Tutorial 7:
TASCARpro on stage: Music and light processing

Summary

Tips and tricks for processing of music. You will create a virtual live mix of a jazz band, and can compare it to a conventional stereo mixing approach. Add motion to your virtual stage, add light to your show. Try ambisonic dance-floor mixing.

What will I learn?

- Mixing, direct-to-reverb ratio, source width control
- Parametric motion control
- All the other things TASCARpro can do

What can I use it for?

- Creative / art / music projects
- Parties in the NeSSy foyer
- Experiments in the Gesture lab

Part 1: Mixing music

We prepared an Ardour (http://ardour.org) session with a recording of a jazz ensemble (http://lalists.stanford.edu/lau/2011/05/0534.html). First try to create a stereo mix (two loudspeakers are already connected). Use gain and panning. If you add audio plugins, make sure to add them pre-fader (you will see later why). Save your session before adding plugins! Evil plugins (there are some out in the open source world) may crash your audio workstation.

Are you happy with the result? Now leave the session Ardour open, but press the mute button on the master fader. Open the "tutorial7.tsc" TASCAR session file. This will create a TASCAR session with the instruments, connected to the Ardour tracks. How does it work? In each track you may find a "send". Each send creates a jack output port. Each Ardour "send" audio port is connected to the corresponding sound vertex in TASCAR.

Now place the band on the stage. The drum set, guitar and bass have more than one sound vertex – try to distribute them in space. If you configure the vertical position of the sound vertices, all objects can remain on the floor (or stage) level.

```
10    <src_object name="Drumset" color="#edb120">
11      <sound maxdist="200" name="snare" connect="ardour:send 2/audio_send 1"
      x="0.3" y="0" z="0.65"/>
```

Now all instruments are still in the origin. You can edit the instrument positions in the TASCAR file by adding a position to each object, as it was already done for the receiver:

```
30    <receiver name="out" type="hoa2d" layout="glpo.spk" caliblevel="130"
      filtershape="triald">
31      <position>0 -4 0 1.6</position>
32    </receiver>
```

Press the "reload" button after each change. Tired of all the manual changes? Use our GNU Octave positioning tool `jazzpos.m`. This will send positions via OSC. But note: These are not saved in the session file; you have to manually edit the positions once you are happy.

Source width: An instrument is usually not a point source. Increase the size of an instrument, e.g., by using the system command `send_osc`:

```
send_osc 9877 /scene/Keyboards/*/size 1
```

You can isolate ("solo") sounds by pressing the "S" button for the desired sounds and receiver pair.

Need some room? Add a room acoustic model of Wilhelm13 (http://www.wilhelm13.de) – un-comment the `facegroup` elements:

```
35    <facegroup importraw="w13.raw" r="1" d="0.4">
36    <position>0 -4 0 -0.4</position>
37    </facegroup>
```

This will add early reflections (second order geometric image source model). To add also diffuse reverberation, you may use an impulse response recorded in the foyer of the house of hearing:

```
40    <receiver name="room" type="omni" image="false" size="20 20 20"/>
```

```
44        <scene name="reverb">
45        <diffuse name="reverb" size="20 20 20"/>
46        <receiver name="out" type="hoa2d" layout="glpo.spk" caliblevel="130">
47        <position>0 -4 0 1.6</position>
48        </receiver>
49        </scene>
50        <module name="system" command="jconvolver hdh.conf"/>
51        <connect src="render.scene:room" dest="jconvolver:in"/>
52        <connect src="jconvolver:out.0w" dest="render.reverb:reverb.0w"/>
53        <connect src="jconvolver:out.1x" dest="render.reverb:reverb.1x"/>
54        <connect src="jconvolver:out.1y" dest="render.reverb:reverb.1y"/>
55        <connect src="jconvolver:out.1z" dest="render.reverb:reverb.1z"/>
```

Do you like your TASCAR mix? Compare the results to your stereo mix! (mute receivers in TASCAR, un-mute master fader in Ardour).

Part 2: Light control with TASCAR

TASCAR can control light. We use the DMX512 standard for controlling lights (https://en.wikipedia.org/wiki/DMX512). TASCAR provides drivers for the network-based Artnet protocol (used in the Gesture lab) and for the openDMX USB device by Entec (used in this tutorial). The tools are documented in the TASCAR manual, section 6.16.

In TASCAR, a light scene can be used to create static or moving light settings. To track objects, first create a audio scene (see `tutorial7light.tsc`):

```
4     <src_object name="src1">
5       <position>0 -5 -5 0
6       2 -5 5 0
7       4 5 5 0
8       6 5 -5 0
9       8 -5 -5 0</position>
10      <sound name="0">
11  <!-- <sndfile name="..." level="70"/> -->
12      </sound>
13    </src_object>
14    <receiver name="out" type="hoa2d" layout="glpo.spk" caliblevel="130"
      filtershape="triald">
15      <position>0 -4 0 1.6</position>
16    </receiver>
```

Now you can setup the lightscene to track objects in the audio scene:

```
19    <lightscene name="ceiling" objects="/scene/src1" parent="/scene/out"
      channels="4" method="nearest" objval="255 0 130 0">
```

Try to setup a second moving source, and track a second light object with a different color (hint: use `objects="/*/src*"`).

Now try to control the lights via OSC. For this, check the list of OSC variables, then open a terminal, and use the command `send_osc` to send the colors.