

Universidad Nacional Autónoma de México

PROGRAMA DE MAESTRÍA Y DOCTORADO EN MÚSICA

TRES ESTUDIOS ABIERTOS PRÁCTICAS PERFORMÁTICAS, AUDIOVISUALES Y EXPERIMENTALES CON LENGUAJES DE PROGRAMACIÓN COMO INSTANCIAS DE CONOCIMIENTO

Emilio Ocelotl Reyes

Tutor principal: Hugo Solís García

Comité Tutor: Iracema e Andrade, Fernando Monreal

29 de agosto de 2022

ÍNDICE GENERAL

1	Introducción	7
1.1	Premisas de investigación	7
1.1.1	Premisa principal	7
1.1.2	Premisas secundarias	8
1.2	Metodología	8
1.2.1	Perspectiva de investigación	8
1.2.2	Recurso(s) de investigación	9
1.2.3	Herramientas	9
1.3	Objetivos	9
1.4	Justificación	10
2	Antecedentes	11
2.1	Trilogía de Investigación	11
2.1.1	Objeto, Paisaje y Efecto	11
2.1.2	Cuidado con la Brecha	12
2.1.3	Tres Estudios Abiertos	12
2.2	Música por computadora y algorítmica	12
2.2.1	Plataformas	12
2.2.2	Jacktrip y la música conectada	13
2.2.3	Fluxus y openGL	13
2.3	Live Coding	14
2.3.1	Primeras expresiones	14
2.3.2	Nodos y circuitos	15
2.3.3	Exploración visual	15
2.4	Proyectos Colindantes	16
2.4.1	Nivel Bajo	16
2.4.2	Nivel Medio	16
2.4.3	Nivel Alto y Ecosistemas	17
2.5	Piezas y obras lejanas	18
2.5.1	Notas de Ausencia	18
2.5.2	Hydra, visuales y el navegador	19
2.5.3	The Stage is (A)Live	19
2.6	Piezas y obras cercanas	20
2.6.1	Pumpumyatchkan	20
2.6.2	Concierto de clausura	20

2.7	PiranhaLab	20
2.7.1	Escritura de y con software	21
2.7.2	Ciclo de Talleres	21
2.7.3	EDGES 2020	21
	Referencias	22

ÍNDICE DE FIGURAS

Figura 1	Openframeworks 1	15
Figura 2	Notas de Ausencia - Marianne Teixido	18
Figura 3	Concierto PUMPUM	20
Figura 4	Concierto coloquio	21
Figura 5	Conversatorio CCD	22

1

INTRODUCCIÓN

En esta sección se hablaría de la perspectiva de investigación. Qué entendemos por escritura con y de software y la intención de usar los conceptos como metáfora.

También se pueden resolver algunas distinciones/contradicciones iniciales necesarias como software para hacer música o para hacer arte y software que es arte.

Punto de partida: estudios del software y el giro de los nuevos medios ¿Cómo se plantea el estado de la tecnología musical frente a estas reconfiguraciones? En particular ¿Qué papel juegan los lenguajes de programación en esta articulación?

Aquí podría plantearse y responderse si una tesis puede ser una pieza convencional de escritura académica o un pedazo de software al mismo tiempo. La cuestión de la acción se puede enunciar desde aquí y podría desarrollarse más abajo. Fuerte influencia de Antithesis de Geoff Cox y en general de su pensamiento.

En este apartado puedo hacer explícita la metodología de investigación. Distinción piezas-marcos de realización tecnológica. El código como puente.

1.1 PREMISAS DE INVESTIGACIÓN

1.1.1 Premisa principal

- Es posible construir un conjunto de módulos granulares audiovisuales para el navegador que permitan, por un lado, reflexionar desde la metáfora de los conceptos musicales, tecnológicos e investigativos y por el otro, explorar la expresividad de piezas audiovisuales alojadas en la web y producidas con lenguajes de programación.¹

¹ Al momento de escritura los prototipos de estos módulos se pueden encontrar en <https://github.com/EmilioOcelotl/grnlcn>

1.1.2 Premisas secundarias

- Los límites de mi lenguaje (de programación-artístico-audiovisual) son los límites de mi mundo.
- Es posible agrupar múltiples eventos audiovisuales en una estructura general temporal.
- La noción de escala de tiempo (Curtis Roads) puede ser un recurso de investigación para la resolución tecnológica, reflexiva y estética.
- La última capa de abstracción podría ser una interfaz live codeable y permitiría resolver la contradicción entre estructuras fijas y estructuras dinámicas con retroalimentación humano-máquina

1.2 METODOLOGÍA

Investigación con y sobre código. Esto quiere decir que el documento principal del proyecto será software y el código que lo posibilita.

