

Sysbas1B

Welkom!

SuperCollider les 3

SuperCollider - basis

- ✓ sneltoetsen
- ✓ variabelen
- ✓ functies
- ✓ objecten (uGens)
- ✓ classes (SynthDef)
 - communicatie

SuperCollider - basis

- communicatie
 - Midi
 - polyfonie
 - OSC

Midi-in

Configuratie

Makkelijkste manier; verbindt alle apparaten:

```
MIDIIn.connectAll;
```

Als je specifiek één apparaat wil verbinden:

```
MIDIClient.init;
```

```
MIDIClient.sources
```

```
MIDIIn.connect(0,MIDIClient.sources.at(0));
```

Midi-in

MIDIFunc

MIDIFunc voor specifieke midi-functie

```
MIDIFunc.noteOn
```

```
MIDIFunc.noteOff
```

```
MIDIFunc.cc
```

Midi-in

MIDIFunc gebruiken

Maak een MIDIFunc om note-on-berichten te ontvangen.
Binnenkomende data wordt gepost.

```
~midinoot = MIDIFunc.noteOn({|...args|  
  args.postln;  
});
```

Resultaat: [100, 21, 0, -336859671]
[vel, note, chan, source-id]

Midi-in

MIDIFunc gebruiken

Belangrijk! Verwijder een MIDI-func weer:

```
~midinoot.free;
```

```
~midinoot = MIDIFunc.noteOn({|...args|
```

```
  args.postln;
```

```
});
```

Anders blijft de oude bestaan en worden er meerdere dingen tegelijk uitgevoerd.

Midi-in

MIDIFunc gebruiken

Maak een MIDIFunc om note-on-berichten te ontvangen.
Binnenkomende data wordt gebruikt om Synth te starten.

```
~midinoot.free;  
~midinoot = MIDIFunc.noteOn({|vel,note|  
  ~sint = Synth(\synthnaam,{\freq, note.midicps,  
                                \amp, vel/127,  
                                \gate, 1}));  
});
```

Midi-in

MIDIFunc gebruiken

Maak een MIDIFunc om note-off-berichten te ontvangen.
Binnenkomende data wordt gebruikt om Synth te stoppen.

```
~midinoot.free;
```

```
~midinoot = MIDIFunc.noteOff({ |vel,note|
```

```
  ~sint.set(\gate,0);
```

```
});
```

Midi-in

MIDIFunc gebruiken

Probleem: elke nieuwe noot die je speelt, overschrijft de vorige noot.

```
~midinoot.free;  
~midinoot = MIDIFunc.noteOn({|vel,note|  
  ~sint = Synth(\synthnaam,{\freq, note.midicps,  
                                \amp, vel/127,  
                                \gate, 1}));  
});
```

Midi-in

MIDIFunc gebruiken

Oplossing: Een array met voor elke noot een positie:

```
~poly = Array.fill(128, 0);  
~midinoot.free;  
~midinoot = MIDIFunc.noteOn({|vel,note|  
  ~poly[note].put(Synth(\synthnaam,{\freq, note.midicps,  
    \amp, vel/127,  
    \gate, 1})));  
});
```

Midi-in

MIDIFunc gebruiken

Poly-noteOff

```
~midinoot.free;
```

```
~midinoot = MIDIFunc.noteOff({|vel,note|
```

```
  ~poly[noot].set(\gate,0);
```

```
});
```

Midi-in

MIDIFunc gebruiken

midi-cc-berichten ontvangen

```
~midicc.free;
```

```
~midicc = MIDIFunc.cc({|...args|
```

```
  args.postln;
```

```
});
```

Midi-in

MIDIFunc gebruiken

midi-cc-berichten ontvangen

```
~midicc.free;  
~midicc = MIDIFunc.cc({|...args|  
  args.postln;  
});
```

resultaat: [1, 54, 0, 1351013728]

[ctl, val, chan, source-id]

OSC

Configuratie

UDP-poort openzetten om berichten op te ontvangen:

```
thisProcess.openUDPPort(9001);
```

Bekijken welke poorten open staan:

```
thisProcess.openPorts;
```

OSC

Data ontvangen

Gebruik een OSCFunc om OSC-data te ontvangen

```
~oscData.free;
```

```
~oscData = OSCFunc({ |...args|
```

```
  args.postln;
```

```
}, '/freq');
```

Resultaat:

```
[ [ /freq, 441 ], 99.340240667, a NetAddr(127.0.0.1, 65109), 9001 ]
```

```
[ osc-bericht , tijd , afzender , poort ]
```

OSC

Data gebruiken

Synth met .set aanpassen

```
~sint = Synth(\test);
```

```
~oscData.free;
```

```
~oscData = OSCFunc({|msg|
```

```
  ~sint.set(\freq,msg[1]);
```

```
},'/freq');
```

Als je meerdere waardes meestuurt, wordt de array `msg` langer