Composing music in 3D using SuperCollider

About me

- Name: Mads Kjeldgaard
- Occupation: Composer and developer
- Work: The Norwegian Center for Technology and Art (Notam)
- Based in: Copenhagen, Denmark

Plan for today

- Part 1: Introduction
- Part 2: Patterns
- Part 3: SynthDefs
- Part 4: Ambisonics

Notam

- Development for art projects (hardware, software, tech and artistic guidance)
- Communities / meetups (SC + spatial audio meetups among others), see website notam.no
- Studios / 3D sound / VR / Visuals / Electronics
- Courses

Contact info

- mail: mail@madskjeldgaard.dk
- web: madskjeldgaard.dk
- github: github.com/madskjeldgaard
- work: notam.no

Follow me on instagram

@madskjeld

Follow me on mastodon

@madskjeldgaard@sonomu.club

What is SuperCollider?

SuperCollider is a platform for audio synthesis and algorithmic composition, used by musicians, artists, and researchers working with sound

It is free and open source software available for Windows, macOS, and Linux.

Why SuperCollider?

- Open source and free
- 20+ years of development
- Efficient, robust and stable
- Incredibly flexible
- Cross platform
- Unique design concepts and features
- Text based -> fast
- Big community

Design

Short history of SuperCollider

SC was designed by James McCartney as closed source proprietary software

Version 1 came out in 1996 based on a Max object called Pyrite. Cost 250$+shipping and could only run on PowerMacs.

Became free open source software in 2002 and is now cross platform.

Overview

When you download SuperCollider, you get an application that consists of 3 separate programs:

1. The IDE, a smart text editor
2. The SuperCollider language / client (**sclang**)
3. The SuperCollider sound server (**scsynth**)

Architecture

The client (language and interpreter) communicates with the server (signal processing)

This happens over the network using Open Sound Control

Multiple servers

This modular / networked design means one client can control many servers

Consequences of this modular design

Each of SuperCollider's components are replacable

IDE <---> SCIDE, (N)Vim, Atom or VSCode

language <---> Python, CLisp, TidalCycles, Javascript

server <---> Max/MSP, Ableton Live, Reaper

Extending SuperCollider

The functionality of SuperCollider can be extended using external packages

These are called Quarks and can be installed using SuperCollider itself

```
// Install packages via GUI (does not contain all packages)
Quarks.gui;
```

Changing SuperCollider

Since SuperCollider is an open source system, any part of it can be modified or extended by the user using classes.

See: Writing Classes for more information.

SC Plugins

SC3 Plugins is a collection of
user contributed code, mostly for making sound

Normally placed in your user extensions folder:

```
Platform.userExtensionDir.openOS()
```

IDE

What is the IDE?

The IDE is the text editor that comes with SuperCollider.
It has some really smart features that are really helpful when writing code.

Important keyboard shortcuts

- Open help file for thing under cursor: **Ctrl/cmd + d**
- Evaluate code block: **Ctrl/cmd + enter**
- Stop all running code: **Ctrl/cmd + .**
- Start audio server: **Ctrl/cmd + b**
- Recompile: **Ctrl/cmd + shift + l**
- Clear post window: **Ctrl/cmd + shift + p**

The IDE as a calculator

SuperCollider is an interpreted language

This means we can "live code" it without waiting for it to compile

A good example of this is using it as a calculator.
Try typing `2+2` and evaluate it:

```
2+2
-> 4
```

Evaluating code

- Lines of code
- Code blocks

Autocompletion

Start typing `Sin` and see a menu pop up with suggestions (and help files).

Use `up/down` arrow keys to navigate and hit `enter` to choose one

The status line

Shows information about system usage

Right click to see server options + volume slider

Help browser

There is an interactive help browser available.

You can select and evaluate all code in the browser and see / hear the results immediately.

Help browser online

There's an online version of the help system available at doc.sccode.org/ which is really helpful for sharing links to documentation.

Post window

This is where you see the resulting return messages of the code you have evaluated

Error messages posted here.

Further learning resources

Videos

Tutorials by Eli Fieldsteel covering a range of subjects:
SuperCollider Tutorials

E-book:

- A gentle introduction to SuperCollider

Paper:

- Introduction to SuperCollider, Andrea Valle
- The SuperCollider Book

Community

- scsynth.org
- sccode.org
- Slack
- Lurk
- Mailing list
- Telegram
- Telegram ES
- Facebook

Awesome SuperCollider

A curated list of SuperCollider stuff

Find inspiration and (a lot more) more resources here:

Awesome Supercollider

Learning to code: Advice

- Practice 5 minutes every day
- Set yourself goals: Make (small) projects
- Use the community
- Contribute to SuperCollider * improve documentation, help out on the forums or make bug reports

A warning

And finally, before we start making sound:

Be really careful! Keep volumes at a reasonably low level to avoid damaging your ears.

Sound demo 1

Let's listen to some music ...