1.2.1 Perspectiva de investigación

El proyecto toma como punto de partida la propuesta de Geoff Cox (la reflexión con y sobre código). Los planteamientos que resolveremos partirán del esquema tesis, antítesis, síntesis. Cada una de las siguientes contradicciones serían abordadas en cada capítulo. No necesariamente están escritas en orden de aparición.

- En lo tecnológico considero que esta reflexión puede contraponer dos paradigmas (programa fijo y programa variable) para encontrar una resolución (interfaz de texto).
- En lo reflexivo, se puede abordar la distinción usuario/desarrollador para dar seguimiento a procesos que marcan reconfiguración de ésta.
- Desde el punto de vista estético se puede abordar la aparente contradicción entre programas eficientes optimizados y otras búsquedas estéticas que plantean ofuscación o incluso el software que se escribe en contra del propio marco en el que está escrito.

- Desde el punto de vista investigativo se podrían abordar la escritura textual y la escritura con código y se podría plantear una resolución para la aparente brecha entre ellas.

1.2.2 Recurso(s) de investigación

Clarificar que la aproximación no es cuantitativa, es decir, si el proyecto toma en consideración algun tipo de medición, esto no será lo principal. No persigo la optimización como valor “estético”. El objetivo del proyecto es explorar las implicaciones alternativas y poéticas del código. Para esto, es necesario tener una aproximación cualitativa en términos de la interpretación intersubjetiva del código y sus instancias en piezas artísticas.

1.2.3 Herramientas

Estándar de investigación (latex, bibtex) en relación al modo personalizado de resolver problemas (terminal, emacs, instancias de código en distintos modos, repositorios en git, org-mode, comentarios que no se ejecutan o imprimen en la última versión del documento, recursos audiovisuales como piezas ejecutables en el navegador, videos, audios e imágenes)

1.3 OBJETIVOS

- Escribir módulos de síntesis granular con WebAudioAPI y AudioWorklets
- Realizar una serie de piezas que acompañen a la investigación
- Establecer un puente reflexivo entre estos dos ámbitos desde los estudios del software
- Desbordar la escritura académica a la escritura con código y performática-interactiva

1.4 JUSTIFICACIÓN

El proyecto busca explorar las posibilidades del navegador como paradigma de renderizado de audio y video y en general, como un "sistema operativo" multiplataforma. Las posibilidades de cero instalación y compatibilidad motivan técnicamente al proyecto.

Otras motivaciones son las reflexiones que resultan de vincular presencia y distancia.

Javascript es un lenguaje de programación ampliamente utilizado por la industria. Encarnar las contradicciones en la escritura de software permite a esta investigación reflexionar sobre las posibilidades de la escritura de software artístico y para el arte, sobre todo en lo que apunta hacia la diversidad tecnológica.

El proyecto busca plantear un aporte a la diversidad de marcos para realizar investigaciones con tecnología. Esta perspectiva plantearía el hacer explícitas ciertas contradicciones detectadas en la escritura investigativa y con software.

Finalmente consideramos que esta investigación puede aportar elementos para invertir las prioridades en lo que respecta a la escritura de software y su investigación: un contexto inclusivo y políticamente activado podría incursionar en reflexiones sobre el buen conocer, como un ámbito del buen vivir.

2 | ANTECEDENTES

Los antecedentes de esta investigación aluden a una trayectoria que va de la transición de la escritura de software para la realización de sistemas interactivos a la escritura de módulos de software audiovisuales. Estas experiencias toman como premisas la optimización y la ligereza de hardware (por ejemplo, con el uso de computadoras de placa reducida como Raspberry Pi o Jetson Nano) y la elaboración sistemas ligeros, accesibles y portables para la síntesis y renderización audiovisual en el navegador.

