

# Praxis LIVE

*open-source hybrid visual IDE for live creative coding*

Neil C Smith

Artist & Technologist

Oxford, UK

Worked for 10yrs in web consultancy for culture, education  
and third-sector

Also over 15yrs in creative applications of Java

Now, freelancing related to open-source Java technologies

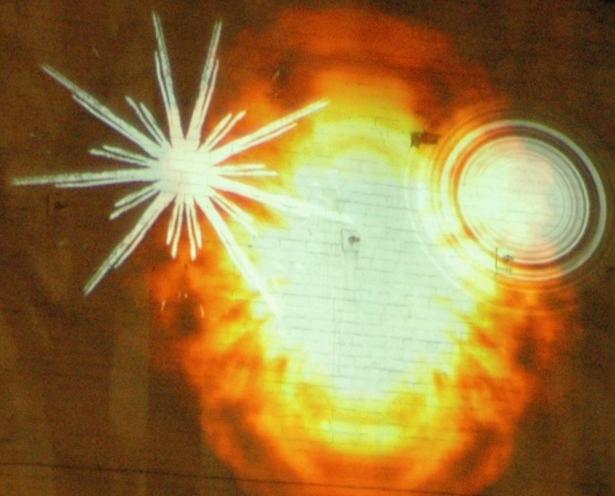
- Praxis LIVE - hybrid visual IDE for live creative coding
- GStreamer 1.x bindings
- JAudioLibs - audio utilities (inc. JNAJack)
- Apache NetBeans

Audio-visual performances

Generative / interactive spaces and projections



















```
import Live
from random import randint

#(1) ADC input 0 (Analog) -> table 0
#(2) AudioOut 0x2
#(3) LED driver

Kvartza
public void setup() {
    audio.freq = 0.02;
    audio.xform((k),frequency/freq / 4);
    audio.smooth(k, noise);
}

void loop() {
    if (k > 700) {
        k = 0;
    }
}
```



hybrid visual IDE for (live) creative coding

hybrid visual IDE *and runtime* for (live) creative coding

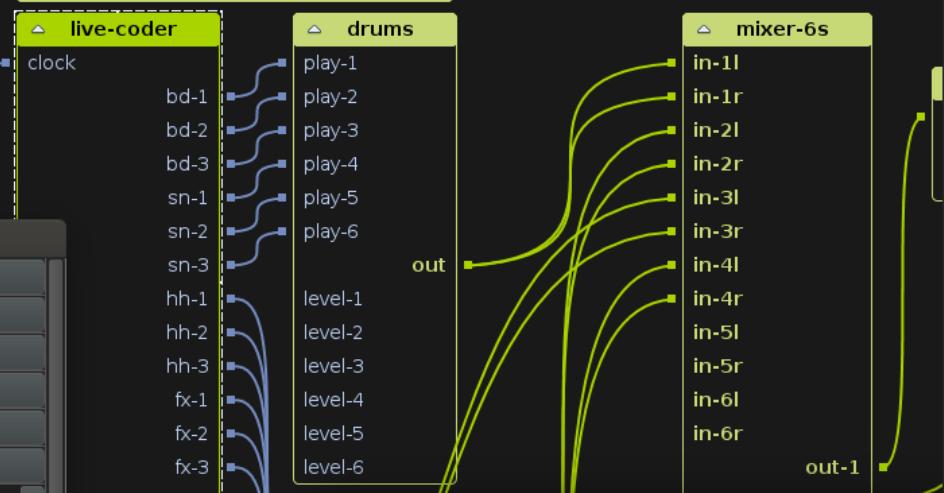
easily create projections, interactive spaces, custom VJ tools,  
sonic instruments, creative IoT

use it as a live digital sketchbook, test out ideas, experiment  
with code

Window Help

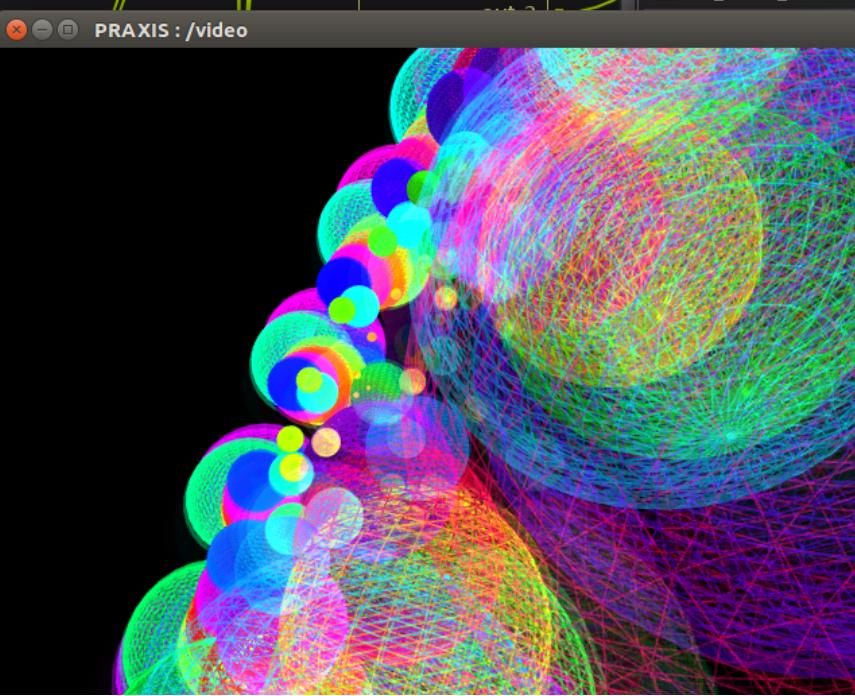
26

Edit the code of this component, uncommenting the example lines and saving to see how they sound.



