics. Some issues could be tested for correctness by conducting a user-based usability test with the corresponding use cases.

---

<sup>1</sup> <http://www.openusability.org>

<sup>2</sup> <http://usability.kde.org>

<sup>3</sup> <http://www.mandriva.com/en/individuals/products/2006discoverylx/packages32>

<sup>4</sup> <http://www.novell.com/linux/suse>

<sup>5</sup> <http://www.kubuntu.org>

## 2. Method

Heuristics, also called guidelines, are general principles or rules of thumb that can guide design decisions (Nielsen, 1990). Heuristic evaluation allows you to catch problems that task-oriented methods, e.g. cognitive walkthrough, would miss. The procedure is based on the observation that no single evaluator will find every problem with an interface, and different evaluators will often find different problems.

Not all problems will be found with this method. It is possible to detect all major problems within an interface that are "heuristically identifiable" with 3 to 5 usability experts, but they can catch 75 percent of the total heuristically identifiable problems. That is, problems with the interface that actually violate one of the ten heuristics (Nielsen, 1994).

### 2.1 Heuristics

Here are ten heuristics from Better Desktop (Heuristics, 2006), a usability project sponsored by Novell. These are similar to the heuristics Nielsen (Nielsen, 1994) comprised from 249 usability problems:

- **Consistency**  
The design is consistent with HIG and the direction of the Gnome community.
- **Minimal Design**  
The design does not include extraneous controls, messages, dialogs, etc. It keeps things as simple as possible and limits number of new UIs that appear for tasks.
- **Limit Memory Requirements**  
The design emphasize recognition instead of recall; its options are obvious.
- **Constructive Error Handling**  
The software provides feedback, ways to escape, and ways to restore defaults. Its error messages are as sane and helpful as possible.
- **Task Based Design**  
The software facilitates task accomplishment. In other words, learning the software should not be a task in itself.
- **Appropriate Language**  
The software's words match with the real world and are in concert with the users' expectations and experience.
- **Provide Help and Docs**  
The software makes accessing documentation and help easy.
- **Accessibility**  
The design is accessible to users all along the continuum of human physical ability.
- **Target Audience Appropriate**  
The designs puts the needs and experience of its target audience at its core.
- **Feedback**  
The software provide clues to the user of what the computer is doing. This is especially important when the user has caused an action to occur.

## 2.2 Test setting

The heuristic evaluation was conducted on a PC running Kubuntu Linux 6.04<sup>6</sup>. Kaffeine is part of the base system and I have tested version 0.8.1. I have applied the Polyester theme<sup>7</sup> and the nuoveXT icon set<sup>8</sup>.

## 2.3 Procedure

As already mentioned, this usability test doesn't involve users. The user interface will be examined by looking for compliancy with the heuristics described above. In addition, I will be using other (scientific) literature where needed.

Every design issue will be formatted in the style of a KDE usability report (Ellsworth, 2003), to enhance readability. This format will also make reproducing and correcting the problem easier.

Note that there will not be a discrimination whether a usability issue is related to KDE, Kubuntu or Kaffeine itself.

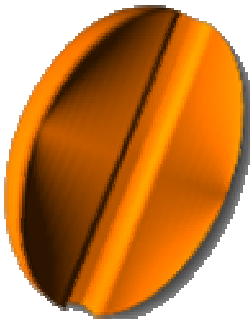
## 3. Results

### 3.1 Use of name suggestive icon (medium)

The Kaffeine application icon is a pictorial representation of a coffee bean, thus it's suggesting the program's name instead of the program's purpose.

Use an icon that suggests the functionality of the application.

Name suggestive icons are less desirable than functional suggestive ones, because an extra layer of abstraction is added (Benson et al., 2004).



**Figure 1** The Kaffeine icon

### 3.2 Extraneous information in Start tab (low)

There are two text labels placed in the top of the Start tab. The top one is the application name, the other one is the version number. Both labels do not display useful information to the user, since it is somewhat redundant. The version number can be useful, but it is not necessary to display it this prominently.

Remove the application name and version number headings.

