

Bipscript



A Domain-Specific Scripting Language for Interactive Music

Yet Another Music DSL??

- Object-oriented scripting language
- Emphasis on *interactive sequence generation*
- Native objects for MIDI, OSC, audio features
- Single-threaded event model
- Analogous to web development



Squirrel Language

- Inspired by the Lua language
- Designed for real-time use e.g. gaming
- Scripting language for e.g. Portal2, L4D2
- Bipscript = custom Squirrel interpreter
- Bipscript also adds a standard class library



Class Library

- Objects for inputs/outputs: audio, MIDI, OSC
- Create networks of audio/MIDI plugins
- Methods for scheduling events
- Event handlers for reacting to external events
- Parse embedded ABC notation, MML, Tablature



Listen and Update State

Create an audio input and onset detector, count audio onsets:

```
local input = Audio.Input("input")
local detector = Audio.OnsetDetector()
detector.threshold(0.9)
detector.connect(input)

local onsetPower = 0
detector.onOnset(function(pos) {
    onsetPower++
})
```

Sequence Output

```
local output = Midi.Output("output")
function sequence(m) {
    local division = 4
    if(onsetPower > 3) division = 8
    if(onsetPower > 5) division = 16
    local len = 1.0 / division
    local note = Midi.Note(36, 127, len)
    for(local i = 0; i < division; i++) {
        output.schedule(note, m + i/division)
    }
    onsetPower = 0
}
```

Just-in-Time Sequencing

This example *sequence* method adds quarter notes, eighth notes or sixteenth notes based on how many onsets detected. The *sequence* method itself is scheduled at three quarters past the beginning of a measure ($m + 0.75$) to schedule for the following measure ($m + 1$):

```
for(local m = 1; m <= 16; m++) {
    Transport.schedule(
        @(measure) sequence(measure),
        m + 0.75, m + 1)
}
```

Interactive Algorithmic Compositions

- “Listen” via OSC, MIDI, extracted audio features
- Update internal state based on this input
- Use the state when generating JIT sequences

Utility Scripts for Live Performers

- Headless custom sample/MIDI player
- Trigger samples and sequences via MIDI, OSC
- Click-track, OSC sequencer, automated mixer..

Live Coding

- Scripts can reload themselves while running
- Heavy objects, e.g. plugins, are cached
- Multiple scripts can share common transport