A la par de la práctica artística con tecnología, los elementos precedentes de esta investigación se han conducido hacia la problematización del software como una instancia tanto de conocimiento como de ideas que conforman un paradigma que atraviesa todo el entramado de personas, investigaciones, escritos y obras involucrado con esta actividad. Estas reflexiones han tenido eco en el campo de investigación denominado estudios del software.

Los antecedentes a continuación descritos no necesariamente tienen un orden cronológico o de importancia. En todo caso es un compendio de perspectivas, plataformas, obras y realizaciones prácticas que desde la perspectiva de quien suscribe este texto, son relevantes de enunciar para captar las incidencias en los casos de investigación.

2.1 TRILOGÍA DE INVESTIGACIÓN

Tres Estudios Abiertos forma parte de una trilogía de investigación.

2.1.1 Objeto, Paisaje y Efecto

La primera parte fue Objeto, Paisaje y Efecto (?), un proyecto de investigación que abordó las nociones de objeto sonoro (?), paisaje sonoro(?) y efecto sonoro (?) para considerar a la escucha como un recurso para la investigación sociológica en música y para la investigación social desde el sonido.

2.1.2 Cuidado con la Brecha

Un segundo punto de investigación involucró un proceso de investigación-producción artística (?). La realización de este proyecto fue un prototipo tecnológico y partió de objetivos que inicialmente estaban propuestos como secundarios pero que más tarde se revelaron como parte del núcleo en la investigación. Estos aspectos son: 1) el proceso de trabajo colaborativo y su implementación con herramientas como git, 2) la reflexión sobre la interacción entre audio e imagen en la composición musical electroacústica y 3) el uso de herramientas libres, personalizadas para la realización de prototipos audiovisuales y para el planteamiento de una observación crítica de procesos creativos donde investigador y artista son el mismo agente. La propuesta de los estudios del software fue incorporada en este momento de investigación.

2.1.3 Tres Estudios Abiertos

Referencia recursiva a esta investigación para explicar la trilogía

2.2 MÚSICA POR COMPUTADORA Y ALGORÍTMICA

2.2.1 Plataformas

A continuación, haremos un breve rastreo de ideas centrales en la escritura de software orientado a la creación musical. El objetivo de este apartado consiste en describir y detectar la presencia del concepto unidad generadora en diversos programas orientados a la generación musical, de entre los cuales está una de las librerías que la presente investigación implementa.

El punto de partida de esta indagación es MUSIC N, proyecto de Max Mathews que sería el parteaguas del paradigma de la música por computadora. Uno de los primeros casos de esta instancia es el principio de programas como Max/MSP y PureData, proyectos representativos de la programación gráfica presente en flujos de trabajo gráfico actuales como TouchDesigner.

Como una observación adicional, es importante detectar instancias de la programación gráfica y de las ideas principales de la música por computadora en plataformas con giros programáticos particulares. Tal es el caso de OpenMusic, que de manera específica está basado en Common Lisp.

Desde la perspectiva de la programación escrita, destacamos el papel que ha tenido SuperCollider en la extensión del paradigma de la música por computadora en la actualidad. Señalamos la importancia de SuperCollider como el motor bajo el cual se pueden ejecutar entornos de programación al vuelo como Tidal Cycles o FoxDot.

Estuary es un caso adicional que permite establecer un puente entre Tidal Cycles como un entorno que ejecuta SuperCollider como motor de audio y el navegador como tecnología multiplataforma sin instalaciones. Esta plataforma utiliza secuenciadores basados en la sintaxis de Tidal Cycles pero a diferencia del entorno que se puede instalar de manera local en una computadora, Estuary utiliza al navegador como motor de audio.

Trazamos estas relaciones para establecer una relación continua y presente entre los entornos anteriormente descritos y una de las librerías utilizadas en los casos de estudio de esta investigación: Tone.js. En este sentido, consideramos importante la adscripción a los principios de la música por computadora para encontrar soluciones personalizadas para el navegador.

