

**Working with Open Source applications for electronic compositions:
audacity, pure data, csound**

Second Day: Going deeper into Csound

1. Transporting our FM example from pd to Csound

- using the **foscili** opcode
- creating a sinewave function table with **ftgen** and **GEN01**
- building a GUI (graphical user interface) with the Widgets window and **invalue / outvalue** (use right-click to create a widget in the Widgets window)
- including MIDI with **ctrl17**

2. Working with Samples and discussing a problem

- playing a sound file with **diskin**
- learning about the parameters *kpitch* (= speed) and *iskip*
- using random choices via **random** and **randomi** (and **seed**)
- making decisions with **if ... then ... else**
- using **event_i** to call another instrument
- performing loops with the **timeout / reinit** construction
- Discussion::

"Why?" or:

we need an *idea* to know where to go in programming ... and then work a lot

3. Granular Synthesis and building your own opcodes

- the **partikkel** opcode
- understanding the main parameters
- building your own opcode with the **opcode** facility
- another example from my theatre music

4. FFT Analysis and Resynthesis

- what the Fast Fourier Transform (FFT) is
- how we can perform an FFT analysis using the Csound Utility **pvanal**
- reading the .pvx file with **pvsfread** and resynthesizing with **pvsynth**
- making a slow motion or a freeze effect with the *ktimpnt* parameter

5. Convolution

- what it is
- very simple to do with the **pconvolve** opcode in Csound
- download some impuls responses from noisevault.com or try it with own sounds