

CS190 Project Specification - LinuxDJ

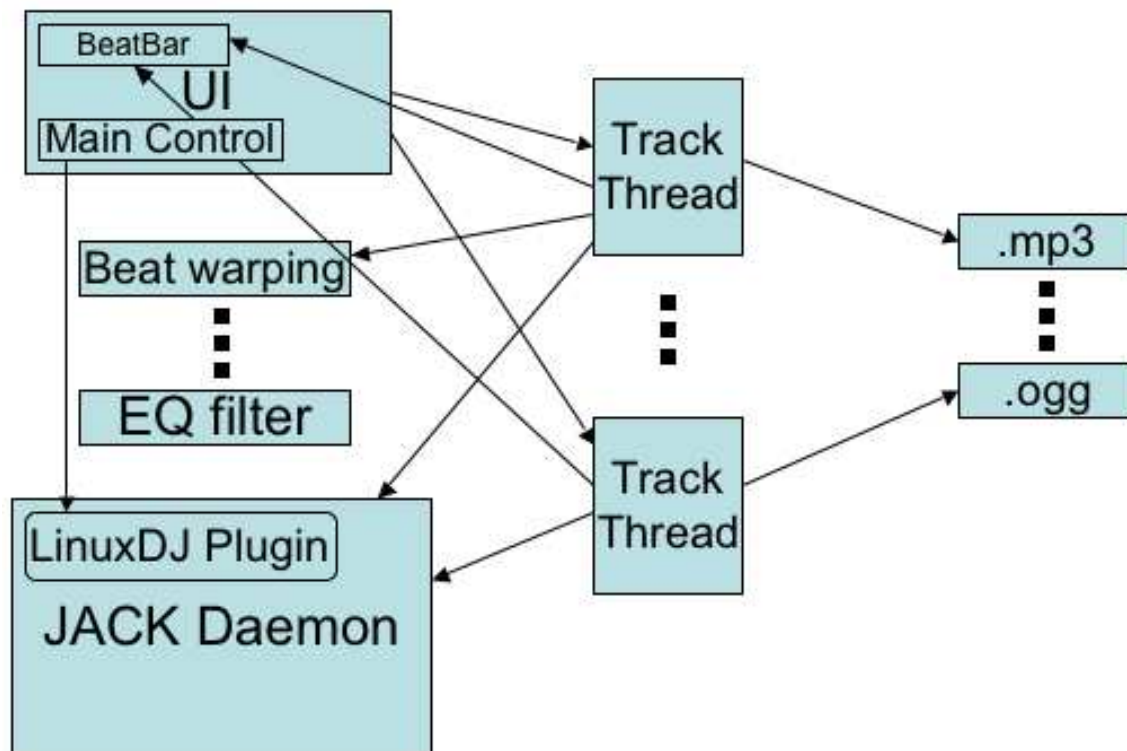
1 Title

LinuxDJ

2 Description

A DJ program for Linux. Currently no decent ones exist. This would be loosely modelled on VirtualDJ, which is a PC product with a Mac beta. It would support audio filters, mixing/blending of audio tracks, stretching or compressing songs to match beats (automated), looping based on number of beats, and possibly more, all real-time. Mixing and filtering of audio tracks can be done on individual, or group basis. The UI will also be an improvement over VirtualDJ, removing or hiding rarely used parts of the interface, and extending the system for managing more than two tracks, in arbitrary dynamic groupings.