□ live coder code.java ×

live\_coder\_code > setup >



```
ateY(sin(millis() * 0.0004));
ateZ(sin(millis() * 0.0005));
ele(abs(sin(millis() * 0.0001)) * 2 + 0.8);
- (int i = 0; i < numCircles; i++) {
  stroke( (i * 7* abs(t) % 255), 255, 230, 100);
  noFill();
  pushMatrix();
  rotate(radians((t + 37) * i));
  translate(sqrt(c*i), sqrt(c*i), dB(fft[((i/10)%fft.length)));
    rect(-width/4,0,width/2,i * 0.05 % 1);
    stroke(255);
  sphere(i * 0.5 % 64);
  popMatrix();
}

dB(float x) {
  if (x == 0) {
    return 0;
  } else {
    return 10 * (float) Math.log10(x);
  }
}
```

# Key features

- Intuitive graphical patching
- Extend at runtime
- Real-time audio / video (OpenGL, GStreamer, JACK)
- MIDI, OSC, GUIs, physical computing
- Create standalone projects
- Cross-platform (Windows, MacOS, Linux ... *Pi!*)

# Free and open source

built on top of many great projects, including

- Processing
- GStreamer 1.x
- Apache NetBeans platform and IDE

Alt-TAB now!

# History

Praxis LIVE v1 (2011)

Custom OpenGL pipeline, limited coding components

Praxis LIVE v2 (2015)

Major rewrite, Processing included,

*Fork All* - custom component coding and real editor

Praxis LIVE v3 (2017)

Java 8, Processing 3, Syphon/Spout, Pi support, third-party  
libs

Praxis LIVE v4 (2018)

Praxis CORE LGPL, recode anything, bezier connections!

