

# Sonic Visualiser - Bug #1991

## Layer export tests have started failing

2020-06-12 11:47 AM - Chris Cannam

<b>Status:</b>	Closed	<b>Start date:</b>	2020-06-12
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			

### Description

The layer export test script is reporting substantial differences in the spectrogram and peak-frequency spectrogram outputs.

Confusing the matter, on the 9th of April I committed some changes to the expected output for these (commit:048d9eed0a1f) and I am no longer clear on why. Neither the previous expected output nor the current expected output is matching the current actual output.

Attached are three versions of the selected-spectrogram output:

- [Actual output from current SV revision](#) (commit:5a397acc59 on Linux)
- [Expected output according to current repo](#)
- [Expected output according to repo prior to the 9th of April](#)

### History

#### #1 - 2020-06-12 11:50 AM - Chris Cannam

- Description updated

#### #2 - 2020-06-12 12:00 PM - Chris Cannam

The last 10 values from the first row of selected-spectrogram for each of the three cases cited above (in the same order):

```
0.000615055,0.000994842,0.000629527,8.51144e-05,0.000180405,0.000461554,0.000846313,0.000258203,0.000554893,0.00086084
0.000613816,0.000995983,0.000628786,8.45989e-05,0.000180812,0.000461687,0.000847238,0.00025846,0.000555437,0.000860612
0.000613816,0.000995983,0.000628786,8.45989e-05,0.000180812,0.000461687,0.000847238,0.00025846,0.000555437,0.000860612
```

So the differences between the actual output now and the two expected output files are far more substantial than the difference between the two expected output files (which are identical in this line - the first difference is in line 3, where we have the odd difference in the sixth dp).

#### #3 - 2020-06-12 12:11 PM - Chris Cannam

Same output, but with fourth and fifth lines added:

- fourth line is the 4.1-pre1 Applmage build
- fifth line is the 4.1-pre1 code built locally just now

I am **almost** certain that these tests passed locally at the point when the 4.1-pre1 release was actually made - if so, then it's something environmental going on. But only almost certain.

(The change from e-05 to e-5 in line 4 will be a Qt QString change)

```
0.000615055,0.000994842,0.000629527,8.51144e-05,0.000180405,0.000461554,0.000846313,0.000258203,0.000554893,0.00086084
```

0.000613816,0.000995983,0.000628786,8.45989e-05,0.000180812,0.000461687,0.000847238,0.00025846,0.000555437,0.000860612  
0.000613816,0.000995983,0.000628786,8.45989e-05,0.000180812,0.000461687,0.000847238,0.00025846,0.000555437,0.000860612  
0.000613509,0.000996341,0.000628716,8.47737e-5,0.00018063,0.000461737,0.000847137,0.000258392,0.000555429,0.000860612  
0.000615055,0.000994842,0.000629527,8.51144e-05,0.000180405,0.000461554,0.000846313,0.000258203,0.000554893,0.00086084

#### #4 - 2020-06-12 03:57 PM - Chris Cannam

(See also #1968 Layer-exports test is fragile. But note that a current debug build produces the same results as the release build in this respect)

#### #5 - 2020-06-12 05:48 PM - Chris Cannam

Equivalent output, but from the 4.1-pre1 Ubuntu .deb running under Ubuntu 16.04:

0.000613509,0.000996341,0.000628716,8.47737e-05,0.00018063,0.000461737,0.000847137,0.000258392,0.000555429,0.000860612

So that's much the same as the 4.1-pre ApplImage running on a newer system.

What's most difficult is being unable to find any code or environment that produces either of the old expected outputs! Without this it's impossible to discover what might have changed.

#### #6 - 2020-06-15 01:43 PM - Chris Cannam

OK, the culprit is **libvorbis**.

The current upstream version 1.3.6 (from 2018) was rebuilt for Arch Linux package 1.3.6-2 ([reproducibility rebuild](#)) and the rebuilt package produces different output. Not wildly different, but definitely different. Rolling back this package restores essentially the original results, differing in at most the occasional least-significant digit (though still confounded by the fact that I changed the expected output file at some point).

Nothing has changed in either the package build directions or the upstream code between these two releases. I wondered if the compiler has started doing some more aggressive `-ffast-math` optimisations - the build does appear to use `-ffast-math` - so I downloaded the source and built it both with and without that flag. It made no difference: both versions exactly matched the output from the current 1.3.6-2 package.

So the cause is a mystery, but it seems clear that we can't expect to use Ogg/Vorbis files if our intention is to provide useful regression tests for anything else in the system.

#### #7 - 2020-06-29 02:33 PM - Chris Cannam

- Status changed from New to Closed

#### Downloads

selected-spectrogram-actual-now.csv	476 KB	2020-06-12	Chris Cannam
selected-spectrogram-expected-now.csv	476 KB	2020-06-12	Chris Cannam
selected-spectrogram-expected-previously.csv	476 KB	2020-06-12	Chris Cannam