

First steps

```
1 #Bekijk een voorbeeld
2 load_example :filtered_dnb
3
4 #Laad het door het te runnen met Alt + r
5
6 #Kijk eens in de help sectie
```



CONTINUUM

Roel Thijs



Sneltoetsen

In Sonic Pi zijn de sneltoetsen wel wat wennen:

- run	Alt + r
- stop	Alt + s
- copy	Alt + c
- paste	Alt + v
-

CONTINUUM

Roel Thijs



Noten

```
1 play 60      1 play :c4
2 sleep 1      2 sleep 1
3 play 62      3 play :d4
4 sleep 1      4 sleep 1
5 play 64      5 play :e4

1 play_pattern (ring 60,62,64)

1 doremi = (ring 60, 62, 64)
2 lengte = [0.5, 0.5, 1]
3 play_pattern_timed doremi, lengte
```

CONTINUUM

Roel Thijs



Akkoorden



```
1 play 60      1 play :c4
2 play 64      2 play :e4
3 play 67      3 play :g4

1 play [:c4, :e4, :g4]
1 play_chord [:c4, :e4, :g4]
1 play chord(:c4, :major)
```

CONTINUUM

Roel Thijs



Synths

```
1 use_synth :prophet
2 play 60, pan: -0.5, amp: 2, cutoff: 100

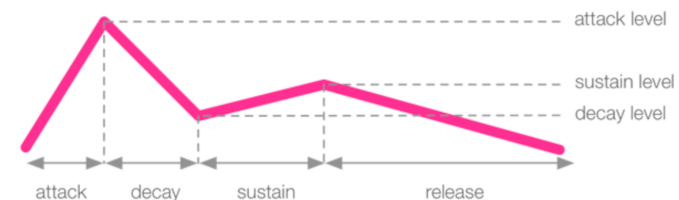
1 use_synth :blade
2 play 60, attack: 0.2, sustain: 1, release: 1
```

CONTINUUM

Roel Thijs



Envelopes



```
1 play 60, attack: 0.1,
2   attack_level: 1, decay: 0.2,
3   decay_level: 0.3, sustain: 1,
4   sustain_level: 0.4, release: 0.5
```

CONTINUUM

Roel Thijs



Loops

```
1 #loop forever
2 loop do
3   play 60
4   sleep 1
5 end

1 #loop n times
2 4.times do
3   play 60
4   sleep 1
5 end

1 #loop n times
2 #met iterator
3 4.times do |i|
4   play 60 + i
5   sleep 1
6 end
```

CONTINUUM

Roel Thijs



Meer loops

```
1 # gebruik in_thread do om
2 # loops tegelijk af te spelen
3 in_thread do
4   loop do
5     play 60
6     sleep 1
7   end
8 end

10 in_thread do
11   loop do
12     play 72
13     sleep 0.25
14   end
15 end

1 use_bpm 120
2
3 in_thread do
4   loop do
5     sample :loop_amen, beat_stretch: 4
6     sleep 4
7   end
8 end

10 in_thread do
11   use_synth :chipbass
12   loop do
13     3.times do |i|
14       play 40 + i
15       sleep 1
16     end
17     play 41
18     sleep 0.5
19     play 40
20     sleep 0.5
21   end
22 end
```

CONTINUUM

Roel Thijs



Live audio

```
1 with_fx :echo do
2   with_fx :octaver do
3     live_audio :micro, input: 1, amp: 5
4   end
5 end
```

CONTINUUM

Roel Thijs



Live loops

```
1 use_bpm 90
2
3 live_loop :beat do
4   4.times do
5     sample :bd_haus, amp: 0.5
6     sleep 0.5
7   end
8 end

10 live_loop :syncedSample do
11   sync :beat #with sync there is no need for sleep
12   sample :drum_heavy_kick, amp: 2
13 end
```

Gebruik live_loop voor "live coding". Je programma blijft lopen terwijl je aanpassingen maakt, ze worden pas actief als je terug 'Alt + r' doet en de huidige loop afloopt.

CONTINUUM

Roel Thijs



Effecten

```
1 with_fx :flanger do
2   with_fx :slicer do
3     live_loop :beat do
4       sample :loop_compus, beat_stretch: 4
5       sleep 4
6     end
7   end
8 end
```

Zet de `with_fx` blokken rond je loops, dat is minder werk voor je cpu

CONTINUUM

Roel Thijs



Samples

```
1 loop do
2   sample :loop_amen
3   sleep sample_duration :loop_amen
4 end

1 loop do
2   sample :loop_amen, beat_stretch: 2
3   sleep 2
4 end

1 loop do
2   sample :loop_amen,
3   onset: rrand_i(0,10),
4   rate: (ring -1,1).choose,
5   attack: 0.1, decay: 0.15
6   sleep 0.25
7 end
```

CONTINUUM

Roel Thijs



Random

```
use_random_seed 123
4.times do
  play rrand_i(50,60) if one_in(3)
  sleep 0.5
end

1 noten = (ring 70, 55, 60, 66)
2 loop do
3   if one_in(4)
4     sample :drum_bass_hard
5     play noten.pick if dice(6)
6   else
7     sample :drum_bass_soft, rate: [1,1,-1].choose
8     play noten
9     sample :loop_amen, onset: rrand_i(0,10)
10  end
11  sleep 1
```



CONTINUUM

Roel Thijs



MIDI / OSC

```
1 live_loop :midi_piano do
2   use_real_time #removes the lag between key press and sound
3   note, velocity = sync "/midi:launchkey_mini_midi_1:2:1/note_on"
4   synth :prophet, note: note, amp: velocity / 127.0
5 end

1 live_loop :midi_triggers do
2   use_real_time
3   note, velocity = sync "/midi:*10/note_on"
4   sample :ambi_choir if note == 36
5   sample :ambi_drone if note == 37
6 end
```



CONTINUUM

Roel Thijs



Meer MIDI

```
1 set :flagBD, 1
2
3 live_loop :midi_triggers do
4   note, velocity = sync "/midi:*10/note_on"
5   if note == 36
6     set :flagBD, get[:flagBD] * -1
7   end
8 end
9
10 live_loop :bassdrum do
11   if get[:flagBD] == 1
12     sample :bd_haus
13   else
14     sample :bd_mehackit
15   end
16   sleep 0.5
17 end
```



CONTINUUM

Roel Thijs



Functies

```
1 define :playHabanera do |repeats|
2   notes = (ring :d, :r, :r, :a, :f5, :r, :a, :r)
3   repeats.times do
4     notes.size.times do
5       play notes.tick - 12 #oktaaf lager
6       sleep 0.25
7     end
8   end
9 end
10
11 playHabanera 2
```



CONTINUUM

Roel Thijs