2.2.2 Jacktrip y la música conectada

Una parte de los antecedentes de este proyecto se vinculan con la actividad del colectivo RGGTRN¹ y del LiveCodeNet Ensamble²

RGGTRN es un colectivo de música por computadora fundado por Luis N. Del Angel y Emilio Ocelotl. Posteriormente se unieron Marianne Teixido y Jessica A. Rodríguez. Como parte de un ejercicio lúdico y de reflexión, el colectivo explora la improvisación audiovisual realizada por medio de código de programación, con una relación al contexto Latinx de sus participantes.

raspis conectadas y jacktrip ¿el trabajo realizado por CCRMA y en general
La labor del colectivo Radiador

Sonobus y la resolución de problemas de streaming en tiempos de pandemia

2.2.3 Fluxus y openGL

Para el caso de la imagen, retomo la influencia que tiene en la comunidad de live coding y en mi experiencia del performance audiovisual con la computadora el papel que tuvo el desarrollo Fluxus³ de Dave Griffiths que

¹ <https://rggtrn.github.io/>. Consultado el 29 de agosto de 2022

² <https://livecodenetensamble.wordpress.com/>. Consultado el 29 de agosto de 2022

³ <https://gitlab.com/nebogeo/fluxus/>

se remonta al 2007. Una característica peculiar de este desarrollo es el uso de una sintaxis tipo LISP que recuerda a desarrollos musicales basados en este lenguaje de programación como OpenMusic.

Detrás de Fluxus también cabe destacar la importancia de sistemas de renderización de gráficos por computadora como OpenGL, que actualmente, son el punto de partida de software de alto nivel involucrado con este proyecto como OpenFrameworks y la variante para el navegador, webGL, que implementa la librería Three.js

2.3 LIVE CODING

Encuentro múltiples formas de abordar el tema. El live coding puede describirse en términos de una comunidad que realiza una práctica y que se enuncia como tal, como un sub-campo creativo que comparte elementos con otra expresiones como la música algorítmica y el arte generativo. Finalmente, el live coding podría acotarse a motivos de investigación académica e independiente y a la realización tecnológica de interfaces de texto que corren sobre motores de audio y video.

De estas múltiples formas de abordar el fenómeno y de acuerdo a los fines de esta investigación, nos preguntamos si el eje que las articula son discusiones sobre el lenguaje y la escritura.

2.3.1 Primeras expresiones

Antecedentes del live coding: la escena mod.

En el manifiesto del live coding se expresan pensamientos que hasta el momento, son vigentes.

Dentro de los antecedentes está la experiencia performática de escribir código al vuelo con fines creativos, audiovisuales y experimentales.

Las primeras expresiones reflexivas y prácticas del live coding no solamente establecen puntos de partida performáticas, sino que al mismo tiempo aportan elementos para el diseño y análisis de sistemas basados en interfaces de código:

“Nuestro argumento nos lleva a través de capas de representación, comenzando con símbolos, luego palabras, lenguaje y notación, para considerar el papel que estas representaciones pueden jugar en la creatividad humana” (? , p. 3)