# Cyber-physical coding

Real-time programming of real-time systems

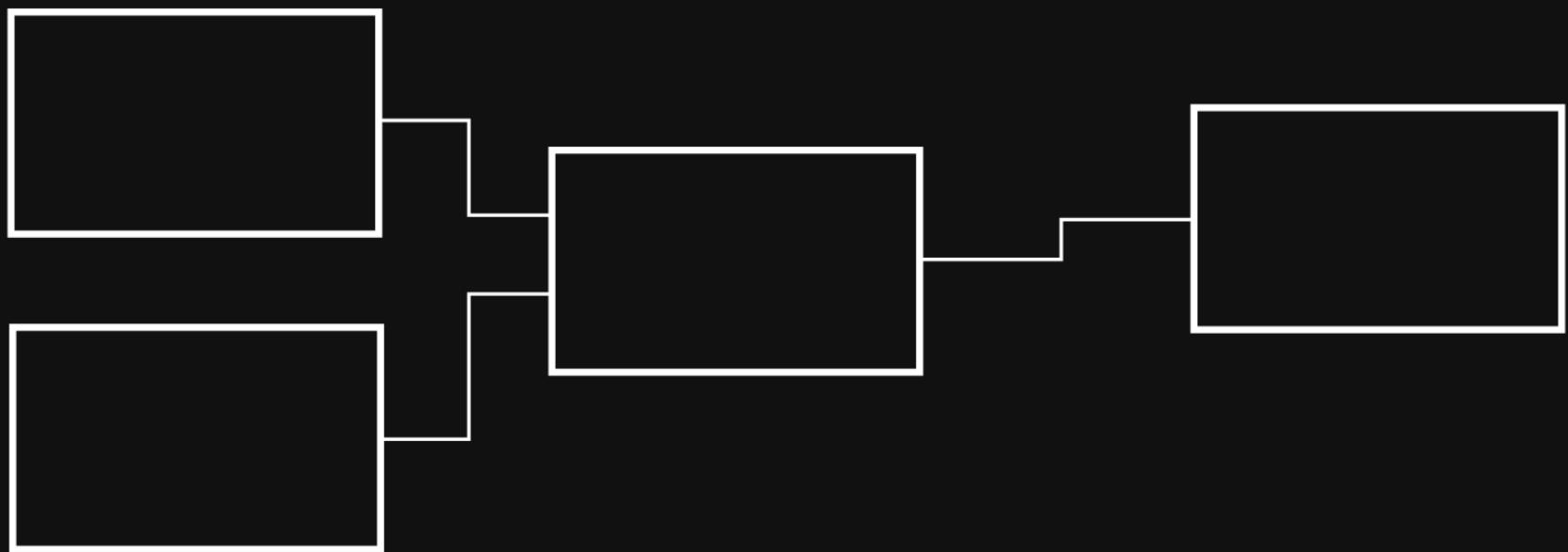
# User code as a first class citizen

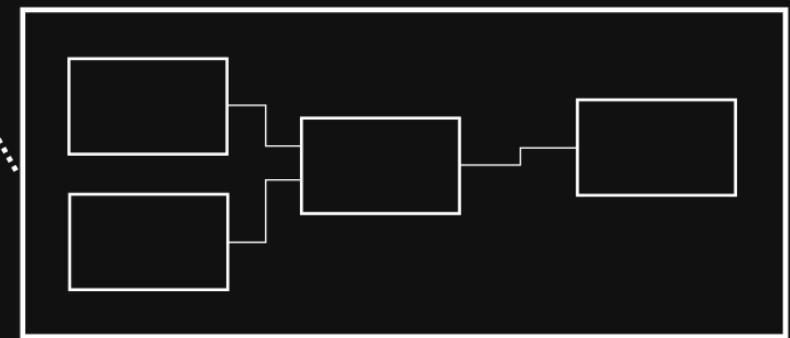
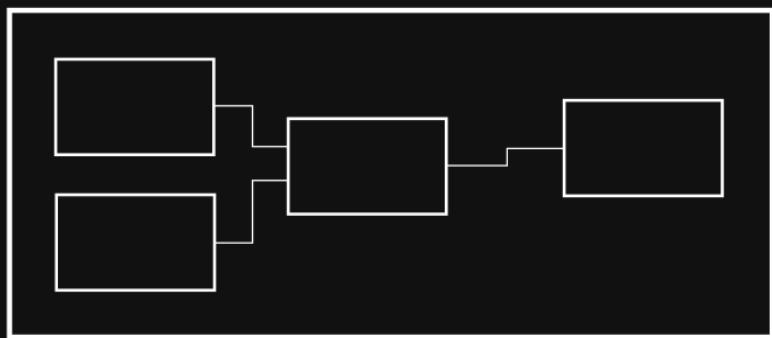
Turtles all the way down

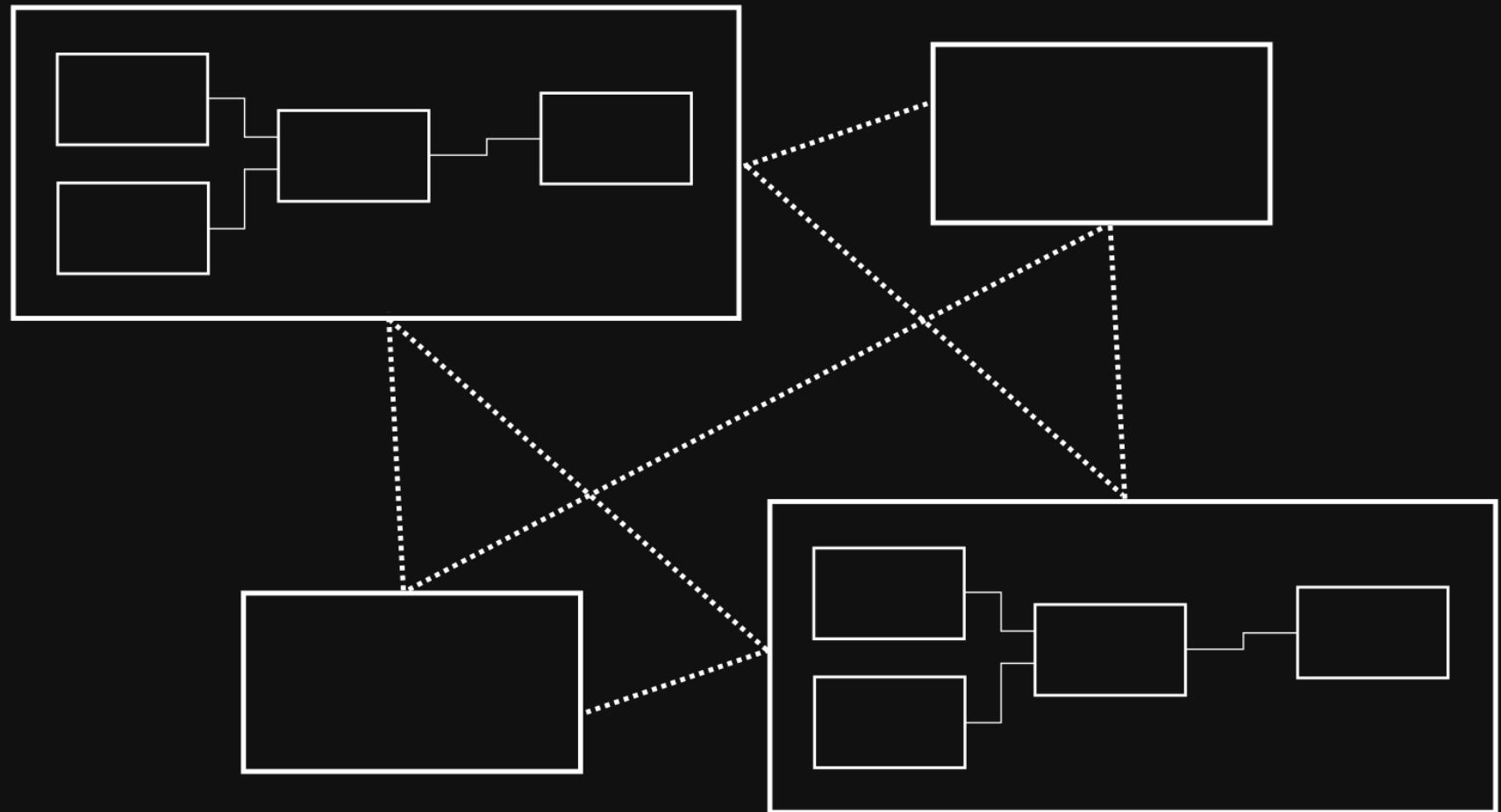
No scripting components, complex plugin API, etc.