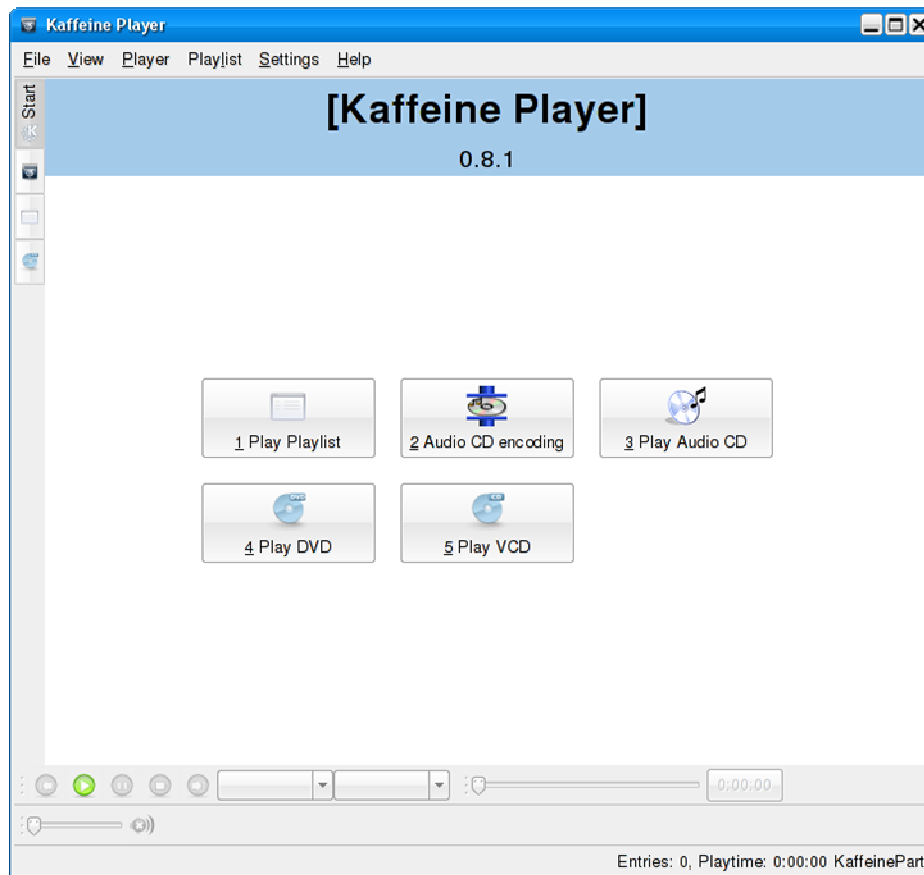
Keep the user interface as clean and minimal as possible.

---

<sup>6</sup> <http://www.kubuntu.org>

<sup>7</sup> <http://www.kde-look.org/content/show.php?content=27968>

<sup>8</sup> <http://nuovext.pwsp.net>



**Figure 2** The main screen after starting Kaffeine

### 3.3 Extraneous information in status bar (medium)

In the status bar of the Kaffeine window, the amount of playlist entries, total play time and the word 'KaffeinePart' are shown (figure 2). The use cases for constant awareness of the presented information in the status bar are not very strong. First of all, the user has to make use of the playlist. Secondly, the word 'KaffeinePart' does not make much sense. Thirdly, the playlist information should be placed in the Playlist tab, which gives the information more context.

Remove the status bar.

Keep the user interface as clean and minimal as possible.

### 3.4 Unnecessary use of button for time display (medium)

The time display, placed next to the progress bar, is a button instead of a common text label. The use of the button interface element allows the user to toggle between display of time elapsed and time remaining (figure 2). However, it is not necessary to have this setting displayed all the time. Especially, because a button is used which introduces more visual complexity.

Use a text label instead of a button for time display.

Keep the user interface as clean and minimal as possible.

### 3.5 Ambiguity in Behavior settings dialog (medium)

In the Configure Kaffeine dialog, the user is able to set the option of pausing playback when the player window is hidden. However, it is not clear what 'hidden' refers to. The playback only pauses when the window is minimized, even though it could be 'hidden' behind another window.

Change the text to something like 'Pause playback when windows is minimized'.

Speak the user's language.

### 3.6 Extraneous information in buttons of Start tab (low)

The buttons in the Start tab are ordered with numbers (figure 2). However, the numeration of the actions has no function whatsoever. It suggest an order or hierarchy that does not exist.

Remove the numbers from the buttons.

Keep the user interface as clean and minimal as possible.

### 3.7 Technical information in Miscellaneous Options dialog (medium)

In the Configure Kaffeine dialog, specifically the Miscellaneous Options, the user can choose the encoding for the Meta tags (figure 3). This is highly technical information, because it requires knowledge about language encoding and metadata. Novice users can be overwhelmed with this kind of information.

Either remove this option and choose a default setting that works best for most users, or place this setting under 'Advanced Options'.

Speak the user's language. Try to maintain a consistent easy to use interface and apply techniques like progress disclosure to hide technical information from the novice user.

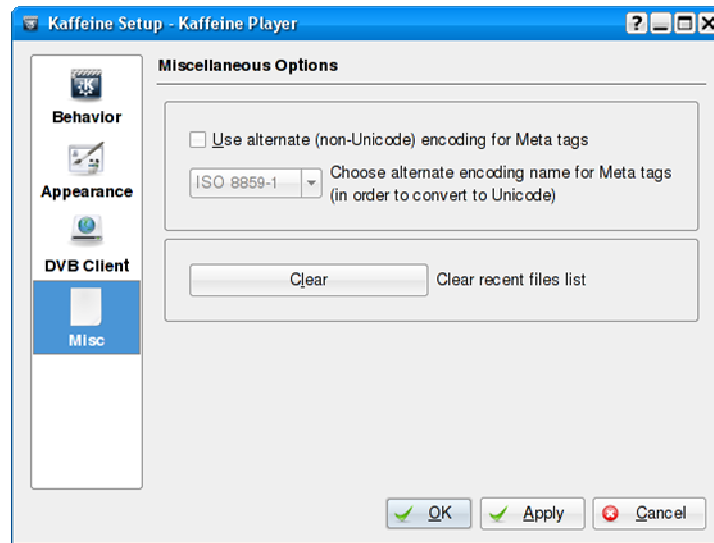


Figure 3 Configure Kaffeine dialog: notice the difference in sophistication between the two options

