

Fyxxi

Workshop

Sonic Pi



| www.fyxxi.be

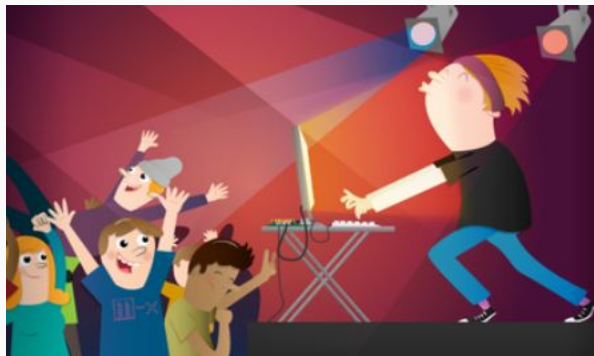


| rpf.io



Overzicht Workshop

- Wat is Sonic Pi?
- Hoe gebruik je Sonic Pi?
- Muziek maken
- “Live Coding”



“You can easily use Sonic Pi tot teach a range of basic computing concepts”

The MagPi Educator's Edition



| www.fyxxi.be

rpf.io |



Wat is Sonic Pi?

Sonic Pi is een live programmeeromgeving op basis van Ruby , oorspronkelijk ontworpen ter ondersteuning van computer- als muzieklessen op scholen. De software werd ontwikkeld door Sam Aaron in de Universiteit van Cambridge in samenwerking met de Raspberry Pi Foundation .

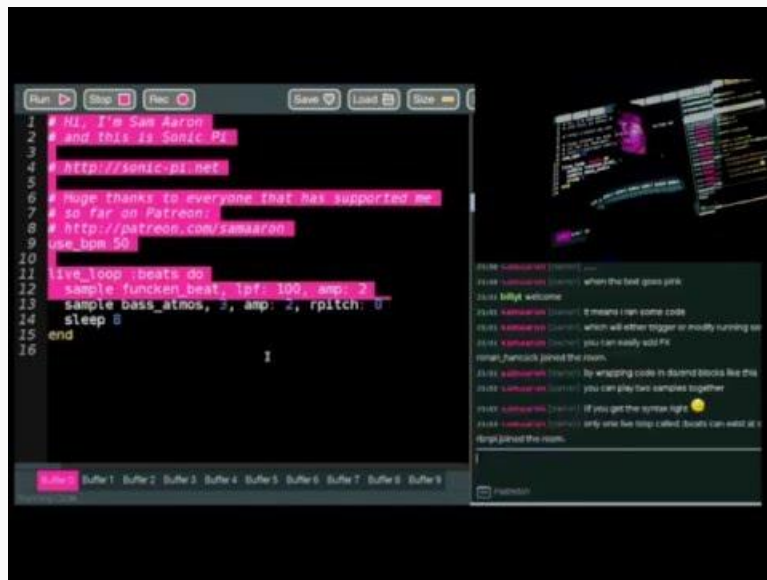


| www.fyxxi.be

rpf.io |



Sonic Pi creator: Sam Aaron



The screenshot shows the Sonic Pi IDE interface. The main editor displays a script with the following content:

```
1 # Hi, I'm Sam Aaron
2 # and this is Sonic Pi
3
4 # http://sonic-pi.net
5
6 # Huge thanks to everyone that has supported me
7 # so far on Patreon:
8 # http://patreon.com/samaaron
9 use_bpm 50
10
11 live_loop :beats do
12   sample :funcken_beat, :pf, 100, amp: 2
13   sample :bass_atmos, 3, amp: 2, rpitch: 0
14   sleep 0
15 end
```

Below the script, there is a live performance view showing a 3D visualization of the sound. The interface includes buttons for Run, Stop, Rec, Save, Load, and Size.



The screenshot shows the Sonic Pi IDE interface with a more complex script. The main editor displays a script with the following content:

```
20 with_fx :lfo, wave: :s, phase: 0, cutoff_min: 0, cutoff_max: 120, res: 0.5, amp: 1 do
21   with_fx :lfo, wave: :s do
22     fmk
23   end
24 end
25
26 live_loop :chords do
27   sync :chords
28   tick
29
30   with_fx :lfo, amp: 0.2 do
31     sample :lfo, amp: 2 if spread(1, 10).look
32     sample :lfo, amp: 2.5 if spread(1, 12).rotated(1).look
33     synth :chords, release: 0.1, cutoff: 100, env_curve: 2, amp: 1 if spread(1, 10).rotated(1).look
34     synth :chords, release: 0.1, cutoff: 120, env_curve: 2, amp: 0.5 if spread(1, 12).look
35     sleep 0.125
36   end
37 end
38
39 live_loop :chords do
40   sync :chords
41   shaps = 1
42 end
```

Below the script, there is a live performance view showing a 3D visualization of the sound. The interface includes buttons for Run, Stop, Rec, Save, Load, and Size.