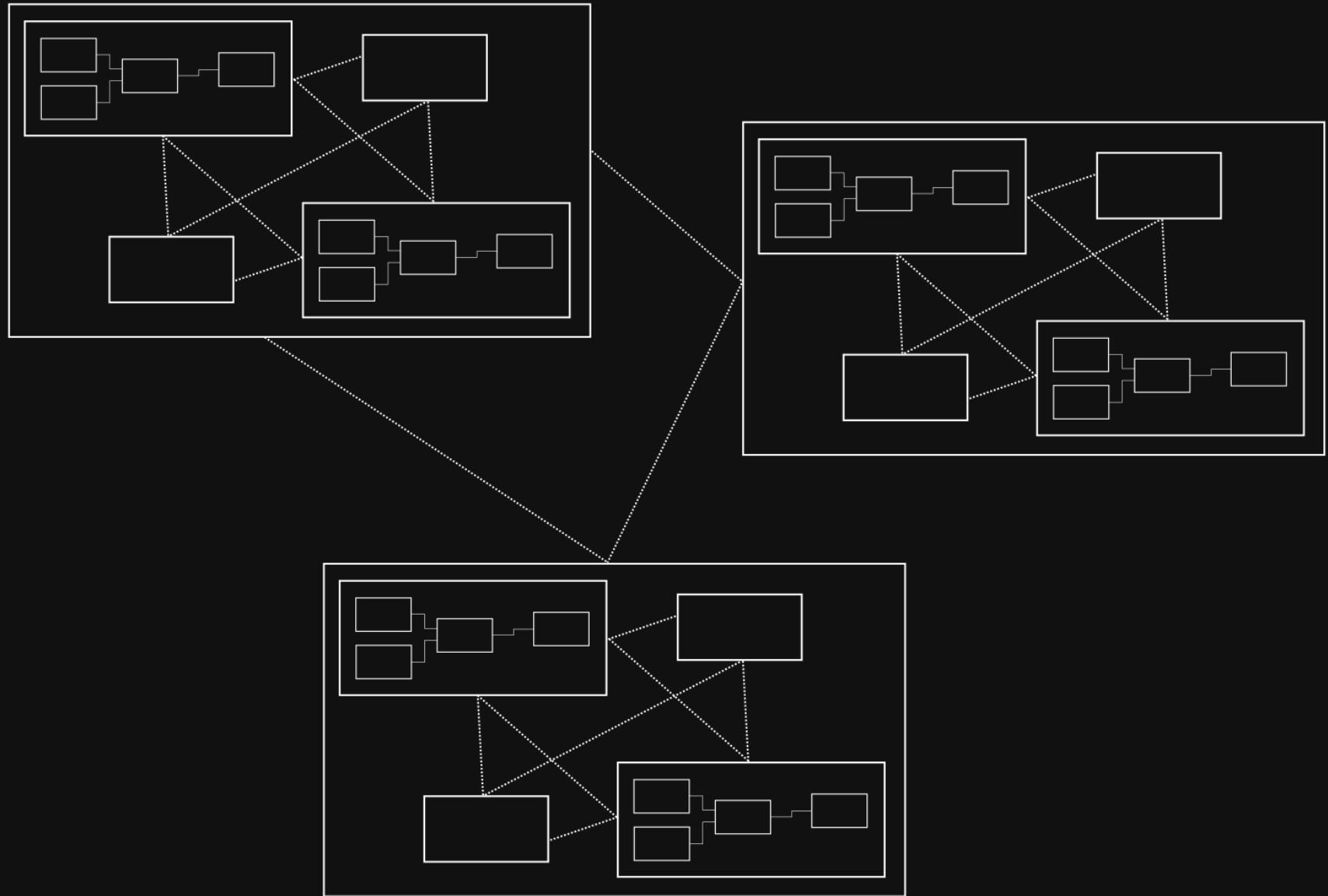
Code down to the sample

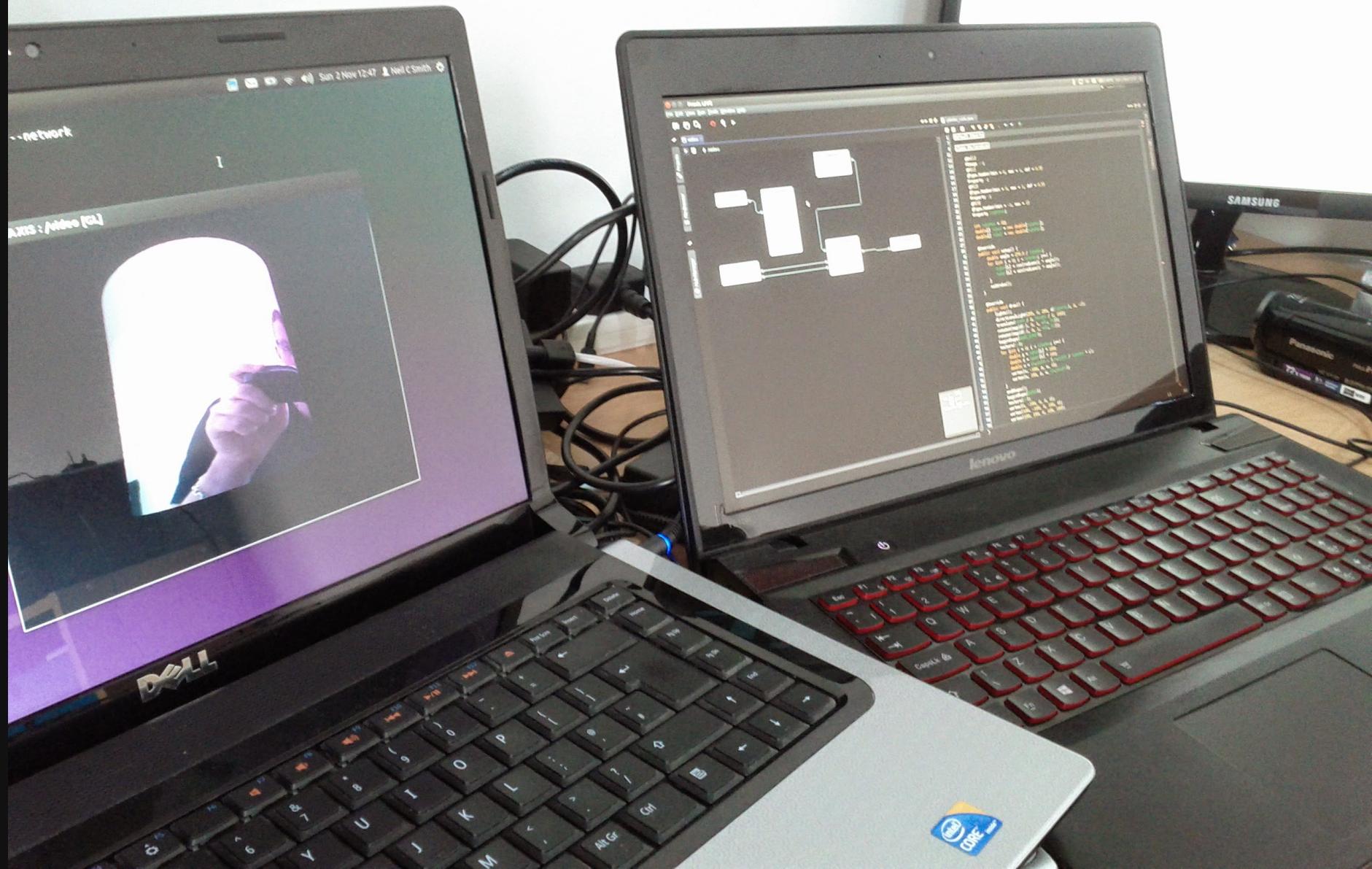
# Forest of Actors architecture

