Intro to computer music with Pd

Charles Martin
2014

why computer music?

timbre   control   not-control   interaction

no humans

automation
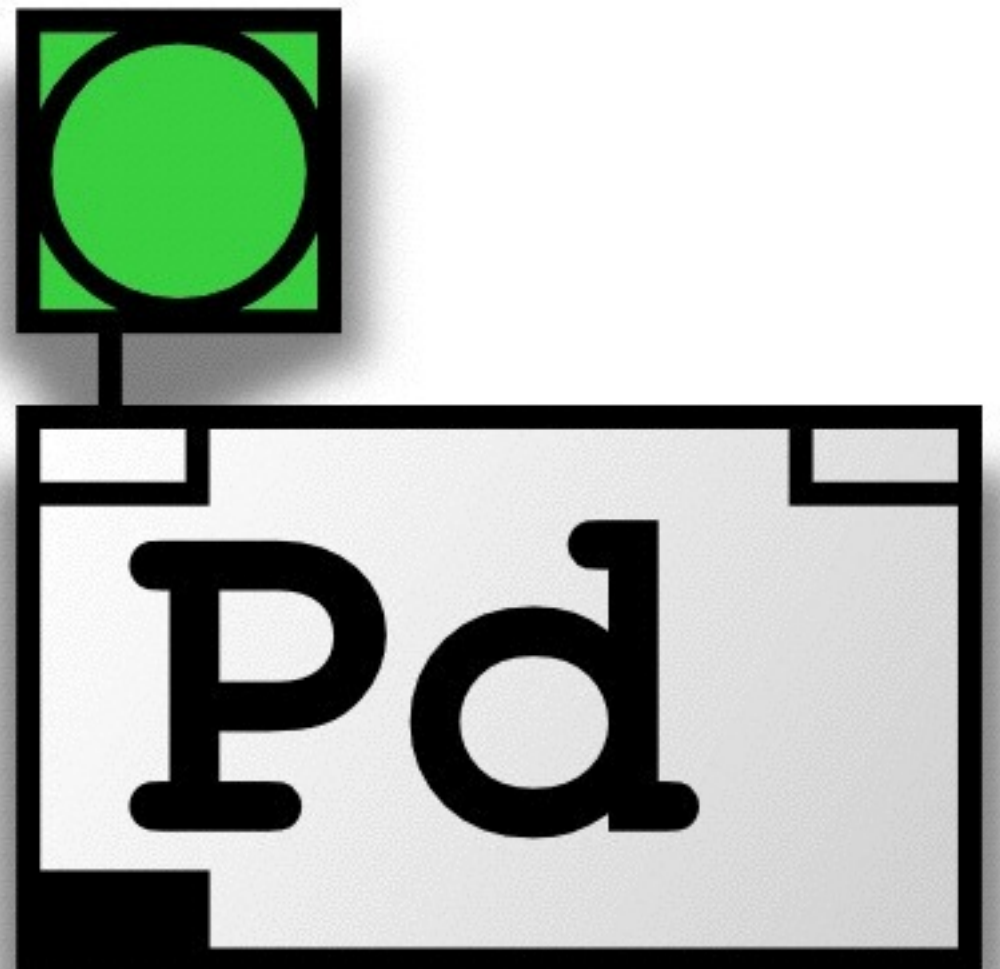
connect to:

synthesis

randomness

robots

internet visuals

audience

lights

recordings

sensors

data

instruments
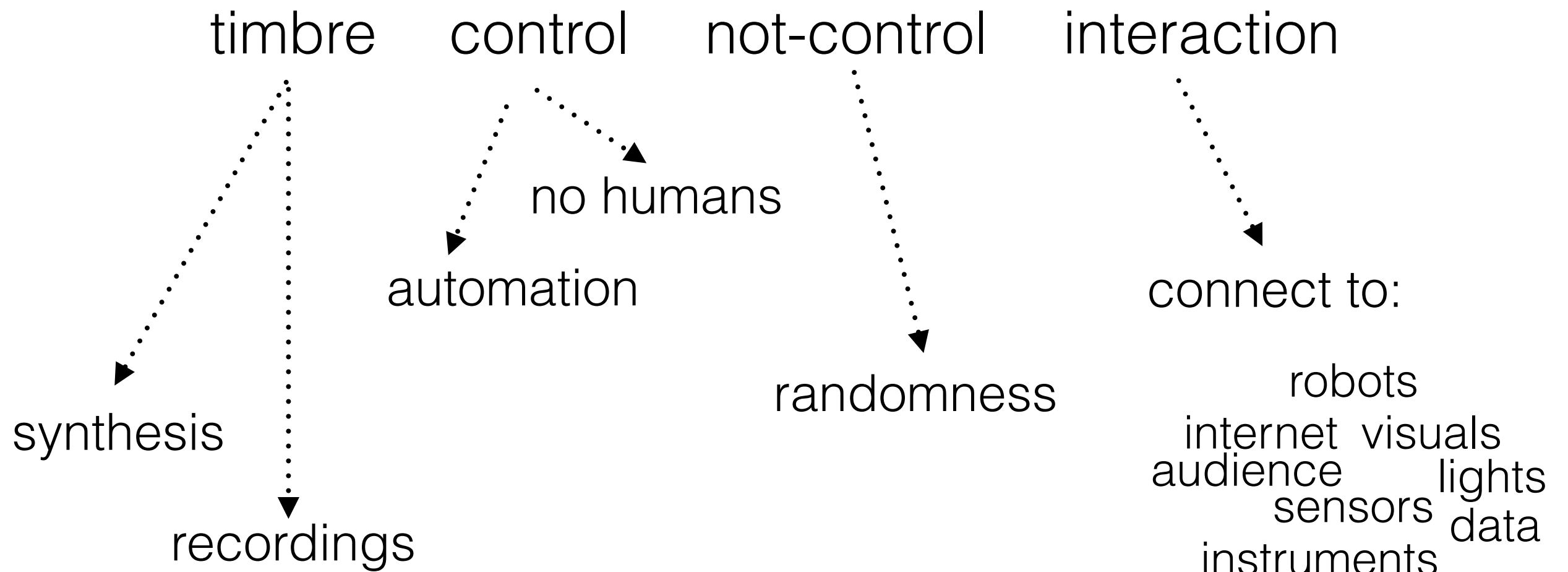
why Pure Data (Pd)?

- free

- visual

- old

- works on everything (mac, windows, linux, iPhones, iPads, iPods?)