### 3.8 Unable to return from Minimal mode from the context-menu (high)

There are three view modes available in Kaffeine. First of all, there is the default normal view mode. Second, the user can also switch to fullscreen mode enabling the possibility to use his entire screen to watch a video. Third, the user can also switch to minimal mode and when enabled, Kaffeine will hide all player controls, except for the window and the video display. This mode can be enabled by pressing the shortcut key 'M', but it can also be accessed from the View menu. However, the user cannot return to normal mode from minimal mode without pressing 'M'. In other words, users have to remember the shortcut key in order to use this function. Also, there is no option to return to normal mode from the context-menu.

Add mouse-over controls, that is for example a button that pops up when the mouse cursor moves over the video. This way the controls are hidden when the mouse cursor is not moved by the user. This return command can also be added to the context-menu.

Limit memory requirements of the user.

### 3.9 Unnecessary display of unavailable subtitle languages (medium)

When playing a DVD, the user can switch to different subtitle languages from both the DVD menu and the Kaffeine controls (figure 4). Kaffeine has drop-down menus placed next to the playback controls

for subtitle and audio languages (figure 2). However, besides the available subtitle languages there are also some placeholder items for possible languages that come with the DVD. Remove the unavailable subtitle languages from the drop-down menus. Keep the user interface as clean and minimal as possible.

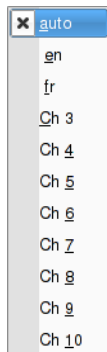


Figure 4 Selecting subtitle language

### 3.10 Unnecessary display of unavailable audio languages (medium)

When playing a DVD, the user can switch to different audio languages from both the DVD menu and the Kaffeine controls (figure 5). Kaffeine has drop-down menus placed next to the playback controls for subtitle and audio languages (figure 2). However, besides the available audio languages there are also some placeholder items for possible languages that come with the DVD. Remove the unavailable subtitle languages from the drop-down menus. Keep the user interface as clean and minimal as possible.

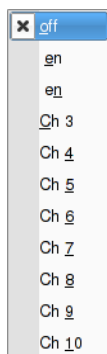


Figure 5 Selecting audio language

### 3.11 Non-contextual commands in context-menu (medium)

The context-menu has nineteen items, divided into eight groups (figure 6). The last group (Video Settings, Equalizer and Effect Plug-ins) is the least contextual group of the context-menu. Aside from that, these commands are likely to be rarely used.

Remove this group of menu items from the context-menu.

Limit memory requirements of the user and keep the user interface as clean and minimal as possible. Only keep often used and context relevant commands in the context-menu.

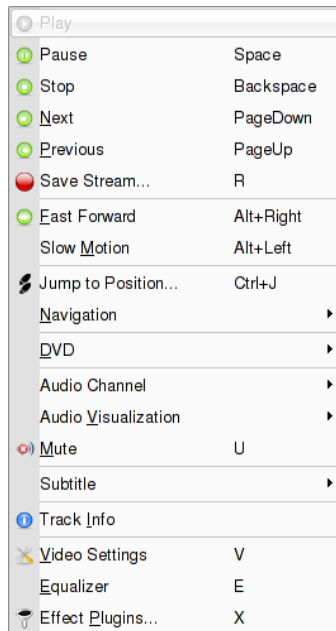


Figure 6 Context-menu of the video display

### 3.12 Auto-resize causes some toolbars changing position (low)

Kaffeine automatically resizes the window accordingly to the video resolution. If the video is small, the window will change to accommodate this size. This can cause toolbars to change position. When this happens, the height of the window increases because there are two lines of toolbars. The result is less contrast between the video and the toolbars and thus, the video gets visually less attention.

Keep the toolbars at fixed positions.

Not having interface elements at fixed positions, can diminish the predictability of UI elements.

### 3.13 Speaker icon represents wrong information (medium)

The mute/unmute button suggest an action, namely muting the audio. However, most multimedia player such as Microsoft Windows Media Player<sup>9</sup> and Totem Video Player<sup>10</sup> have the mute/unmute button to suggest it's state, rather than the action.

Conform to the concepts that are most commonly used, so in this case the button should suggest the state of the audio.

Research has shown that when 80% or more of big sites do things in a single way, then this is the de-facto standard and you have to comply. Only deviate from a design standard if your alternative design has at least 100% higher measured usability (Nielsen, 1999).

