cloud-5: A System for Composing and Publishing Cloud Music

Michael Gogins
http://michaelgogins.tumblr.com

Irreducible Productions
New York

18 September 2024

Introduction

- cloud-5 is a new system for composing, performing, and publishing computer music, using only Web standards.
- In this talk, I focus on the *motivations* for creating cloud-5.
- Implementation details may be found in my paper, and in the cloud-5 GitHub repository.
- I show how cloud-5 accommodates composing, live performance, and online publishing in one environment.

Motivations

- *Primarily* I want to make the kind of music that I like to hear, but nobody else seems to be making, using algorithmic composition and synthesis.
  - As I wander down this thorny path, I find myself wanting to perform some pieces live.
  - I also find myself wanting to make some pieces that play indefinitely, that are "always on."
  - And to generate visuals *from* my pieces, or to transform visuals *into* pieces.
- *Secondarily*, I need to drastically cut time spent maintaining software:
  - By using Web standards to run identically on all platforms...
  - ...which simplifies my software infrastructure.
  - ...and simplifies online publication.

Fundamental Dilemma

- My *tertiary* motivation is ongoing changes in how music is disseminated.
  - If musicians use social media to publish music, it will be captured by social media (e.g.YouTube and SoundCloud) in order to gather personal data from them and their audience, to sell for targeted advertising.
  - This is annoying, and social media have cut mechanical royalties to practically nothing.
  - Social media also promote stylistic conformity and ghettoization.

- Yet if musicians want to avoid capture by advertisers, difficulties arise:
  - On some social media, musicians can avoid ads, but only by paying for subscriptions.
  - There are only a few social media that are free of both ads and subscriptions, but they are not popular.
  - Musicians can avoid social media altogether, but only by setting up their own Web sites.
- Fortunately, setting up personal Web sites has gotten easier and cheaper over time.
- And if musicians do create their own Web sites, that opens up wonderful new possibilities.

Online Music

- Music on a Web site doesn't need to be streamed from files – *it can be Web pages actually generating music in real time*.
- Writing pieces as Web pages makes it possible to use all the resources of contemporary Web browsers.
- Within its security sandbox, a browser is basically an operating system plus a fast general-purpose programming language plus a game engine plus a high-resolution media player.
- Therefore, Web standards do not impose any particular limitation on audio quality, or other technical aspects (except for local file access), as compared with native applications.

Cloud Music

- I call music that is played by Web pages on the World Wide Web *cloud music* because it exists only in the cloud, the omnipresent computing infrastructure of the World Wide Web.
- I feel it is very important to grasp that cloud music is essentially *a new medium for music* and as such, *it offers new possibilities of musical expression*.
  - A piece can simply be an algorithmically generated piece of fixed duration, "tape music."
  - A piece can generate visuals.
  - Visuals can generate music.
  - A piece can interact with its listener.
  - Listeners can use a piece to create new pieces.
  - A piece can play indefinitely (or not).

New possibilities

- cloud-5 is certainly not the first system for publishing music as Web pages on the World Wide Web:
  - Gibber, for live coding.
  - Strudel, for live coding.
  - Web-IDE, for online Csound pieces.
  - ...and many others.
- These systems support multiple users, and provide a playground for shared learning and experimentation.
- By contrast, cloud-5 is designed to host *permanent* pieces that use algorithmic composition and synthesis *at a high level of power...*
- *...yet with low overhead for development and maintenance.*

Design

- The cloud-5 system is a combination of:
  - The sound processing language Csound, compiled from C to WebAssembly.
  - The live coding system Strudel, written in JavaScript.
  - The algorithmic composition system CsoundAC, compiled from C++ to WebAssembly.
  - GLSL shaders.
- Each component comes with its own JavaScript API, which cloud-5 encapsulates in custom HTML elements.
- All libraries, assets, and all other pre-requisites are *static resources* on the server filesystem.
- There is no build system for pieces.

Composing

- A cloud-5 piece is a Web page written with a text editor.
- A high-level menu system provides the shell for a piece; menu items show/hide overlays.
- The composer writes code (JavaScript, Csound, Strudel, GLSL...) to fill out, connect, and control custom elements.
- A naming convention makes it clear what composers need to add to the elements in order to make actual pieces.
- The computer music playpen extension for Visual Studio Code enables editing any code in a piece, and immediately playing that revision from a local Web server.
- The versionless nature of Web standards means pieces should remain playable into the indefinite future.

