Introduction to Creative Coding

Week 8 - Connecting applications using Open Sound Control

Thomas Deacon, 2019

Topics

Week 8

- Recap week 7
- Open Sound Control protocol
- Sending messages
- Receiving messages
- Controlling apps
- Setting up networks

Recap wk 7

Serial Communications

- Hardware >>> Software
- Software >>> Hardware
- Read ASCII commands
- Communication protocol because each side of the process can understand how messages are sent, and therefore, how to be received. Same today for OSC!

What is Open Sound Control (OSC)?

An open-source protocol for communication among computers, sound synthesizers, and other multimedia devices that is optimized for modern networking technology.

Features:

- Open-ended, dynamic, URL-style symbolic naming scheme
- Symbolic and high-resolution numeric argument data
- High resolution time tags
- "Bundles" of messages whose effects must occur simultaneously

Why learn OSC?

Connecting different computers, connecting processing apps to control music software, sending data from VR to control other software and devices, interconnecting devices such as raspberry Pi to create networks of interaction.

How does it work?

OSC messages are sent over a network so you must define where you want to send your message. You must define the following things:

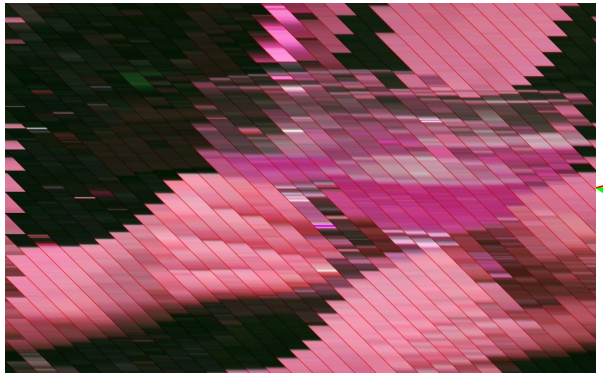
- IP Address – The IP address of the device where you send your message. If you want to send data from one software to another on your computer, you can use the localhost IP address 127.0.0.1
- Port – The port number where you are sending your message. This could basically any number, but a lot of ports are reserved for other purposes. Use a 4- or 5-digit number to stay out of the range of most common ports.
    - Example: 7777
- Address Pattern – This usually looks like an URL. It is used to differentiate between different messages you are sending on one port.
    - Example: /sensor/flex
- The actual message is the data that you are sending. You have to also define what the data type is (string, int, float etc.) The Processing library we use does the data type definition for us. Examples:
    - /sensor/accelerometer/x 0.765

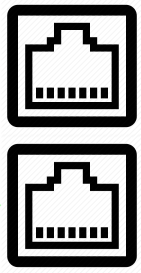
How does it work? - Local Machine

Send on 8000

Sending from Processing to Ableton

Listen on 8000





Sending from Ableton to Processing

Send on 9000

Listen on 9000

How does it work? - Over Network

Sending from computer to ipad app

Router

Sending from Ipad to computer

Computer App
IP: 192.168.1.72

IPad App
IP: 192.168.1.38

Still do ports send and listen as before

Resources for each week available at:

https://github.com/VizRCA/intro-to-creative-coding