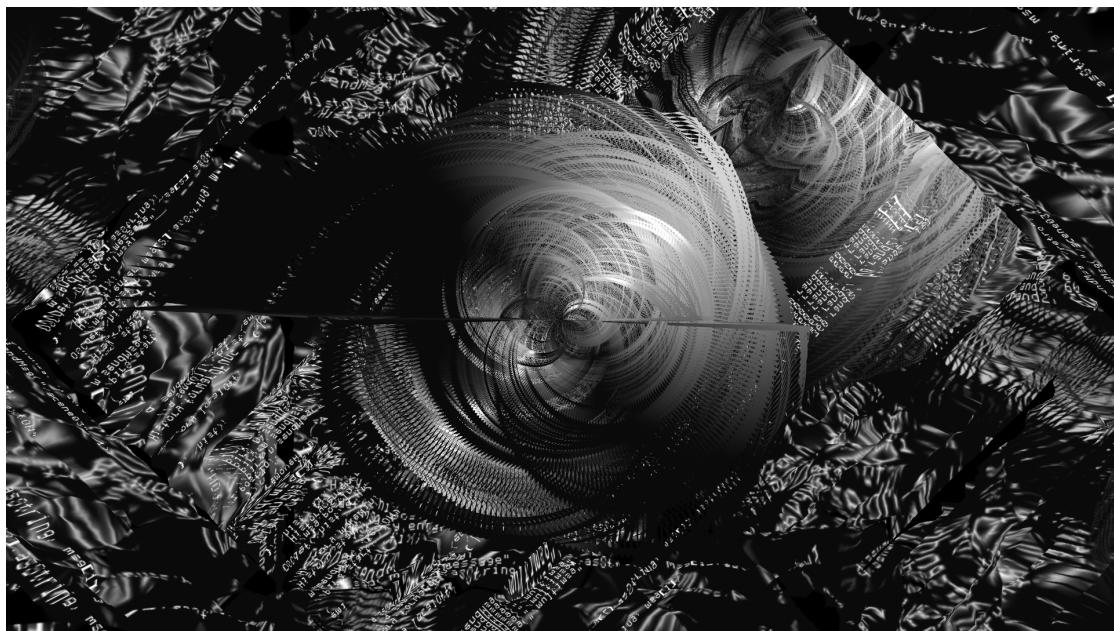


Figura 1: Captura de render realizado con OpenFrameworks

La noción de capas de representación permite analizar la complejidad del proceso artístico con código en términos de la relación entre humanos y los aspectos cognitivos, sociales y políticos que les atraviesan, y los no-humanos en lo que respecta a las posibilidades de conocimiento instanciado presentes en ellos.

Live coding desde cero y la improvisación.

2.3.2 Nodos y circuitos

La práctica de la programación al vuelo, delimitada técnica, escénica, musical y visualmente se origina en Inglaterra.

Como lo describen ? para los casos de Barcelona y Ciudad de México.⁴

2.3.3 Exploración visual

Exploración visual para la integración con el sonido.

El campo del live coding tuvo un giro importante con la llegada de hydra de Olivia Jack. La rápida expansión de esta plataforma y la sintaxis que recuerda

⁴ Un ejemplo reciente se encuentra en: <https://youtu.be/n5kwi4eRAE4>

la conexión de flujos de energía similar a los sintetizadores analógicos favorecieron la producción de piezas e interpretaciones enfocadas en la imagen y en algunos otros casos a la par del audio. La importancia de esta plataforma ha delimitado una estética basada en la transformación de píxeles, de formas y de transformaciones basadas en funciones matemáticas.

En palabras de Olivia Jack, los procesos que realiza la computadora hay un eje de profundidad, en términos del despliegue gráfico físico de la computadora de espacios bi o tridimensionales, el resultado es bidimensional.

La lógica de Hydra es modular, esto quiere decir que puede incorporarse como un componente adicional a proyectos que no necesariamente se centren en la producción visual con esta plataforma. Esta modularidad juega a favor de lenguajes de programación como Javascript.

2.4 PROYECTOS COLINDANTES

Algunos proyectos cercanos tecnológica, conceptual y estéticamente a *Tres Estudios Abiertos* son:

2.4.1 Nivel Bajo

- Ruffbox⁵
- WebAssembly/Rust Tutorial⁶
- Flocking⁷

2.4.2 Nivel Medio

- supercollider.web⁸
- Web Audio API⁹
- Tone.js¹⁰

⁵ <https://github.com/the-drunk-coder/ruffbox>

⁶ <https://www.toptal.com/webassembly/webassembly-rust-tutorial-web-audio>

⁷ <https://github.com/continuing-creativity/Flocking/>

⁸ <https://github.com/khilnani/supercollider.web/>

⁹ https://developer.mozilla.org/es/docs/Web/API/Web_Audio_API

¹⁰ <https://tonejs.github.io/>

- SuperCollider.js¹¹

2.4.3 Nivel Alto y Ecosistemas

Nivel Alto

- Hydra¹²
- Troop¹³
- flok¹⁴
- tilt¹⁵
- LiveLab¹⁶
- timeNot¹⁷
- seis8s¹⁸
- Camposónico¹⁹
- INSTRUMENT²⁰