### 3.14 No presets available for Video Settings or Equalizer (low)

The user can finetune video and audio playback. Unlike most equalizers, Kaffeine does not have presets available. The user is also unable to save a certain video setting.

Ship presets for video settings and equalizer and allow the user to create presets.

Limit memory requirements of the user.

<sup>9</sup> [http://en.wikipedia.org/wiki/Windows\\_media\\_player](http://en.wikipedia.org/wiki/Windows_media_player)

<sup>10</sup> [http://en.wikipedia.org/wiki/Totem\\_\(media\\_player\)](http://en.wikipedia.org/wiki/Totem_(media_player))

### 3.15 Technical information in Deinterlace Quality dialog (low)

The deinterlace settings in the Deinterlace Quality dialog are explained in detail (figure 7). The large amount of text makes the dialog look less appealing.

Remove the detailed explanation and save these for documentation, since this information is only intended for technical users. Tooltips can also be used, but the text must be shorter then.

Keep the user interface as clean and minimal as possible.

### 3.16 Bad alignment in Deinterlace Quality dialog (medium)

The user can switch between six deinterlace settings in the Deinterlace Quality dialog (figure 7). There is a slider widget to switch between the different settings which is accompanied with an elucidation on the right. However, the text is not properly aligned with the corresponding tickmarks.

Align the text properly with the corresponding tickmark positions.

By making use of the Gestalt principle 'alignment', it is easier for the user to distinguish related items (McCracken, 2004).

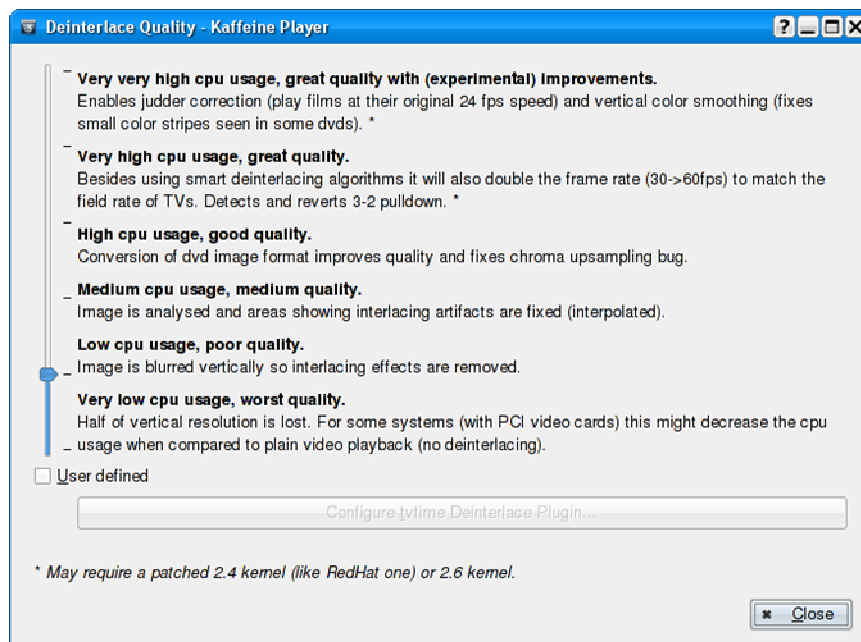


Figure 7 Deinterlace Quality dialog

### 3.17 Display of file types unassociated with Kaffeine in Open File dialog (low)

In the Open File dialog, the user is able to choose from a list of files filtered on supported media formats (figure 8). However, it also shows files that are not supported by Kaffeine, such as PNGs.

Only show files that Kaffeine is able to open by default. There should of course be an option to view files with other file extensions.

Keep the user interface as clean and minimal as possible.

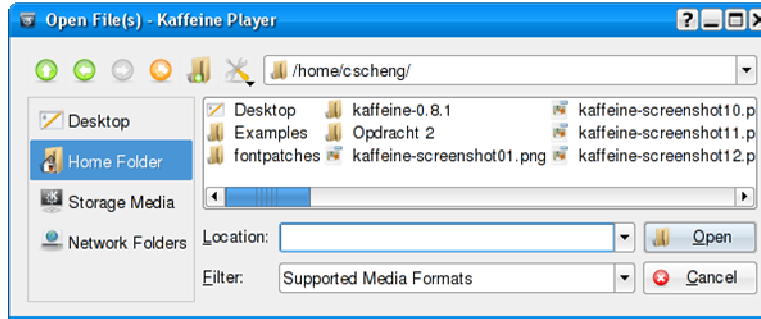


Figure 8 Open File dialog

### 3.18 Inconsistent naming of Open File dialog (low)

The user can open a file from the Open File command which can be found in the File menu. When one clicks on that command, a dialog appears from where the user can select a file to open. However, this dialog is titled 'Open File(s)' (figure 8).

Change the respective menu item to 'Open File(s)'.

Maintain a consistent user interface. Try to keep the same designations for certain commands and concepts.

