

Samsara

Jerod Sommerfeldt

sam·sa·ra /səm'sɑrə/

:the indefinitely repeated cycles of birth, death, and misery caused by karma

:conceived of as having no perceptible beginning or end

:unbroken chain that follows continuously

Performance Directions

Samsara is a work for vibraphone and interactive electronics.

For the percussion:

Yarn mallets and two bass or cello bows are required. Tapping with finger pads is also required.

The motor should remain off for the duration of the piece.

The score is unmeasured, but there are 5" divisions in each system. Each page lasts for 60".

Rhythms are approximate and bowings are held for the duration of each straight line.

Trills and bowed tremolos are also used within the work.

For the electronics:

One [or two] microphone(s) are needed for the vibraphone, running to an audio interface.

Only stereo playback is needed, though the percussionist will likely want a stage monitor speaker.

A laptop running RTcmix and/or Pure Data is required.

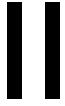
Both the RTcmix code and Pure Data patch are available from the composer.

1000000

Jerod Sommerfeldt

The image displays three staves of musical notation for a vibraphone, each with a treble and bass clef. The notation includes various techniques and dynamic markings:

- Staff 1:**
 - Technique:** "tap with fingers" (first measure), "mallets" (second measure).
 - Dynamic Markings:** *pp*, *mf*, *pp*, *f*, *ppp*, *mp*.
 - Pedal:** "ped" with a line spanning the first two measures.
- Staff 2:**
 - Technique:** "bowed" (second measure), "tap with fingers" (third measure).
 - Dynamic Markings:** *ff*, *p*, *mp*, *f*.
 - Pedal:** "ped" with a line spanning the first two measures.
- Staff 3:**
 - Technique:** "bowed" (first measure), "mallets" (second measure).
 - Dynamic Markings:** *p*, *mp*, *pp*.
 - Pedal:** "ped" with a line spanning the first two measures.