Ecosistemas

- Estuary²¹
- Sema-engine²²

¹¹ <https://github.com/crucialfelix/supercolliderjs/>

¹² <https://github.com/ojack/hydra>

¹³ <https://github.com/Qirky/Troop>

¹⁴ <https://github.com/munshkr/flok>

¹⁵ <https://github.com/munshkr/tilt>

¹⁶ <https://github.com/ojack/LiveLab>

¹⁷ <https://github.com/luisnavarrodelangel/seis8s>

¹⁸ <https://github.com/AFrancoB/timeNot>

¹⁹ <https://github.com/diegovdc/camposonico>

²⁰ <https://github.com/punksnotdev/INSTRUMENT>

²¹ <https://github.com/dktro/estuary>

²² <https://github.com/frantico/sema-enginef>

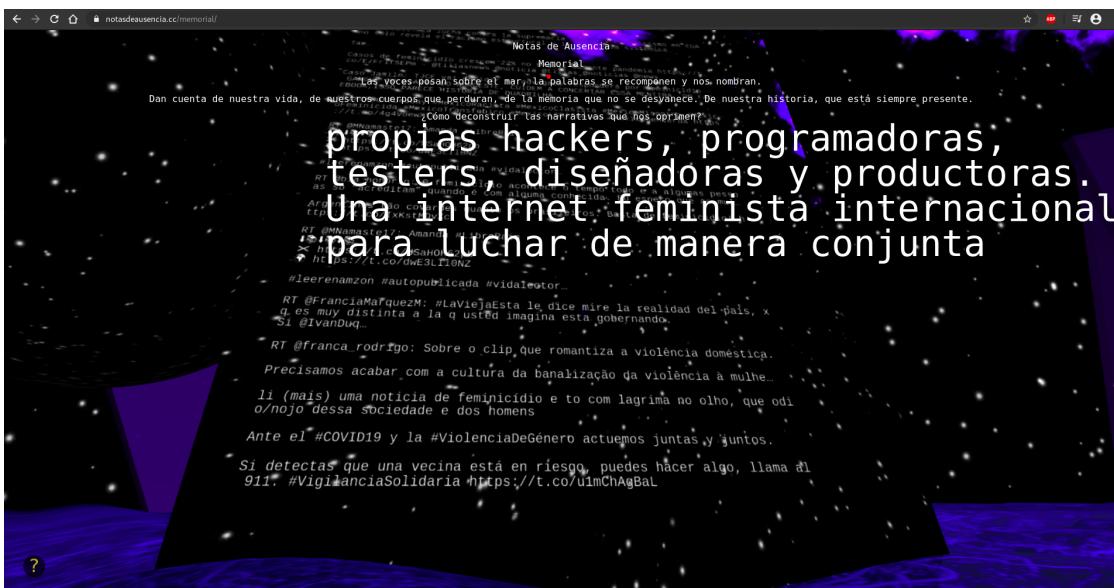


Figura 2: Captura de Notas de Ausencia de Marianne Teixido. Consultado el 29 de agosto de 2022en: <https://github.com/MarianneTeixido/notasdeausencia>

2.5 PIEZAS Y OBRAS LEJANAS

2.5.1 Notas de Ausencia

Notas de Ausencia de Marianne Teixido es un ensayo generativo en la web. Utiliza texto-dato que por medio de la computadora como agente resignificante, destruye estructuras discursivas para resemantizar la narrativa sobre las desapariciones de mujeres en México y América Latina.

El tiempo y espacio virtual conforman una partitura para la memoria y la denuncia. La narrativa, semi-autónoma, argumenta a partir de textos tomados de tweets, poemarios, libros y artículos feministas que explican desde la teoría las desapariciones forzadas, el feminicidio y la violencia de género. Estos elementos están presentes como texto, imagen y sonido en un espacio tridimensional diseñado a manera de memorial.

La narrativa de la pieza está articulada mediante la intervención de dos bots que interactúan en Twitter. El primero realiza una búsqueda de tuits a partir de un filtro de palabras clave escritas como hashtags como: #MéxicoFeminicida, #MadresEnBúsqueda, #ViolenciadeGenero, #NiUnaMenos, entre otros.²³

²³ Al momento de escritura, el bot se encuentra activo en la cuenta: <https://twitter.com/notasausencia> (Consultado el 29 de agosto de 2022)

Posteriormente, recomparte el tuit que contiene alguno de los hashtags de la base de datos previamente delimitada. El segundo bot retoma los textos de los tuits seleccionados y los remixea para generar un segundo texto automático por medio de cadenas de Markov.

Esta pieza se realizó en el contexto de la exhibición en línea Creaciones con Algoritmos: Visualización y Sonificación de Datos del Centro de Cultura Digital en abril de 2020. Su salida oficial se realizó en video sin embargo, el planeamiento original contempló la creación de un espacio virtual en la web que pudiera visualizar en tiempo real la información viva proveniente de los tweets.