### 3.19 Technical term used in Audio Encoding dialog (medium)

When the user wants to rip a CD, the user has to select the audio tracks manually and then hit the 'Encode' button. This word might be a bit too technical for users, since it is not part of everyday language.

A simple button called 'Start' will suffice.

Speak the user's language.



Figure 9 Audio encoding

### 3.20 Unable to select multiple tracks in Audio Encoding without Control/Shift-key (high)

When the user wants to rip a CD, the user has to select the audio tracks manually and then hit the 'Encode' button (figure 9). However, the user can only select multiple tracks by holding the Control-key (or Shift-key).

Place check-boxes left of the tracks, so the user can select multiple tracks without knowing that the Control-key is required. Select all tracks by default.

Limit memory requirements of the user. Keep the user interface as intuitive as possible.

### 3.21 Unexpected command to return to normal speed from slow-motion (medium)

The user is able to slow down video playback from the Player menu (figure 10). However, to return to normal speed, the user has to click Fast Forward. There is no Normal speed menu item available.

Add a submenu for playback speed which contains the necessary commands for slow motion, fast forward and the ability to return to normal speed.

Limit memory requirements of the user.

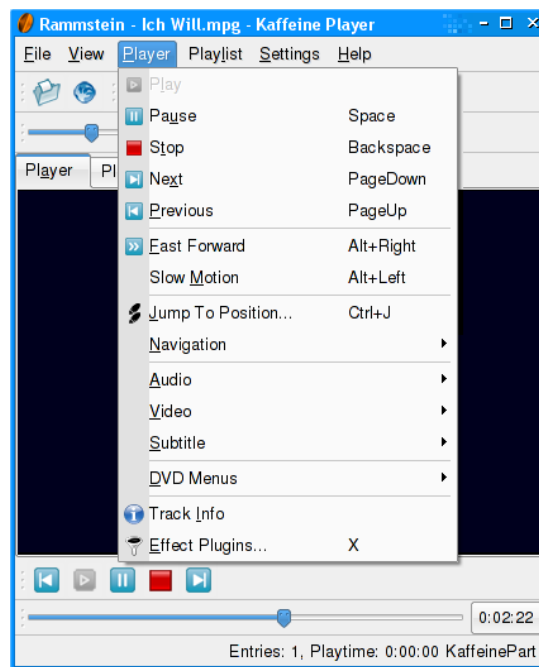


Figure 10 Player menu

### 3.22 Playlist not scaling well with resizing of window (high)

Kaffeine automatically resizes to the resolution of the video. This behavior causes interface elements to resize accordingly, but some elements do not scale well (figure 11). The Browser pane in the Playlist tab gets very small as well as the thumbnail (now playing) view. The text in the playlist becomes unreadable, because there is no more space left.

Use an external browser dialog and remove the thumbnail view to save more space for the playlist pane.

Keep the user interface as clean and minimal as possible.

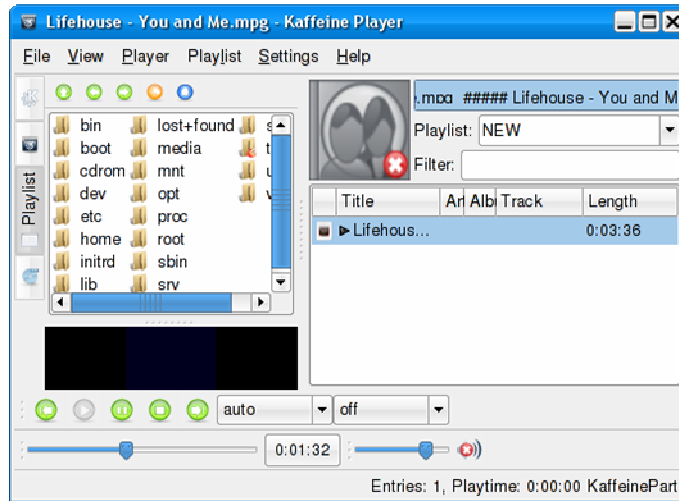


Figure 11 Resizing to a small video

### 3.23 Unnecessary inclusion of subtitle/audio language drop-down menus in Controls toolbar (medium)

In the Controls toolbar, the user can select subtitle/audio languages from drop-down menus. These interface elements are only relevant for DVD playing. However, these can also be accessed from the DVD menu, Player Menu and context-menu. Still, the drop-down menus are displayed by default. Remove them from the Controls toolbar.

Make use of knowledge the user has. Commit to these mental models. Keep the user interface as clean and minimal as possible.

### 3.24 Unnecessary usage of Now Playing banner in Playlist tab (medium)

There is a banner placed in the Playlist tab which shows the current track that is now playing, or is 'loaded', in other words when the user hits the play button that particular song will play. However, this information is already available from both the playlist as well as the window title. The limited space for the interface elements is therefore not effectively used and this problem rises when the player window auto-resizes to a low-resolution video, e.g. VCD quality MPEG (figure 11).

Remove this banner from the playlist tab.

Keep the user interface as clean and minimal as possible. Use the limited screen size to communicate both effectively and efficiently with the user.

