Audacious - OLD, PLEASE USE GITHUB DISCUSSIONS/ISSUES - Feature #355

Shuffle is not random

October 03, 2013 14:30 - Виктор Пономарёв

Status:	Rejected	**Start date:**	October 03, 2013
Priority:	Minor	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Affects version:	3.4.1		

Description

In shuffle mode the player is playing the same songs. Can do something to shuffle be random, and not be "shuffle" with same intervals?

History

#1 - October 03, 2013 17:55 - John Lindgren

I'm not sure what you mean. Of course it will play the same songs, but in a random order. Can you please explain better?

#2 - October 03, 2013 18:11 - Виктор Пономарёв

My English is bad. I'll try to explain better.

There is a list of 300 songs. If I turn to play the songs in shuffle mode, play the same songs, but do not play other songs from this list. As if the player chose a certain interval, and includes songs in sequence. But with 300 songs played only a part of them, the other part is not played at all. I watch it about 1 year. I'm tired of hearing the same thing. I want a real change.

#3 - October 05, 2013 01:05 - John Lindgren

So, what you're saying is that over a year of observation, you've seen certain songs played repeatedly and others never played. Do I understand you correctly?

#4 - October 05, 2013 07:56 - Виктор Пономарёв

Yes, John. It looks like this:

1...50...100...150...200...250...300 – is number of songs (N=300).

If play with shuffle mode, songs are played in sequence:
1...50...100...150...200...250...300, or
1...52...99...151...203...248...299, or
2...49...101...148...199...251...298.
It is an intensified model of play. I want to show that, for example, 5 – 45 songs never played.

The situation changes if the playlist is constantly replenished. If the compositions were 300, now is, for example, 400 (N=400):
1...60...120...180...240...300...360...400 – near these numbers player will choose songs for play.
As can be seen from the description, the player chose exemplary interval N/7. Again, my description is only a model for a visual explanation.

I propose to change the function of mixing, adding another a random variable:
S – function of shuffle mode,
N – the number of songs in a playlist,
A – random of N now,
B – additional random variable (from 1 to, for example, 50).
S(n)=A+B
In such a case can be played songs other than 1 and 50, and near these numbers.
It is my suggestion. Perhaps you can find a better solution. I'm just giving an idea.

#5 - October 06, 2013 00:36 - John Lindgren

I've never seen the behavior you describe. Audacious gets the playback order from the rand() function in the C run-time library (seeded with the current time(), as is common practice). It sounds like your C library has a broken random number generator.

#6 - October 06, 2013 01:32 - John Lindgren

- Category deleted (audtool)

#7 - October 06, 2013 06:54 - Виктор Пономарёв

May be. I conclude on the basis of the 1 year observations. I listen my playlist constantly and watch that some songs never played. I watched it and when listening playlist on the other computer.

#8 - October 09, 2013 14:56 - John Lindgren

- Status changed from New to Rejected

Resolution: unreproducible.

#9 - June 14, 2014 08:22 - michael b

John.
I too am having the same issue with the shuffle play. Whatever the method is it is not random.
I have 342 songs and playing and in 8 hours I can hear some songs upto 4 times EVERY day. And other songs are never played.
But if I manually forward to the next song then its does seem to be random and I'm unable to see a repeat easily.

Honestly, you have an issue. I'm now tired of the repeated some and missing on others so bye bye Audacious! sorry

Michael

#10 - October 03, 2016 12:22 - Thorsten Lisker

While this is an old thread, I am definitely observing the same issue: the list of actually played songs appears not truly random over the course of ~8 weeks, but is very much like Виктор Пономарёв described (that's what brought me here). If it takes the rand() function, fine - but is this somehow re-using the same seed or re-using a previously generated list!? Just wondering whether more people are having this issue. Thanks!

I am indeed getting closer to what Michael said, "I'm now tired of the repeated some and missing on others..."

This is Audacious 3.7.2 on kubuntu 16.04

Thorsten

#11 - February 27, 2017 11:27 - xyz dragon

- File rand-test.cpp added

I didn't look at the code, but I think one problem could be that audacious does something like this:

```
songToPlay = rand() % nSongs
```

But this means, that not only the order, but also the amount each song is played gets randomized. This can be tested easily with the attached file. E.g. For 10 songs to play tack 4 gets played the first time after 44 tracks being played! In the meantime track 2 gets played 8 times. At average each song should have been played 4.4 times already.

Or for the example used above, i.e. 300 songs, the attached program returns:

```
All songs were played at least one time after 1665 tracks being played.
These are the top ten most often played tracks out of 300 total tracks to play after 1665 tracks being played, i.e. at average each track should have been played 5.55 times!
Track  228 played   13 times
Track  284 played   13 times
Track  290 played   12 times
Track  129 played   12 times
Track   40 played   11 times
Track  297 played   11 times
Track    5 played   11 times
Track    4 played   10 times
Track  225 played   10 times
Track  224 played   10 times
These are the 10 least often played songs:
Track  271 played    2 times
Track  207 played    1 times
Track   48 played    1 times
Track   44 played    1 times
Track  191 played    1 times
Track  175 played    1 times
Track  156 played    1 times
Track  255 played    1 times
Track  266 played    1 times
Track  116 played    1 times
Track  277 played    1 times
```

If this is really how audacious shuffles the track list, then instead it should preshuffle the tracklist internally using randomized swaps and then play that list. This guarantees that after 300 out of 300 tracks being played each track will be played exactly one time, but with a random order. And then after 300 tracks being played the internal tracklist needs to be shuffled again. This of course means additional work for audacious because it needs to manage the internal shuffled state of the tracklist e.g. in an internal variable or in /tmp/.shuffled-13315.m3u.

#12 - February 27, 2017 14:52 - John Lindgren

xyz dragon, you're the first person in this thread to make a real attempt to understand how the shuffle algorithm works. Thank you for that.

Really though you should have looked at the code; it's not quite that simple. Audacious does keep track of an internally shuffled list so that each song is played once before repeating, but that list is currently not saved when you close the program. So if you have a short playlist and play it to completion, you'll hear each song once. If you play a whole collection of 1000+ songs on shuffle, restarting Audacious every day, you'll hear some songs more often than others, as you observed in your test program.

#13 - March 18, 2017 16:32 - xyz dragon

@John Lindgren Ok, I see your point.

For users running into this problem. Just use "Playlist -> Sort -> Random Order" and then play that shuffled playlist sequentially.

Files

rand-test.cpp	2.82 KB	February 27, 2017	xyz dragon