Dos características definieron el diseño de la pieza: ubicación de la pieza en un espacio tridimensional y la interacción y visualización de la información proporcionada por los bots. Para la realización, el proyecto retomó módulos iniciales de Panorama y partió del uso de Three.js como un entorno de trabajo que pudiera conectar los dos momentos de la pieza antes descritos.

2.5.2 Hydra, visuales y el navegador

Uno de los referentes en términos de live coding con resultados visuales es la obra de Olivia Jack. La realización de éstas va de la mano con el software desarrollado por ella misma: Hydra. El proyecto está alojado en la web y la descripción se define como: “[U]na plataforma para live codear visuales, en la que cada ventana del browser puede ser usado como un nodo de un sintetizador de video, modular y distribuido”²⁴

Hydra es un proyecto abierto y libre, lo cual permite que una comunidad de usuarios se involucren en el uso, mantenimiento y expansión del repositorio. Algunos proyectos que utilizan directa o indirectamente hydra.

Destacamos del proyecto la sintaxis personalizada que permite controlar el resultado visual a partir de funciones y parámetros. Estas funciones pueden anidarse y concatenarse, lo cual da como resultado una serie de instrucciones aglutinadas y encadenadas. Este continuo coincide con la analogía del patcheo de los sintetizadores modulares de audio y video.

2.5.3 The Stage is (A)Live

The stage is (a)Live²⁵ - Johana Chicau y Renick Bell

²⁴ Consultado el 29 de agosto de 2022en: hydra.ojack.xyz

²⁵ <https://www.geometries.xyz/theStageIsAlive/>

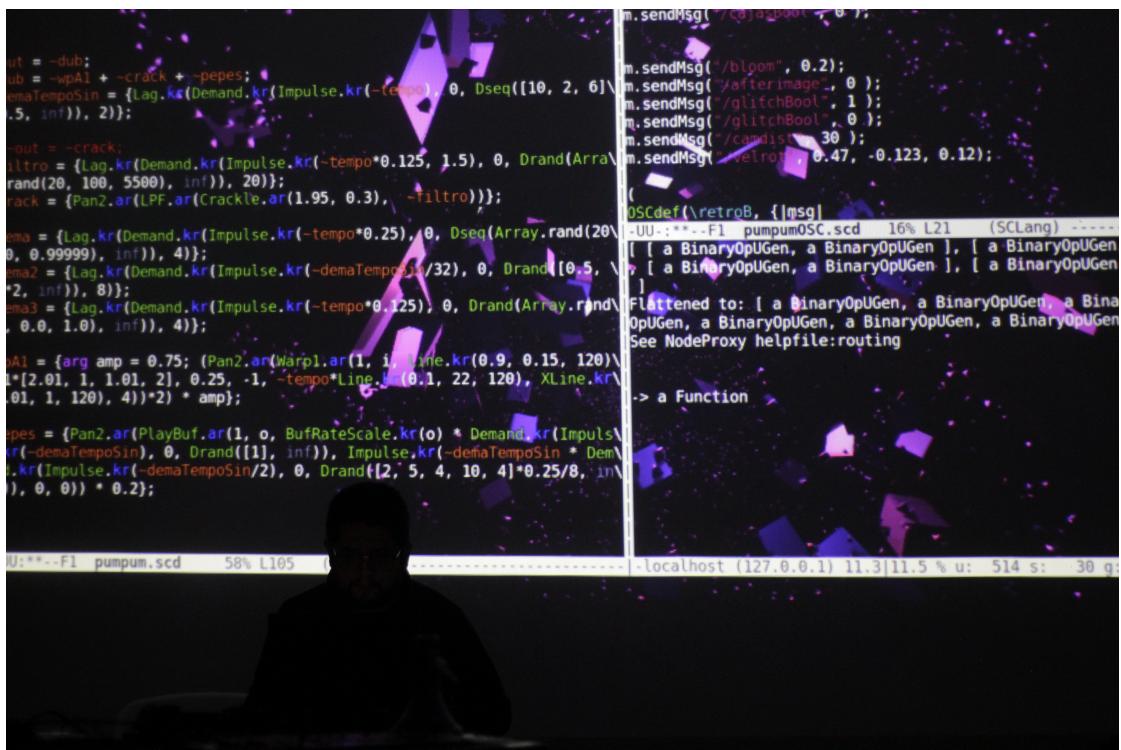


Figura 3: Presentación en el festival PUMPUM

2.6 PIEZAS Y OBRAS CERCANAS

2.6.1 Pumpumyatchkan

2.6.2 Concierto de clausura

Concierto virtual realizado en el marco del coloquio de alumnas del Programa de Posgrado en Música de la UNAM