### 3.25 Unnecessary display of Player Window thumbnail in Playlist tab (medium)

There is a thumbnail view of the current playing video in the Playlist tab (figure 11). Paragraph 3.22 already showed that the space in the Playlist tab should be efficiently used. However, the thumbnail view is extraneous, since it is not relevant to tasks concerning the playlist.

Remove the thumbnail view from the Playlist tab.

Keep the user interface as clean and minimal as possible.

### 3.26 Inconsistent naming of Track Info dialog (low)

The user can view technical information about a video or music track that is currently playing. This functionality can be accessed from the Player menu and context-menu. The command is called 'Track Info'. However, the dialog that opens is called 'Information'.

Change the text labels to match.

Maintain a consistent user interface. Try to keep the same designations for certain commands and concepts.

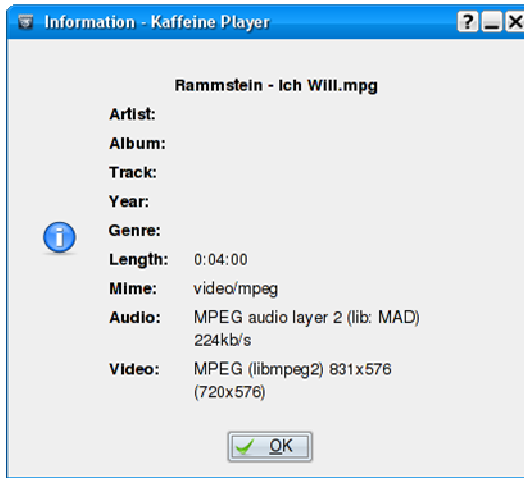


Figure 12 Track information

### 3.27 Track Information displays extraneous information (low)

The user can request track information for both music and video. When viewing the track information about a video, the user is shown types of metadata that are not available for video files.

Remove this extraneous information from the dialog.

Keep the user interface as clean and minimal as possible.

### 3.28 Unable to enter position with numeral keys in Jump to Position dialog (medium)

Kaffeine enables the user to jump to a user-defined position in playback (figure 13). The widget that is used to enter the time is a spinbox. Unfortunately, the user is only able to enter the time from the up and down buttons.

Enable the user to enter the time using the numeral keys.

Even though it is already possible to enter the time from the current dialog, experienced users should be given flexibility to use different forms of interaction [Nielsen, 1993].

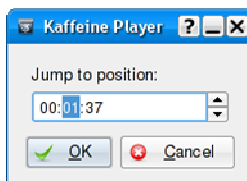


Figure 13 Jump to Position dialog

### 3.29 Low affordance of option in Configure Kaffeine dialog (medium)

Kaffeine can embed itself in the system tray. This option in the Configure Kaffeine dialog does not state the reason or practicality for enabling this (figure 14).

Remove this option, because its limited usefulness does not outweigh the drag for closing an application that is embedded in the system tray.

Keep the user interface as clean and minimal as possible.

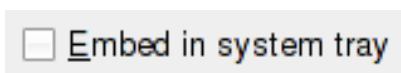


Figure 14 Embed in system tray

### 3.30 Unexpected pausing behavior when minimizing Player Window (high)

Playback pauses when the user minimizes the Player Window. This behavior is default in Kaffeine. However, it contrast with settings in other often used multimedia players, such as Windows Media Player and Totem Video Player.

Change the default behavior to not pause playback on window minimization.

Users with high motivation should feel the power of the application immediately. This also applies to users with low motivation and mandatory use, but they should also feel in control (Mayhew, 1992).

### 3.31 Unnecessary New Folder button in Open Directory dialog (low)

The user is able to add a new folder using a button in the Open Directory dialog (figure 17). However, this functionality will probably be not used, since creating a new directory is no close association for creating and opening a directory. Use cases might be opening multiple directories at the same time or adding more hierarchy to the file system, but they are not very strong.

Remove the New Folder button.

Keep the user interface as clean and minimal as possible.

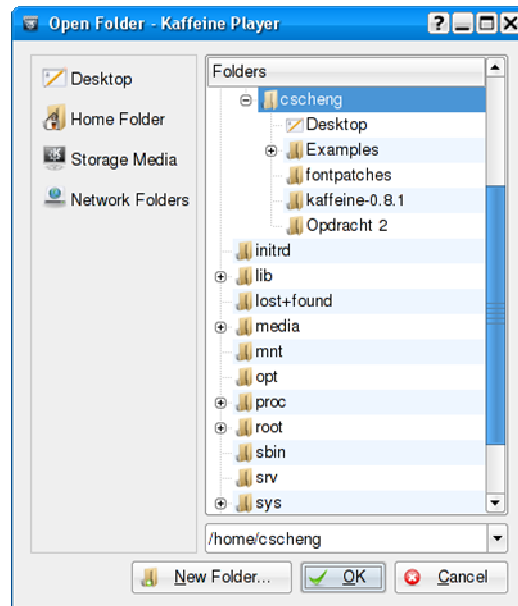


Figure 15 Open Folder dialog

### 3.32 Unnecessary display of horizontal line in Open URL dialog (low)

There is a horizontal line placed in the Open URL dialog (figure 16). This is not necessary, because use of spacing can separate (groups of) visual interface elements by using the Gestalt principle proximity (McCracken, 2004).