mallets

bowed

vibes

ped

mallets

bowed

rapid

vibes

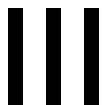
ped

ped

tap with fingers

vibes

ped



mallets

vibes

pp *mf* *pp* *f*

ped

ped

vibes

mp *pp* *pp* *pp* *pp* *p* *mf*

ped

ped

bowed

mallets

vibes

ppp

rapid

ped

IV

bowed

vibes

ped

mallets

vibes

bowed

vibes

ped

V

bowed

vibes

pp *ppp*

ped

vibes

mp *mf*

ped

mallets

vibes

ppp *mp*

ped


```

1  load("DELAY")
2  load("SPECTACLE2")
3  rtinput("AUDIO")
4
5  //Samsara master scorefile, playable from Terminal window or Pure Data
6  ////////////////////////////////////// I
7
8  srand()
9  inskip = 0
10 amp = 9.0
11 // envelope tables read from DC offset! Will need to change path for performances...
12
13 env1 = maketable("textfile", 1000, "/Users/jerod_s/Desktop/Caccia/envtable.txt")
14 deltime1 = 0.14
15 feedback1 = 0.7
16 ringdur1 = 3.5
17 inchan = 0
18
19 pan1 = maketable("random", 1000, "low", 0.0,0.5) // more toward L
20
21 DELAY(0, inskip, 300, amp*env1, deltime1, feedback1, ringdur1, inchan, pan1)
22
23 start1a = irand(1.0, 4.0)
24 inskip = 0
25 amp = 9.0
26 env2 = maketable("textfile", 1000, "/Users/jerod_s/Desktop/Caccia/envtable.txt")
27 deltime2 = 1.4
28 feedback2 = 0.3
29 ringdur2 = 1.0
30 pan2 = maketable("random", 1000, "high", 0.5,1.0) // more toward R
31
32 DELAY(0, inskip, 300, amp*env2, deltime2, feedback2, ringdur2, inchan, pan2)
33
34 ////////////////////////////////////// II
35
36 inchan = 0
37 inskip = 0
38 ringdur = 15 // play after indur elapses, while delay lines flush
39 amp = 5.0
40 wet = 0.8
41
42 fftlen = 1024 // yielding 512 frequency bands
43 winlen = fftlen * 2 // the standard window length is twice FFT size
44 overlap = 2 // 2 hops per fftlen (4 per window)
45 window = 0 // use Hamming window
46
47 // input envelope (spanning <indur>)
48 ienv = maketable("line", 1000, 0,0, 1,1, 19,1, 20,0)
49
50 // output envelope (spanning <indur> + <ringdur>)
51 oenv = maketable("curve", 1000, 0,1,0, 2,1,-1, 3,0)
52
53 eqtablen = fftlen / 2
54 mineqfreq = 0
55 maxeqfreq = 0
56
57 // EQ curve: -90 dB at 0 Hz, ramping up to 0 dB at 200 Hz, etc.
58 eq = maketable("line", "nonorm", eqtablen, 0,-90, 200,0, 8000,-3, 22050,-6, 44100, 0)
59
60 deltablen = fftlen / 2
61 mindelfreq = 0
62 maxdelfreq = 0
63
64 mindt = .4 // Delay times
65 maxdt = 3
66 seed = srand()
67 deltime = maketable("random", "nonorm", deltablen, "even", mindt, maxdt, seed)
68
69 minfb = .1 // Feedback times
70 maxfb = .8
71 fbtime = maketable("random", "nonorm", deltablen, "even", minfb, maxfb, seed)
72
73 pan = makeLFO("saw", 0.5, 0, 1) // sine for smooth, saw for clicks (later on)
74
75 SPECTACLE2(60, inskip, 240, amp*oenv, ienv, ringdur, fftlen, winlen,
76 window, overlap, eq, deltime, fbtime, mineqfreq, maxeqfreq,
77 mindelfreq, maxdelfreq, 0, wet, inchan, pan)
78
79 ////////////////////////////////////// III
80
81 inchan = 0
82 inskip = 0
83 ringdur = 15 // play after indur elapses, while delay lines flush
84 amp = 8.0
85 wet = 0.8
86
87 fftlen = 1024 // yielding 512 frequency bands
88 winlen = fftlen * 2 // the standard window length is twice FFT size
89 overlap = 2 // 2 hops per fftlen (4 per window)
90 window = 0 // use Hamming window
91
92 // input envelope (spanning <indur>)
93 ienv = maketable("line", 1000, 0,0, 1,1, 19,1, 20,0)
94
95 // output envelope (spanning <indur> + <ringdur>)
96 oenv = maketable("curve", 1000, 0,1,0, 2,1,-1, 3,0)
97
98 eqtablen = fftlen / 2
99 mineqfreq = 0
100 maxeqfreq = 0
101
102 // EQ curve: -90 dB at 0 Hz, ramping up to 0 dB at 200 Hz, etc.
103 eq = maketable("line", "nonorm", eqtablen, 0,-90, 200,0, 8000,-3, 22050,-6, 44100, 0)
104
105 deltablen = fftlen / 2
106 mindelfreq = 0
107 maxdelfreq = 0
108
109 mindt = .4 // Delay times
110 maxdt = 3
111 seed = srand()
112 deltime = maketable("random", "nonorm", deltablen, "even", mindt, maxdt, seed)
113
114 minfb = .1 // Feedback times
115 maxfb = .8
116 fbtime = maketable("random", "nonorm", deltablen, "even", minfb, maxfb, seed)
117 LFOfreq = maketable("random", "nonorm", 1000, "even", 0.1, 9.0, seed)
118 pan = makeLFO("saw", LFOfreq, 0, 1) // sine for smooth, saw for clicks (later on)
119
120 SPECTACLE2(180, inskip, 120, amp*oenv, ienv, ringdur, fftlen, winlen,
121 window, overlap, eq, deltime, fbtime, mineqfreq, maxeqfreq,
122 mindelfreq, maxdelfreq, 0, wet, inchan, pan)
123
124 ////////////////////////////////////// IV-V
125
126 inchan = 0
127 inskip = 0
128 ringdur = 15 // play after indur elapses, while delay lines flush
129 amp = 8.0
130 wet = 0.8
131
132 fftlen = 1024 // yielding 512 frequency bands
133 winlen = fftlen * 2 // the standard window length is twice FFT size
134 overlap = 2 // 2 hops per fftlen (4 per window)
135 window = 0 // use Hamming window
136
137 // input envelope (spanning <indur>)
138 ienv = maketable("line", 1000, 0,0, 1,1, 19,1, 20,0)
139
140 // output envelope (spanning <indur> + <ringdur>)
141 oenv = maketable("curve", 1000, 0,1,0, 2,1,-1, 3,0)
142
143 eqtablen = fftlen / 2
144 mineqfreq = 0
145 maxeqfreq = 0
146
147 // EQ curve: -90 dB at 0 Hz, ramping up to 0 dB at 200 Hz, etc.
148 eq = maketable("line", "nonorm", eqtablen, 0,-90, 200,0, 8000,-3, 22050,-6, 44100, 0)
149
150 deltablen = fftlen / 2
151 mindelfreq = 0
152 maxdelfreq = 0
153
154 mindt = .4 // Delay times
155 maxdt = 3
156 seed = srand()
157 deltime = maketable("random", "nonorm", deltablen, "even", mindt, maxdt, seed)
158
159 minfb = .1 // Feedback times
160 maxfb = .8
161 fbtime = maketable("random", "nonorm", deltablen, "even", minfb, maxfb, seed)
162 LFOfreq = maketable("random", "nonorm", 1000, "high", 5.0, 18.0, seed)
163 pan = makeLFO("saw", LFOfreq, 0, 1) // sine for smooth, saw for clicks (later on)
164
165 SPECTACLE2(180, inskip, 120, amp*oenv, ienv, ringdur, fftlen, winlen,
166 window, overlap, eq, deltime, fbtime, mineqfreq, maxeqfreq,
167 mindelfreq, maxdelfreq, 0, wet, inchan, pan)

```