

Audacious - Bug #1073

The current file playback ends <Buffer size option value> ms earlier

February 22, 2021 09:59 - Igor Kushnir

Status:	Rejected	Start date:	February 22, 2021
Priority:	Minor	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Affects version:	4.1		
Description			
Steps to reproduce			
<ol style="list-style-type: none">1. Set Audio=><i>Buffer size</i> option value to 10'000 ms in Audacious Settings.2. Restart Audacious to make sure that the configuration takes effect.3. Play a short file (longer than 10 seconds). Alternatively seek close to the end of a longer file (but not as close as 10 seconds).4. Look at the audio progress bar until the playback ends.			
Actual result			
<p>The UI reports that the current track playback ends 10 seconds before the progress bar reaches the end. The next track appears to start 10 seconds earlier. The next-track desktop notification appears 10 seconds earlier. A configured <i>Song Change</i> command runs 10 seconds earlier. And so on. The exception is the actual audio playback itself: it ends in time - 10 seconds later than the UI indicates.</p>			
Expected result			
<p>The <i>Buffer size</i> option value does not affect the UI and General plugins.</p>			
Additional information			
<p>I have recently increased my <i>Buffer size</i> to 2500 ms because of playback stuttering during heavy filesystem use by other programs. I have noticed this issue because I had configured starting playback of a next file as a custom <i>Command to run at the end of the playlist</i> in <i>Song Change</i> plugin. Therefore the last two seconds of audio of the last playlist item are cancelled by the next scheduled file playback. A workaround is to delay running the custom command by <i>Buffer size</i> ms:</p>			
<pre>bash -c "sleep 2.5 && <custom command>"</pre>			
<p>But the next-track desktop notification still appears too early.</p>			

History

#1 - February 24, 2021 15:50 - John Lindgren

This is the way it's designed to work. The playlist (and therefore most of the UI) follows the track currently being decoded. The audio stream will lag behind more or less depending on buffering.

#2 - February 24, 2021 16:00 - Igor Kushnir

John Lindgren wrote:

This is the way it's designed to work. The playlist (and therefore most of the UI) follows the track currently being decoded. The audio stream will lag behind more or less depending on buffering.

This looks like a design mistake then, because users usually care which track is currently playing, not which one is being decoded. I think the bug should remain open - perhaps it will be addressed during a future major overhaul of the relevant code.

#3 - February 25, 2021 16:04 - John Lindgren

I personally hope there aren't any more major overhauls of the relevant code.

#4 - December 18, 2021 17:07 - John Lindgren

- *Status changed from New to Rejected*

Closing this as "won't fix"; the effort required is too high for a small benefit.