Wat heb je nodig?

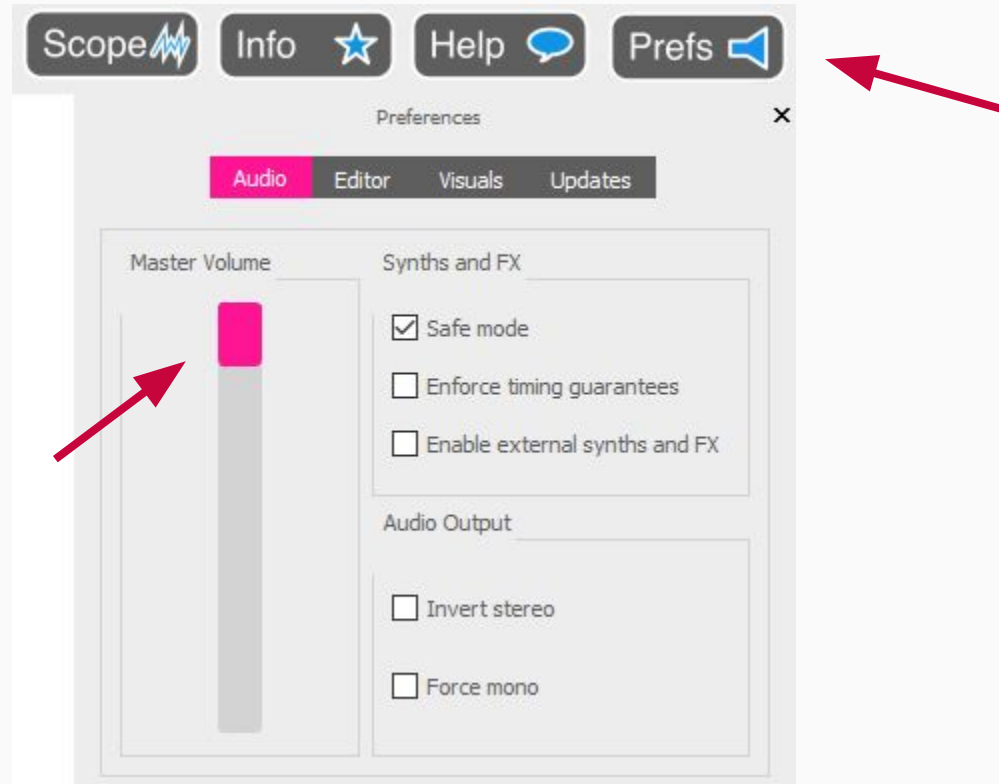


| www.fyxxi.be

rpf.io |



Volume verhogen



Play

Run



- Selecteer Buffer 1 en tik in:

```
play 60
```

- Wat gebeurt er met de volgende code?

```
pley 60
```

- Wat gebeurt er met de volgende code?

```
play 60  
play 67  
play 69
```



Sleep

- Maak gebruik van 'sleep'.

```
play 60  
sleep 1  
play 67  
sleep 1  
play 69  
sleep 1
```

- Experimenteer met de duur.

```
sleep 0.5
```



Maak je eigen song!

Loop

- Selecteer Buffer 2 en tik in. Welk liedje herken je?

```
play 60  
sleep 0.5  
play 62  
sleep 0.5  
play 64  
sleep 0.5  
play 60  
sleep 0.5
```



Loop

- Dit eerste deel herhalen we 2 keer.

```
2.times do
  play 60
  sleep 0.5
  play 62
  sleep 0.5
  play 64
  sleep 0.5
  play 60
  sleep 0.5
end
```

Bepaalde code is ingesprongen met 2 spaties.



Oneindige Loop

- Wat als je een fragment continu wil laten herhalen?

```
loop do  
  play 60  
  sleep 0.5  
end
```



Muzieknoten

Je kan ook de standaard notatie gebruiken. Plaats volgende code in 2 nieuwe buffers. Hoor je een verschil?

play 60	play :c4
sleep 1	sleep 1
play 62	play :d4
sleep 1	sleep 1
play 64	play :e4
sleep 1	sleep 1
play 65	play :f4
sleep 1	sleep 1
play 67	play :g4
sleep 1	sleep 1
play 69	play :a4
sleep 1	sleep 1
play 71	play :b4
sleep 1	sleep 1



Verander het geluid

Het is tijd om het geluid van de muziek wat interessanter te maken. We kunnen dit doen door het synthesizer geluid van je liedje te veranderen. De standaard sound van Sonic-Pi is 'beep'.

```
use_synth :fm  
2.times do  
  play 60  
  sleep 0.5  
  play 67  
  sleep 0.5  
end
```

Om een ander geluid te gebruiken, zet je deze code bovenaan op de eerste regel.