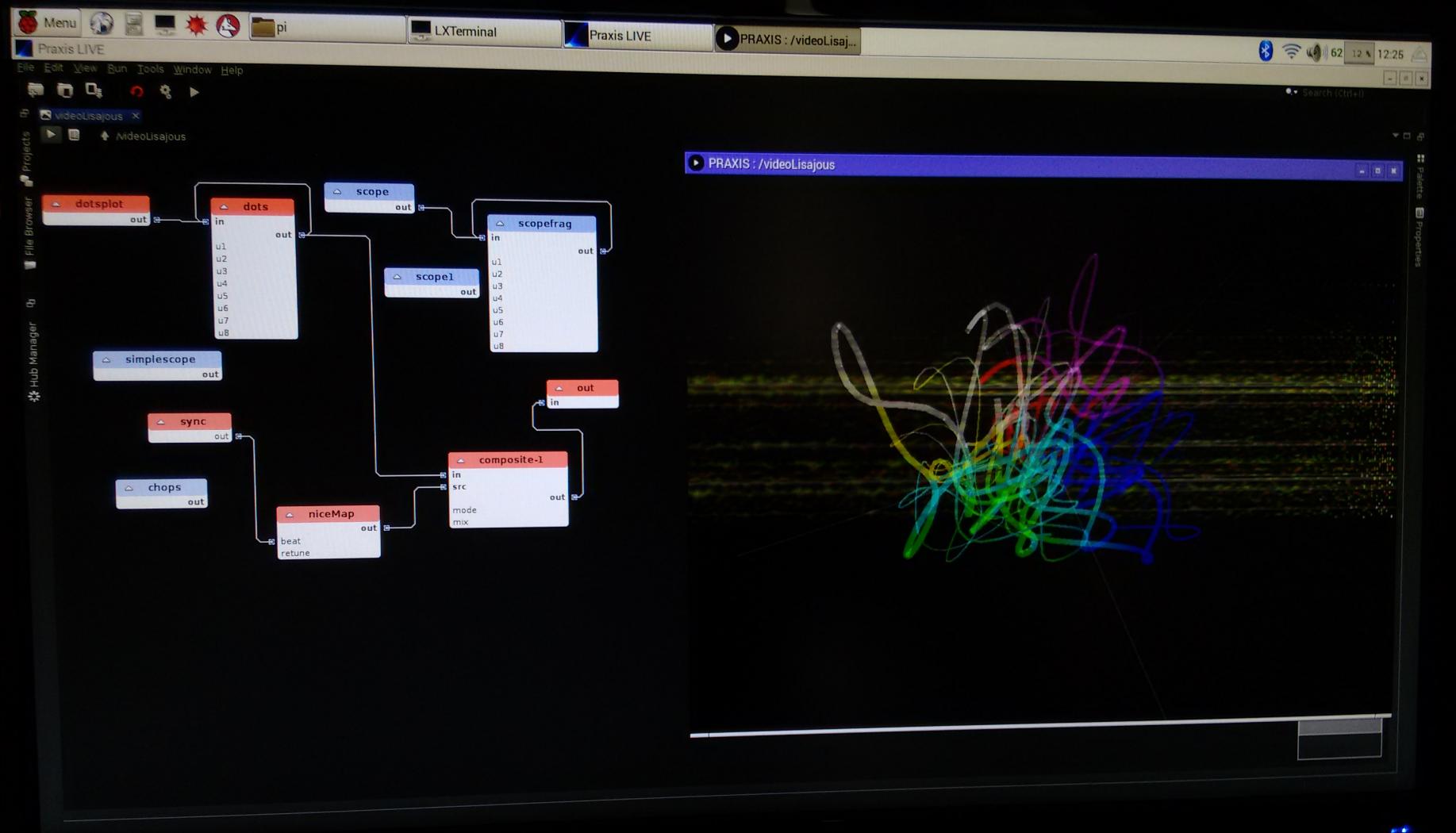












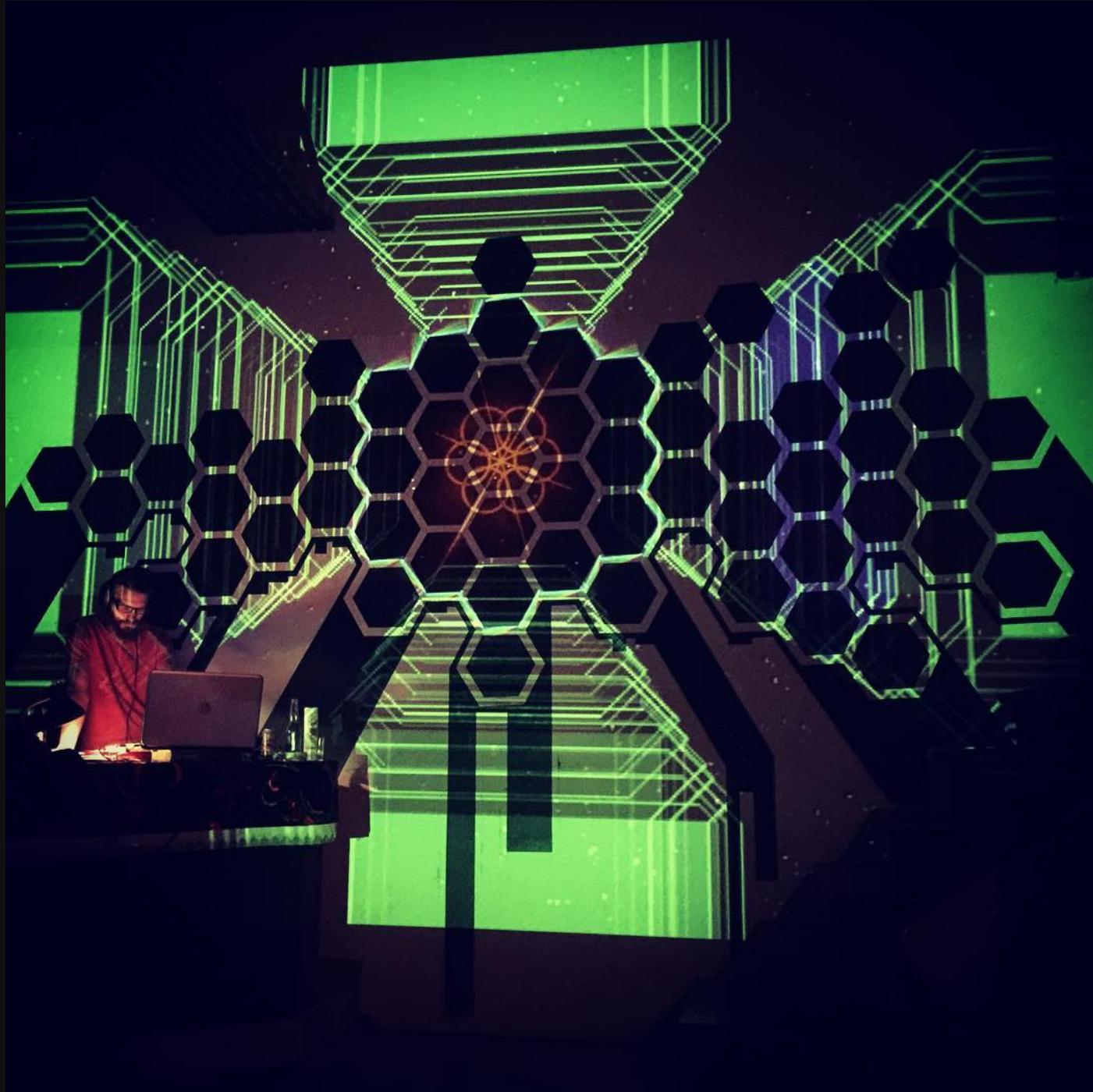
# Praxis LIVE for Processing

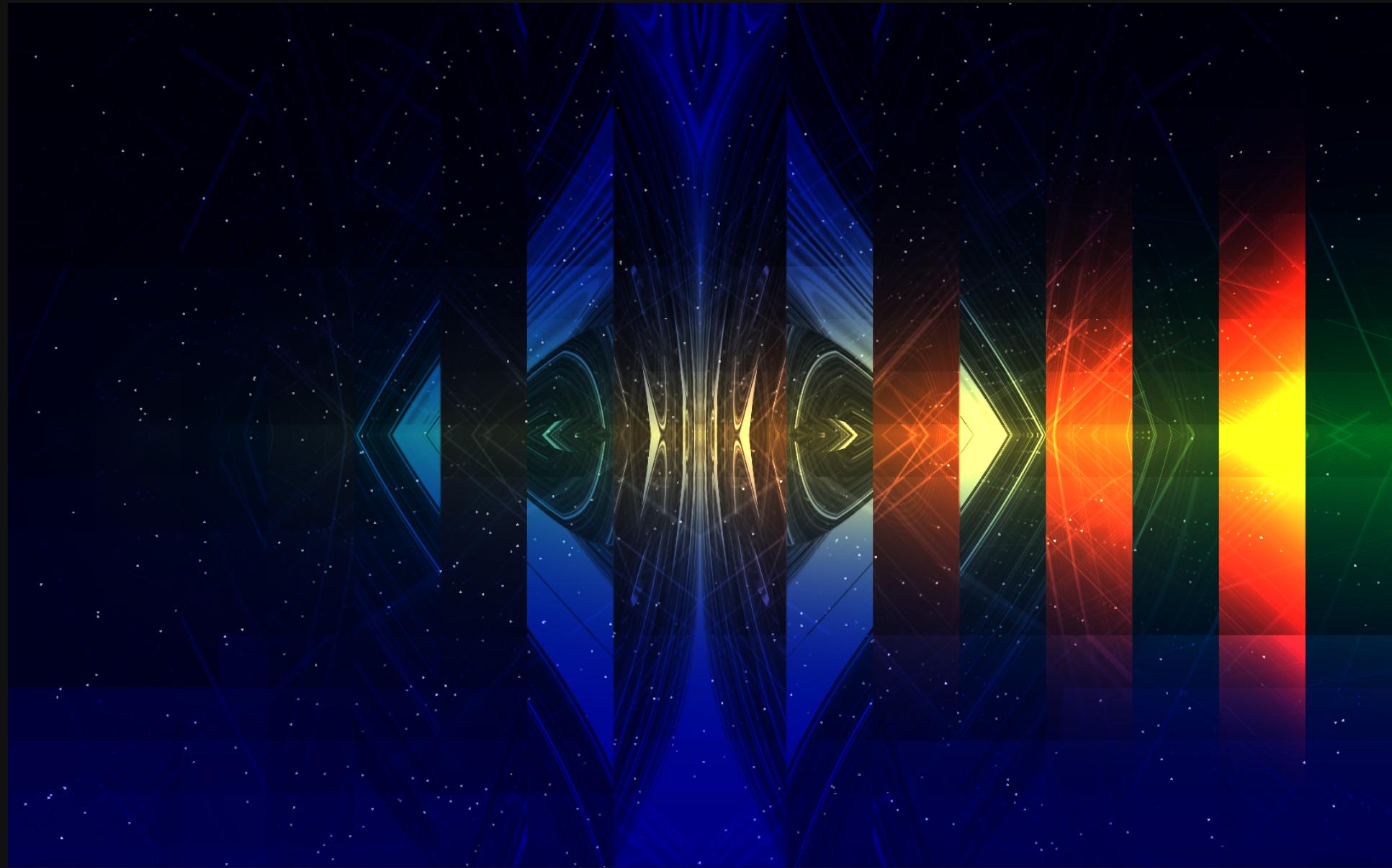
- Real-time live programming
- Multiple "sketches" and pre-built components
- GStreamer 1.x video playback and capture
- Blending "fixed"
- Resource management and optimization
- Threading / distribution done right
- Professional IDE base
- Added extras (audio, GUI, OSC, MIDI, TinkerForge ...)
- Fully libre stack!

What might you build?