Keep the user interface as clean and minimal as possible.

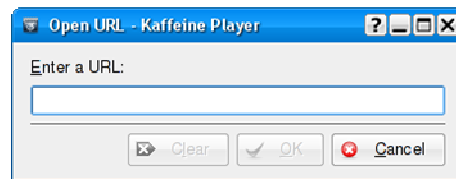


Figure 16 Open URL dialog

### 3.33 Inconsistent placement of Clear button in Open URL dialog (medium)

There is Clear button placed in the Open URL dialog (figure 16), which allows the user to clear the input of the URL text field. However, instead of placing it left of the field consistent with other KDE applications, it is placed below it.

Either remove the Clear button, or place it left of the text field. The necessity of the Clear button has been discussed elsewhere<sup>11</sup>.

Keep the user interface as clean and minimal as possible.

### 3.34 Unnecessary explicit notion of Xine engine (high)

Kaffeine uses the Xine multimedia playback engine to play multimedia files. The explicit notion of this is highly questionable, since it requires an extensive amount of technical knowledge. Moreover, it causes the configuration dialogs to be separated (figure 17), instead of having settings centrally organized in the Configure Kaffeine dialog. This separate Xine Settings dialog is not very intuitive, since it also requires experience with configuring Xine-ui, the standard front-end for Xine-lib.

Integrate the Xine engine into the user interface. This is for the most part already done, as you can see with the controls, but not with settings.

Speak the user's language. Only programmers are aware of the distinction of the back-end and the front-end of an application.

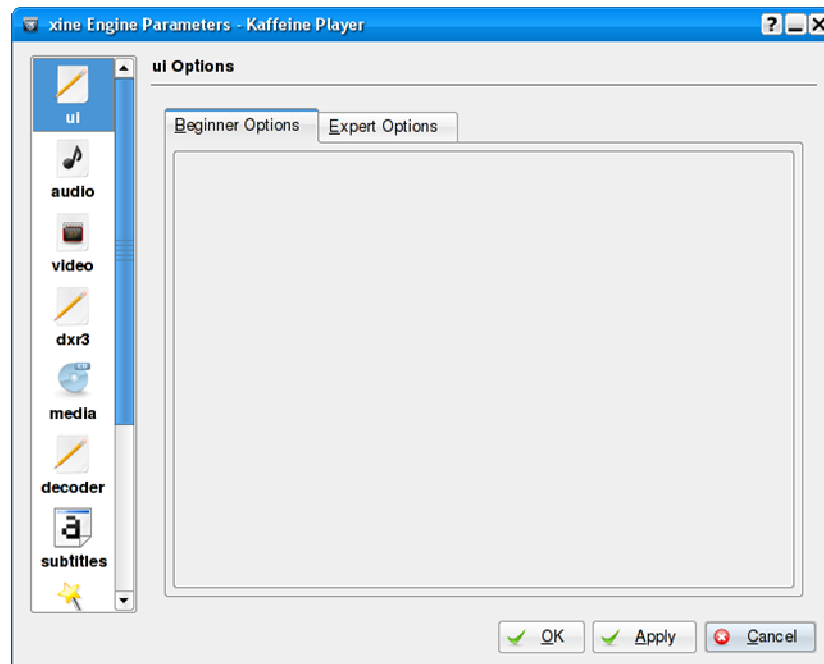


Figure 17 Xine Settings dialog

### 3.35 Incorrect label in View menu (low)

In the view menu, there is a command to preserve the video aspect ratio (figure 18). Instead of 'Keep Aspect Ratio', it is called 'Preserve Original Aspect'.

Change the label to 'Keep Aspect Ratio'.

Communicate clearly to the user.

<sup>11</sup> <http://lists.kde.org/?l=kde-usability&m=114532406509037&w=2>

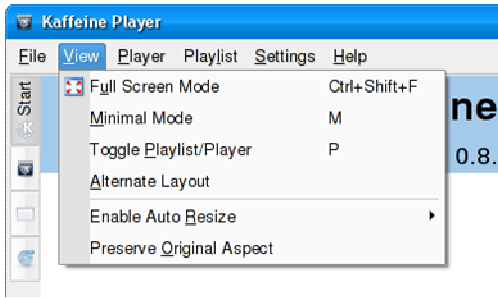


Figure 18 View menu

### 3.36 Redundant information in Audio CD tab (low)

There is a text label placed in the top of the Audio CD tab. This text label tells the user in which tab he currently is looking at. This is not very useful, since the information is already provided in the text label of the tab itself. It is also inconsistent, since not every tab has a text label in the top.

Remove this text label.