Hier is 'FM' de naam van het geluid.



Geluiden uitproberen

Er zitten veel verschillende soorten sounds in Sonic-Pi. Ontdek ze zelf!



Detuned Pulse Wave

note:	52	amp:	1	pan:	0	attack:	0
decay:	0	sustain:	0	release:	1	attack_level:	1
decay_level:	sustain_level	sustain_level:	1	env_curve:	2	cutoff:	10
detune:	0.1	pulse_width:	0.5	dpulse_width:	pulse_width		

```
use_synth :dpulse
```



Samples

Je kan ook muziek creëren met samples. Samples zijn melodieën die al voor je opgenomen zijn. Het is een hele makkelijke manier om leuke liedjes te maken. Gebruik hiervoor de volgende code.

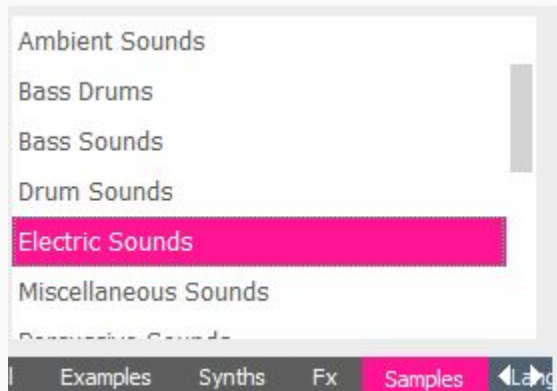
```
sample :naam van de sample
```

```
2.times do  
  sample :loop_amen  
  sleep 1.753  
end
```



Samples uitproberen

Er zitten veel verschillende samples in Sonic-Pi. Ontdek ze zelf!



Electric Sounds

amp:	1	pre_amp:	1
decay:	0	sustain:	-1
lpf_attack:	attack	lpf_decay:	decay
lpf_init_level:	lpf_min	lpf_attack_level:	lpf_decay_level
lpf_release_level:	lpf	lpf_env_curve:	2
hpf_attack:	attack	hpf_sustain:	sustain



2 tunes tegelijkertijd

Muziek heeft vaak een herhalende achtergrondtrack, waarbij een afzonderlijke melodie wordt gespeeld. Tot nu toe heb je in Sonic Pi één deuntje gespeeld. Laten we proberen twee tunes tegelijkertijd spelen!

Maak hiervoor gebruik van de volgende code:

```
in_thread do
end
```



2 tunes tegelijkertijd

- Open een nieuwe buffer

```
in_thread do
  loop do
    sample :loop_amen
    sleep 1.753
  end
end
```

```
in_thread do
  16.times do
    play 75
    sleep 1.753
    play 74
    sleep 0.25
  end
end
```

Deze eerste 'thread' zal fungeren als de melodie van je muziek.

Deze tweede 'thread' is de onderliggende muzieklijn.



Live code!

Met Sonic-Pi kun je ook live muziek programmeren. Let's go!

- Open een nieuwe buffer en gebruik de **live_loop**!

```
live_loop :beats do
  sample :bd_haus
  sleep 0.5
end
```

- Run (ALT+R) de code en pas aan!
- Sleep is noodzakelijk!



Live code!

Ook meerdere live_loops naast elkaar zijn mogelijk.

- Open een nieuwe buffer.

```
live_loop :foo do
  play 60
  sleep 0.5
end
live_loop :bar do
  sample :bd_haus
  sleep 1
end
```



Effecten toevoegen

- Een van de meest lonende en leuke aspecten van Sonic Pi is de mogelijkheid om gemakkelijk studio-effecten aan uw geluiden toe te voegen.

```
with_fx :reverb do  
  sample :guit_e_fifths  
end
```



Effecten toevoegen

- Of effecten toevoegen aan een ander effect!

```
with_fx :reverb do  
  with_fx :distortion do  
    sample :guit_e_fifths  
  end  
end
```



Effecten uitproberen

Er zitten veel verschillende effecten in Sonic-Pi. Ontdek ze zelf!



Echo

amp:	1	mix:	1	pre_mix:	1	pre_amp:	1
phase:	0.25	decay:	2	max_phase:	2		

```
with_fx :echo do  
  play 50  
end
```

Standard echo with variable phase duration (time between echoes) and de...