3 System Model Diagram



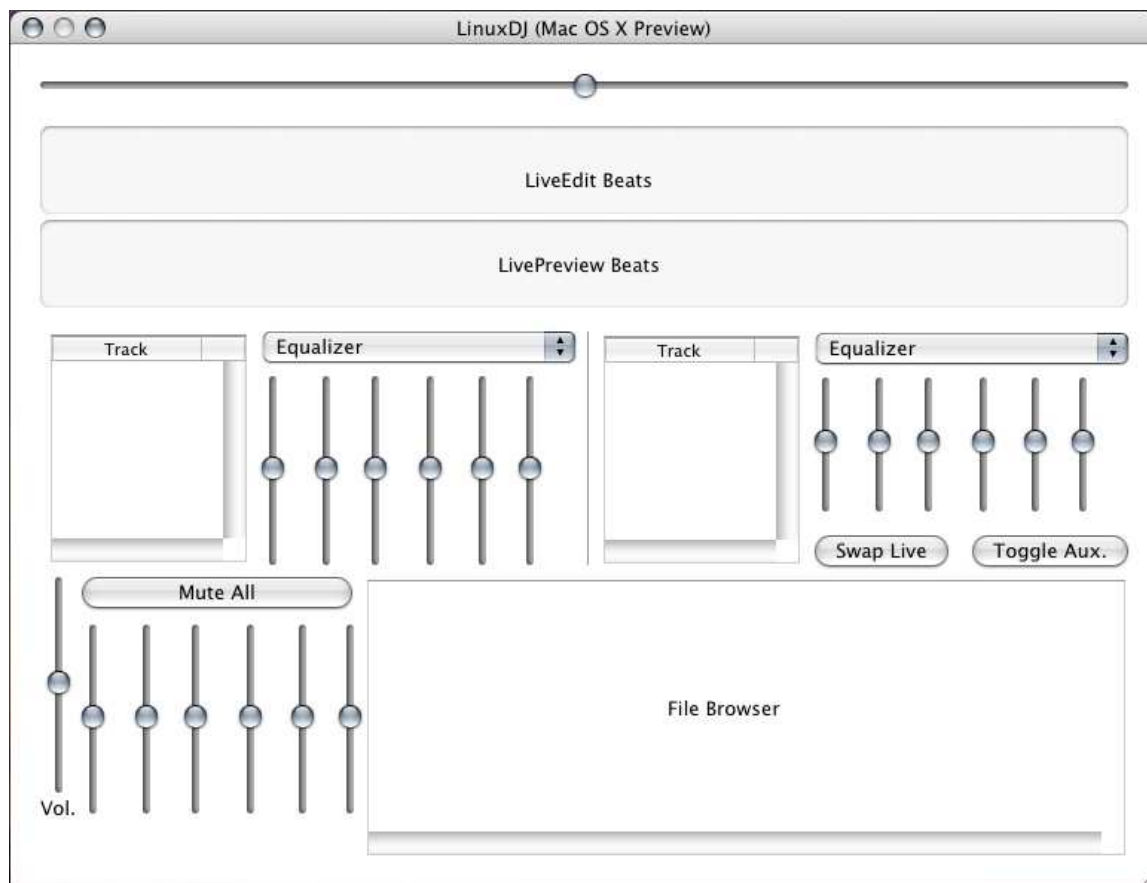
4 System Model Description

The UI is the main interaction source. Through it, settings are determined, which are propagated to the individual threads which manage each sound track. Each thread manages *all* effects associated with *one* track. Each track then sends its data to the appropriate JACK server as it processes. Modifications to the global control panel (e.g. global volume) are processed by sending messages to the LinuxDJ JACK plugin, which modifies the joined output of all the processing threads for the main output.

Each thread makes calls to one or more filter classes during processing. When first opening, the thread makes calls to the appropriate file format handler, and continues to read through that.

Each thread also sends data about that track's beat to the GUI, to be used in the Beat Bar display.

5 User Interface Diagram



6 User Interface Description

6.1 Layout

The topmost object, spanning the width of the application, is the Fade Bar. Below that, also full-width, is the pair of Beat Bars.

Below those, there is a pair of similar panels, the Live-Edit on the left, and Live-Preview on the right.

Below that, there is another pair of panels, different from each other, and shorter than the previous pair. On the left is a relatively narrow Master Control Panel, and the remainder of the width is taken up by the File Selection pane.

6.2 Set Fade Bar

A slider, intended to be slid from one end to the other. Each end represents one of the sets being the sole source of external audio, with the middle being a blend of the two. This will usually be used for swapping which audio is active for the listening audience. If the sets are swapped (see Live-Preview Panel), this bar is adjusted automatically to match.