reality: more than one tool

- lots of computer musicians use a mix of tools

  - Ableton - for messing with sound files and loops

  - Logic/Mainstage or other for synths

  - Pd for gluing things together

- communication through OSC or MIDI

Music for Vibraphone and Computer

Facetracker

The actor's movements are influenced by vibrating m

OSCulator → Pd

Nordlig Vinter
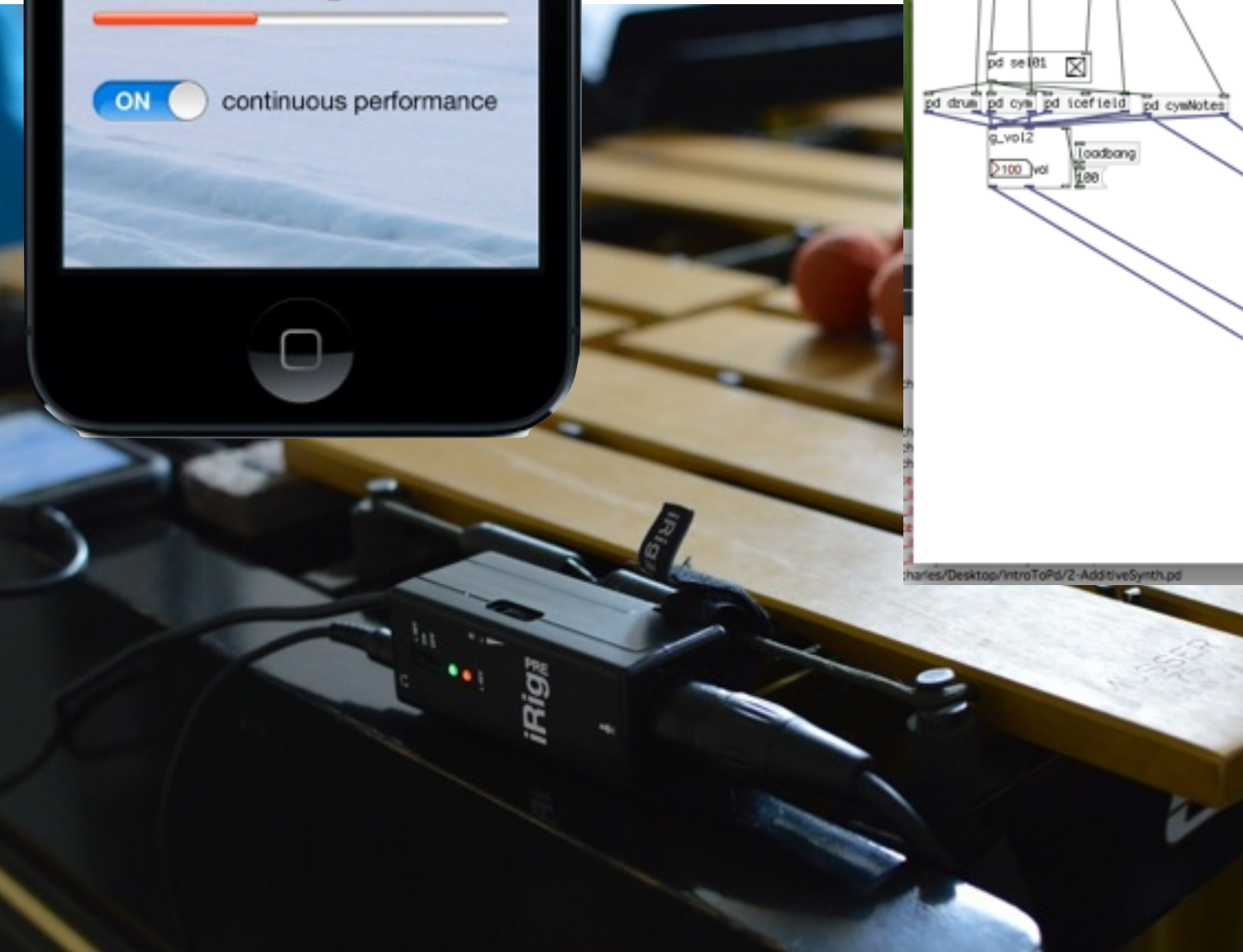
Examples: Live Patching

- Synthesis:
    + "hello world" sine tone
    + Basic Additive sine Tones
    + FM Synthesis
- Recordings:
    + Playing back a sound file
    + messing with a sound file
- Control
    + ADSR Envelope Generator
    + Controlling multiple instruments
- Not-Control
    + Random Values
    + Markov Chains
- Interaction
    + Fiddle and Bonk
    + Sensors
    + New Interface
    + MIDI / OSC

why not Pd?

- visual programming sucks (write once, read never)

  - where does the program start? hard to tell.

  - make 10 synths? Sure - copy and paste.

  - make 100 synths? Uh….

- can do in SuperCollider for example:

```
100.do({{SinOsc.ar(Rand.new(35.0,2000.0),0,[0.1,0.1],0)}.play;})
```

links:

- Getting started with percussion and computer music (2011): http://charlesmartin.com.au/blog/2011/3/27/getting-started-in-computer-music-and-percussion.html

- How to learn Pd (2014): http://charlesmartin.com.au/blog/2014/3/11/how-to-learn-pd

- https://github.com/cpmpercussion/ComputerMusicIntro