Parameters wijzigen

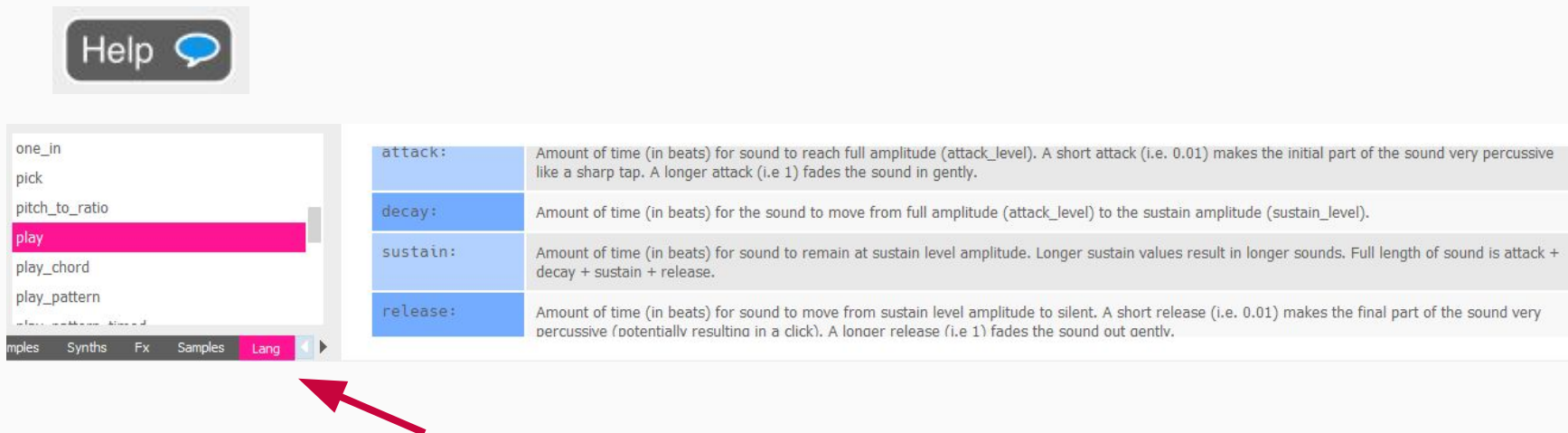
- Soms wilt u geluiden misschien langer of met een ander tempo laten spelen. Dit kan eenvoudig worden bereikt door de parameters van de code die u gebruikt te wijzigen.

```
play 60, attack: 1, release:3
```



Parameters uitproberen

Er zitten veel verschillende parameters in Sonic-Pi. Ontdek ze zelf!



Help

one_in
pick
pitch_to_ratio
play
play_chord
play_pattern

mples Synthz Fx Samples **Lang**

attack:	Amount of time (in beats) for sound to reach full amplitude (attack_level). A short attack (i.e. 0.01) makes the initial part of the sound very percussive like a sharp tap. A longer attack (i.e 1) fades the sound in gently.
decay:	Amount of time (in beats) for the sound to move from full amplitude (attack_level) to the sustain amplitude (sustain_level).
sustain:	Amount of time (in beats) for sound to remain at sustain level amplitude. Longer sustain values result in longer sounds. Full length of sound is attack + decay + sustain + release.
release:	Amount of time (in beats) for sound to move from sustain level amplitude to silent. A short release (i.e. 0.01) makes the final part of the sound very percussive (potentially resulting in a click). A longer release (i.e 1) fades the sound out gently.

Ook bij 'samples' vind je tal van parameters terug zoals bv. cutoff:, pan:, rate: of amp:.



Maak je eigen song!

Note	Octave										
	-1	0	1	2	3	4	5	6	7	8	9
C	0	12	24	36	48	60	72	84	96	108	120
C#	1	13	25	37	49	61	73	85	97	109	121
D	2	14	26	38	50	62	74	86	98	110	122
D#	3	15	27	39	51	63	75	87	99	111	123
E	4	16	28	40	52	64	76	88	100	112	124
F	5	17	29	41	53	65	77	89	101	113	125
F#	6	18	30	42	54	66	78	90	102	114	126
G	7	19	31	43	55	67	79	91	103	115	127
G#	8	20	32	44	56	68	80	92	104	116	
A	9	21	33	45	57	69	81	93	105	117	
A#	10	22	34	46	58	70	82	94	106	118	
B	11	23	35	47	59	71	83	95	107	119	

*Maak gebruik van deze tabel om
bestaande liedjes om te zetten
naar de programmeertaal van
Sonic-Pi.*

Opslaan



Meer weten?



Welkom vriend :-)

Welkom bij Sonic Pi. Hopelijk ben je net zo enthousiast als ik ben, synthese, programmeren, compositie en live spelen gaat leren en

Maar wacht, dat was onbeleefd van mij! Ik moet mij nog voorstellen [@samaaron op Twitter en ik zal heel blij zijn om je daar te kunnen Performances] (<http://facebook.com/livecoderssamaaron>) waar ik,

Mocht ie bedenkingen hebben. of ideeën. om Sonic Pi te verbeteren