Example

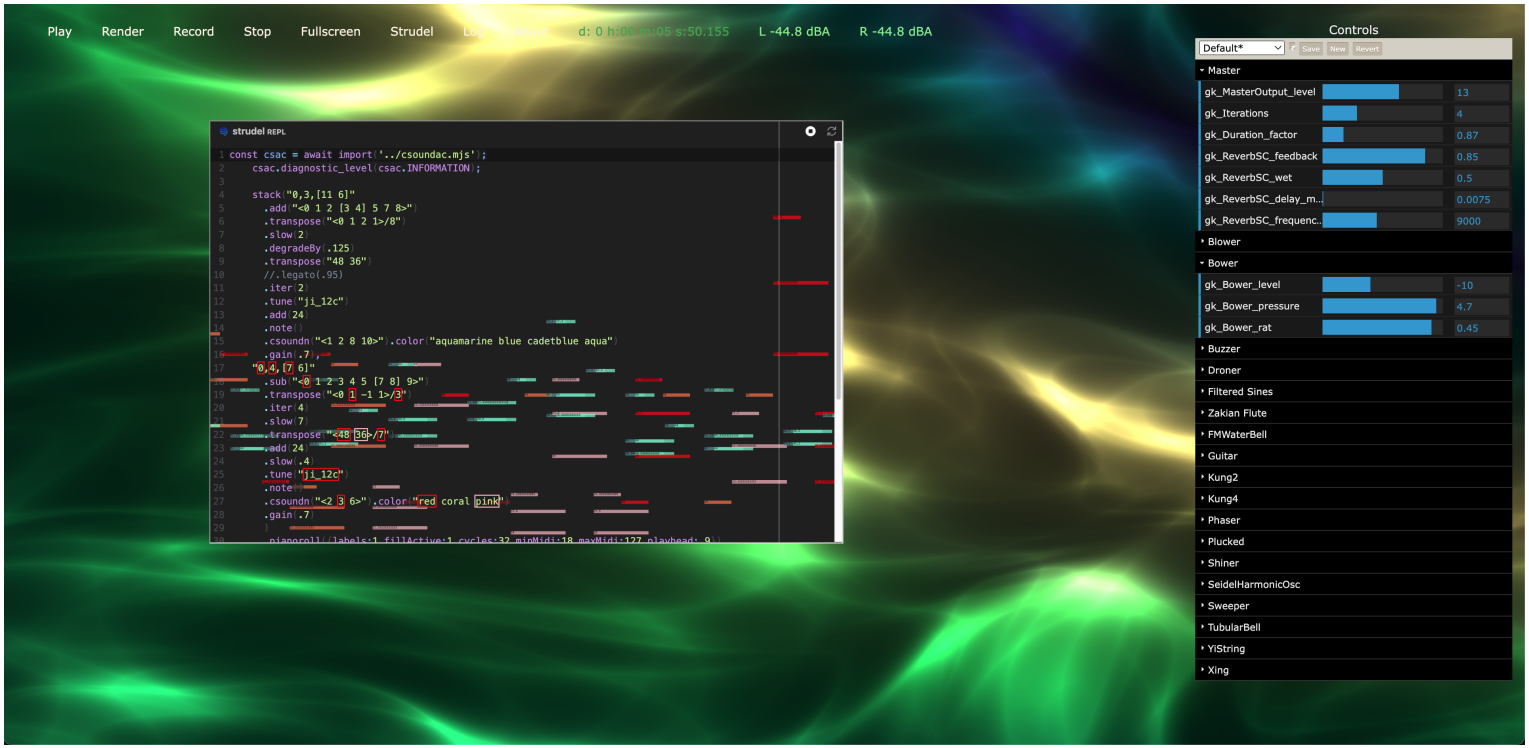

Figure 1: cloud-5 Piece with Strudel and Audio Visualization

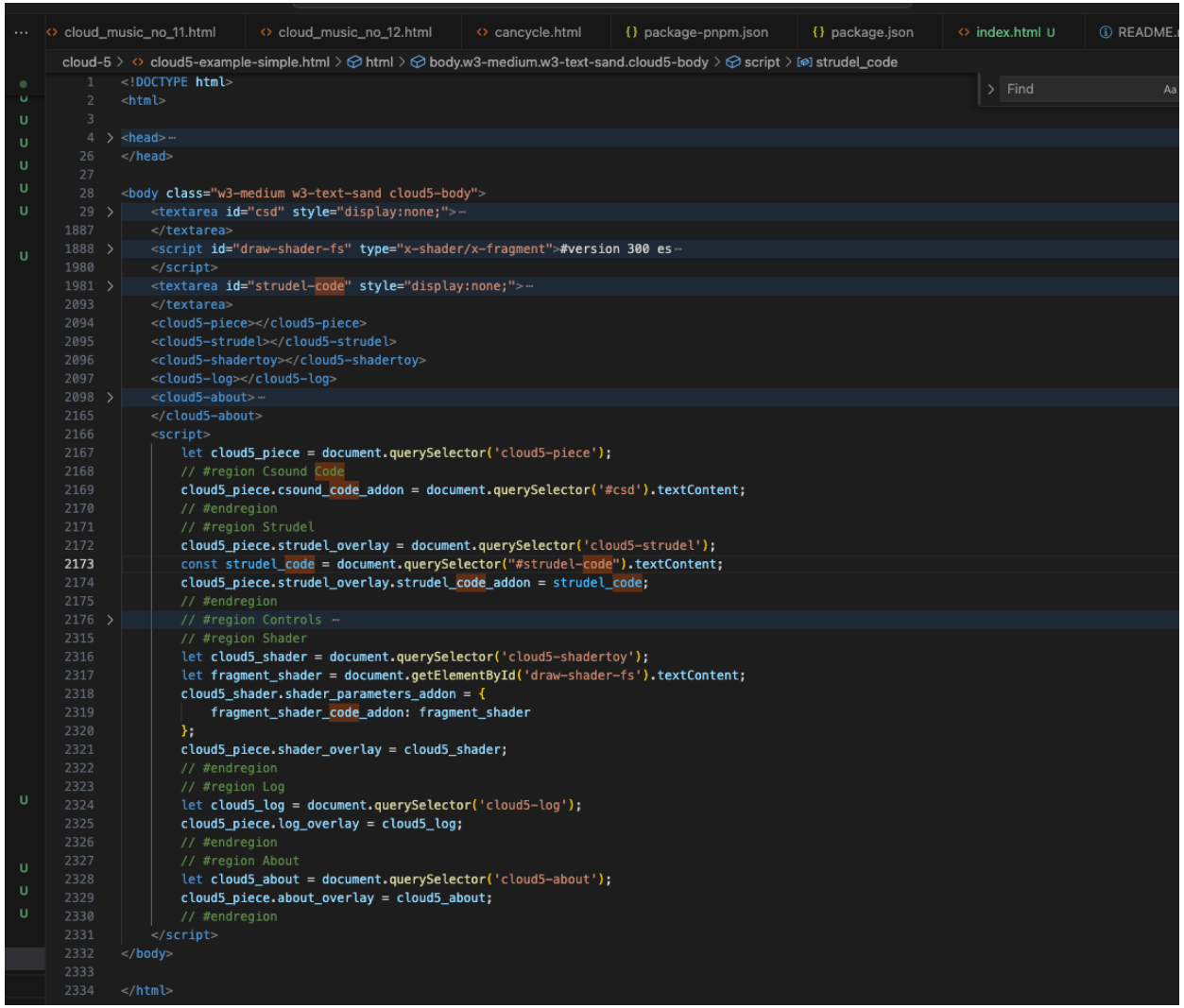

Figure 2: Outline of code

Publishing

- Copy the cloud-5 directory to any Web server. It contains all the libraries and assets that you need.
- The cloud-5 directory can be the server's root directory, or it can be a subdirectory.
- The Web server can be a local server for writing pieces or performing live pieces, or a server hosted on the Internet for publishing pieces online.
- Musical compositions, of course, are simply Web pages in the cloud-5 directory.
- Write the index page of your Web server to list and link to pieces to be published.
- Period.