Keep the user interface as clean and minimal as possible.

### 3.37 Awkward No Album Art available icon (medium)

In the Playlist tab, there is Album Art displayed where available (figure 19). At least, the icon shows a CD jewelcase with two persons on the cover and an 'X' sign. The icon is painted with shades of grey. This is not necessarily a bad thing, but the contrast is also low. This makes it more difficult for the human eye to visually distinguish the shapes.

Either add more color to increase presented information, or use text to display it.

Communicate clearly to the user.



Figure 19 No album art available?

### 3.38 Tiny browser buttons in Playlist tab (medium)

There is a browser pane placed in the Playlist tab, where users can drag items from the browser to the playlist. The buttons that are accompanied with the browser are very small and with a low-contrast icon theme, shapes are hardly distinguishable.

Make the icons look bigger, or remove the browser pane completely (see 3.22).

Small icons are visually harder to distinguish than their larger counterparts.

### 3.39 Low affordance of Alternate Layout command in View menu (low)

Aside from the default window layout, there is an alternate layout available. This layout only applies to the Playlist and Audio CD tabs. When the user clicks the 'Alternate Layout' in the Player tab, nothing happens.

Place this option in the Configure Kaffeine dialog, since this option will probably be not frequently used. Aside from this, this will also reduce the amount of menu items, which will benefit the minimal amount of interface elements.

Communicate clearly to the user.

### 3.40 Low leverage of Play button (low)

The Play button starts or resumes playback. This button also functions as a pause button in several other multimedia players, such as Windows Media Player<sup>12</sup> and iTunes<sup>13</sup>. This solution results in a cleaner interface.

Use a Play/Pause button instead of two separate buttons.

User interfaces have to deal with a limited screen size. Use the space as effectively as possible (Mullet and Sano, 1994).

## 4. Discussion

After a short evaluation, a total of 40 usability issues were found. For each issue that has been found, its severity has been graded. There were 15 issues discovered of low severity, the majority of 20 issues were marked medium, and the remaining 5 issues are considered highly severe.

The heuristic evaluation consisted mostly of creating screenshots of every part of the interface. Afterwards, every screenshot was examined and every issue found was noted. The behavior and usage of the application was minimally tested. User testing is needed to get accurate and reliable results about task-oriented usability issues. This concerns issues: [3.8](#), [3.20](#), [3.21](#), [3.28](#), [3.29](#), [3.30](#) and [3.34](#).

None of the issues that I have found are related to software bugs, i.e. they were design related. The application never crashed during testing. I did have the necessary multimedia codecs installed, so issues regarding meaningful feedback remain unknown. This is however important to know, since most desktop Linux distributions do not ship multimedia codecs.

## 5. Conclusion

Based on the results of the heuristic evaluation, more attention should be given to the user interface of Kaffeine. The main problem is the placement of certain user interface elements. Some offer redundant information or interaction methods, while others are placed in screens or dialog where they have no meaning.

Kaffeine technically does a great job and it is very capable of what the developers claim it can do. However, its interface reflects complexity by showing both extraneous and noncontextual information. This may confuse novice and beginner users, thus slowing down the learning process.

Summarizing: the three main points for improving the usability of Kaffeine are: (1) get rid of all the unnecessary information, (2) integration of Xine settings interface with Kaffeine and (3) place information where it belongs.

---

<sup>12</sup> <http://elecboy.com/photos/wmp11.jpg>

<sup>13</sup> [http://www.sg.hu/kep/2003\\_10/1020itunes1.jpg](http://www.sg.hu/kep/2003_10/1020itunes1.jpg)

## References

- Benson, C. Elman, A., Nickell, S., Robertson, C.Z. (2004). *GNOME Human Interface Guidelines 2.0*. Retrieved June 22, 2006, from: <http://developer.gnome.org/projects/gup/hig/>
- Ellsworth, E. (2003). *A brief introduction to writing a usability report*. Retrieved April 27, 2006, from: <http://usability.kde.org/activity/usabilityreports/howto/>
- Heuristics. (2006). Retrieved June 22, 2006, from: <http://www.betterdesktop.org/wiki/index.php?title=Heuristics>
- Mayhew, D.J. (1992). *Principles and Guidelines in Software User Interface Design*. Prentice Hall.
- McCracken, D.D., Wolfe, R.J. (2004). *User-centered website development*. New Jersey: Pearson Prentice Hall.
- Mullet, K., Sano, D. (1994). *Designing Visual Interfaces: Communication Oriented Techniques*. Prentice Hall.
- Nielsen, J. (1990). *Heuristic evaluation of user interfaces*. Proc. ACM CHI'90 Conf.
- Nielsen, J. (1993). *Usability Engineering*. London: Academic Press.
- Nielsen, J. (1999). *When bad design elements become standard*. Retrieved April 27, 2006, from: <http://www.useit.com/alertbox/991114.html>
- Nielsen, J., Mack, R.L. (1994). *Usability inspection methods*. New York: John Wiley & Sons.