6.3 Set Beat Bars

A pair of time-scrolling bars, each of which displays vertically indications of beats in the music. The top bar corresponds to the external music feed, and the bottom corresponds to the DJ preview. The two are stacked vertically so that beats in the preview set can be adjusted to match (or correspond in some way) to the beats in the currently playing set. Within each bar, each track of the corresponding set appears as a different-colored overlay, so each track can be adjusted to match beats in its own set. When only a single track is active for editing in a given set (see the Live-Edit and Live-Preview Panel descriptions), dragging horizontally within a set slides the layer for that track, shifting its phase to sync up one track with an arbitrary part of another.

6.4 Master Control Panel

A panel including master volume control, a mute button, sliders for balance, etc., all of which affect the external music feed.

6.5 Live-Edit Panel

This panel is split into two main parts. The left side is a list of the tracks for the externally playing set. Next to each item is a check box. Selecting the checkbox adds that track to the group of tracks which are modified by the controls on the right side.

The right side has a drop-down box at the top, which allows the user to select between the effects list, looping controls, and an equalizer, which are applied simultaneously to all tracks in that group.

Unchecking a track removes it from that group, but changes made to that track's play properties while it was in the editing group still apply to that song.

When multiple tracks are selected which have pre-existing customizations which differ, changes are applied still to each track, relative to its individual customizations. Editing one of those tracks alone later will merge all applied effects in the displayed settings.

6.6 Live-Preview Panel

This panel is identical to the Live-Edit Panel, with a couple exceptions. First, this panel has two extra buttons: "Swap Live" and "Toggle Aux Out." The first of these buttons swaps this panel (visually) with the Live-Edit Panel. Clicking it should have the two panels visually switch sides, remove the two extra buttons from what was originally Live-Preview, and add them to the panel which has become Live-Preview. The second button toggles the auxiliary output (headphones) between listening to the Live-Preview panel, and listening to the currently live music.

6.7 File Selection

The lowermost pane of the display is a file browser of typical form, with a tree on the left, and a current-directory-contents display on the right. Files from the right pane can be dragged and dropped into the track lists on either of the Edit or Preview panels' track lists.

7 Nonfunctional Requirements

All effects and changes to the audio tracks which are currently playing must manifest in real time up to a reasonable number of tracks and effects. This cannot be stressed enough. The software must be stable, even if a user applies enough transformations to slow the machine. It should be intuitive and easy to use for people familiar with the idea of mixing tracks. Should run on any Linux system with the correct audio support systems and GUI toolkits. Documentation should be present, but most users should not need to consult the documentation frequently - the UI should be "discoverable." Should not depend directly on more than one audio support system.

8 Functional Requirements

By priority (in decreasing order):

1. Real-time audio processing - if effects cannot be applied in real-time, the software becomes audio editing software, not live DJ software, and loses significant usefulness.
2. Support for fading between groups of tracks - this is the most commonly used feature of DJ software, and in some cases is the bare minimum for calling software a piece of DJ software.
3. Support two simultaneous, distinct outputs. One is for live play, the other can be used by the DJ to preview an assembly of tracks to play after the current set finishes.

4. Looping support, including looping a particular subsection of a track - for live remixing, this is an absolute must.
5. Independent effects controls for each song being mixed - a must.
6. Support for altering track speeds and matching beats automatically - also listed as a must for smooth transitions and interesting remixes.
7. Support for more than 2 songs - lower priority, because it's nice, but the final upper limit depends on the software's efficiency, and the hardware used.
8. Better UI than existing software - most amateur DJ software on all platforms has a terrible interface. VirtualDJ on Windows is probably the best out there, but is still cluttered with largely-unused features. There should be auto-hiding of unused UI elements. Adaptations must also be made to support the arbitrary grouping of songs, and layering effects on these groups.
9. Supporting many file formats for input.
10. Audio effects filters - a nice touch, though not strictly necessary.

9 Risks

Anything real-time is inherently hard, because it imposes very strict response time requirements. In this case, an external program, such as the JACK daemon for audio mixing may be used to facilitate this, and any external dependencies are also risky, as those can be updated independently of this software. Audio processing also would require some research, and may be difficult to implement. Basically all of the requirements, including the ambitious UI are of significant difficulty, and doing any of them wrong makes the software unuseable. Constructing the UI in a counterintuitive way could also make the functional improvements of this software over prior works less likely to be used, reducing the value of LinuxDJ. It also requires two distinct, separable audio outputs.