maxD - @supermedia\_art



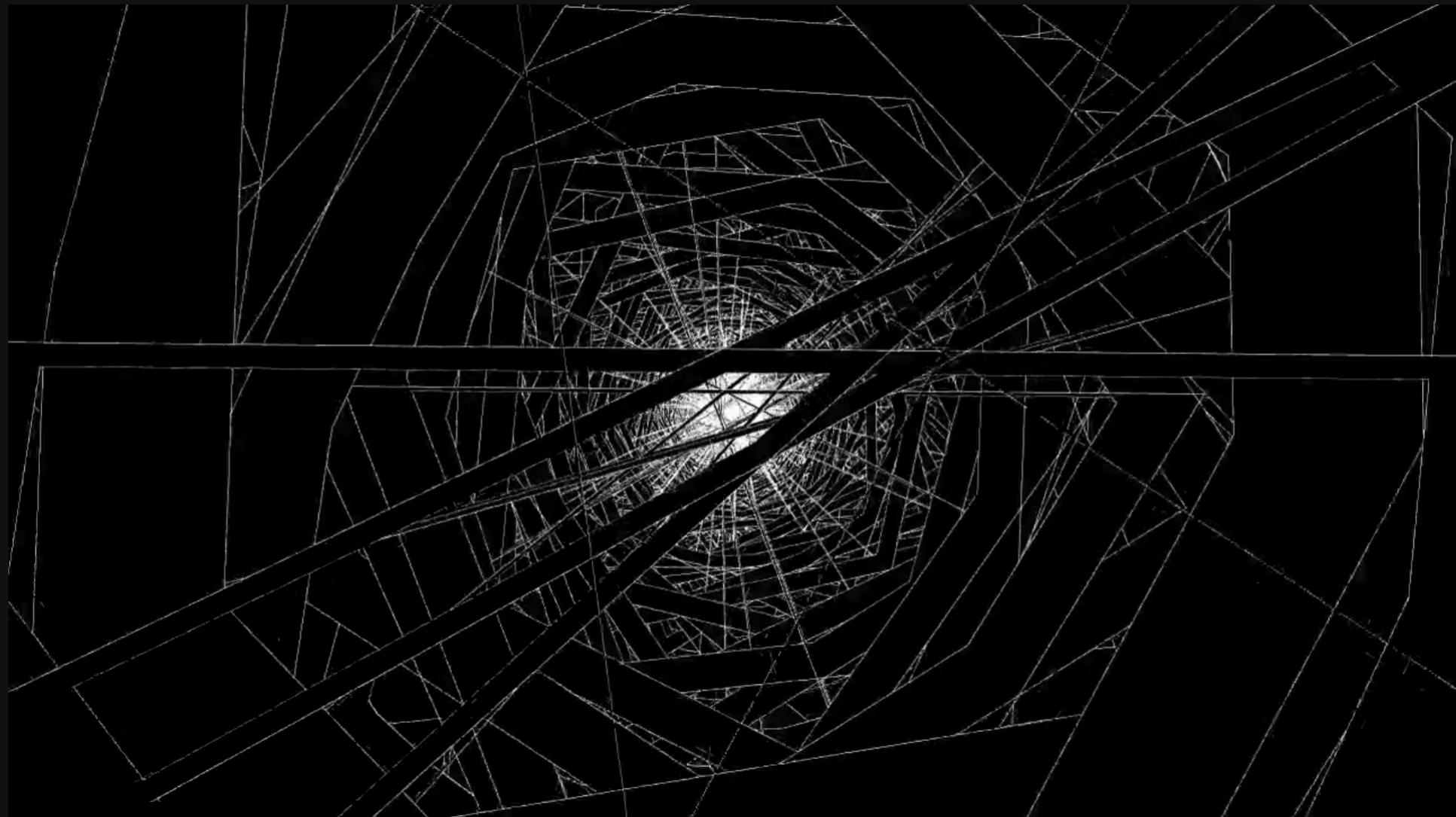




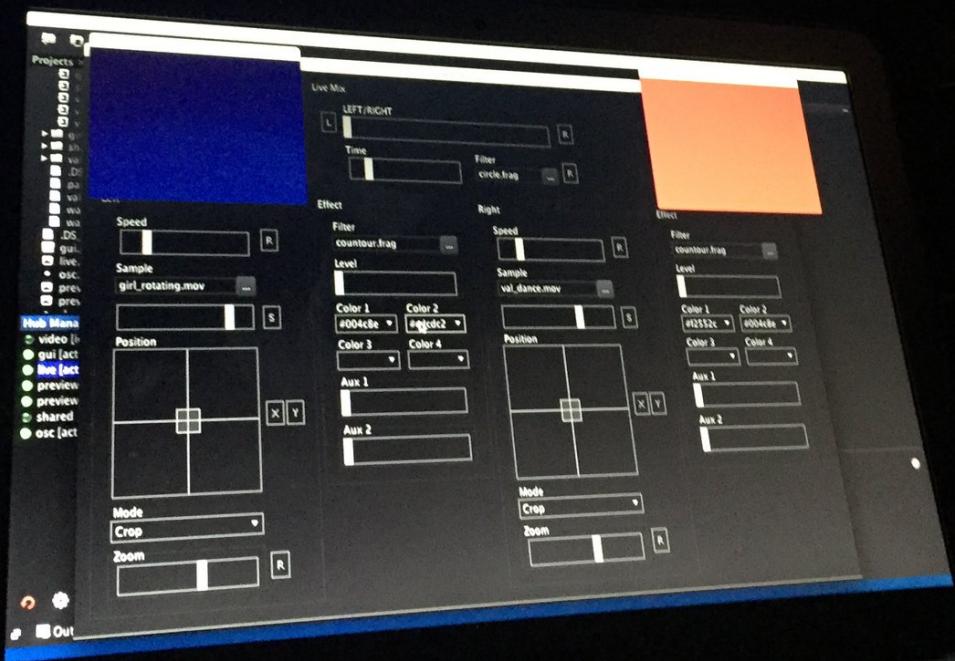
Mattias Wright / Punkomat







@y8



# Nice things people have said

*Praxis LIVE is an amazing hybrid of programming and node-based. It's a great way to build friendly reusable code.*

Howard Wong

*Overall, speed of development with Praxis is stunning. Real-time coding FTW! @y8*

*I'm a big fan of Praxis LIVE ... very fun thing to noodle with.*

Dan Hett

*Monumental MaxD*

# Workshop

Tomorrow, 10:30-12:30

# Thank you!

[www.praxislive.org](http://www.praxislive.org)

<https://github.com/praxis-live/>

<https://twitter.com/PraxisLIVE>

<https://gitter.im/praxis-live>