2.7 PIRANHALAB

Otro antecedente de este proyecto es la práctica y reflexión planteada en colectivo por *PiranhaLab*²⁶.

²⁶ “PiranhaLab es un laboratorio interdisciplinario que trabaja en las tripas del software”. <https://piranhalab.github.io/> (Consultado el 29 de agosto de 2022)

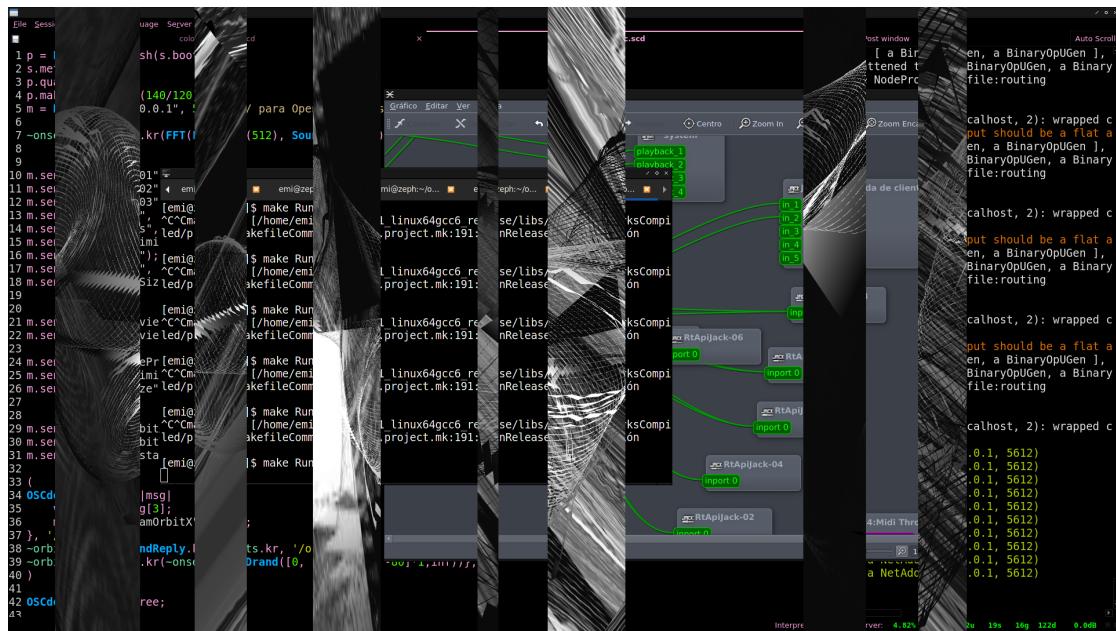


Figura 4: Captura del concierto del Coloquio de alumnas <https://youtu.be/HwBTRQKrqPs>.

2.7.1 Escritura de y con software

2.7.2 Ciclo de Talleres

El ciclo de talleres realizado en el Centro de Cultura Digital (CCD) en co-participación con el Laboratorio de Tecnologías Libres²⁷ permitió plantear dos conclusiones que se heredan a *Tres Estudios Abiertos*: La difuminación de la distinción usuario/desarrollador como una motivación para la escritura de software y la procuración de diversidad en la escritura de software en América Latina.

2.7.3 EDGES 2020

La escritura de espacios para el ciclo de conciertos EDGES 2020 realizado por el Taller de Imágenes en Movimiento del Centro Multimedia (CMM) permitió la exploración de entornos tridimensionales inmersivos en el navegador en el contexto del encierro causado por la pandemia de COVID-19. Técnica y conceptualmente la escritura de estos espacios digitales influye en el presente

²⁷ Actualmente Laboratorio de Tecnologías Compartidas

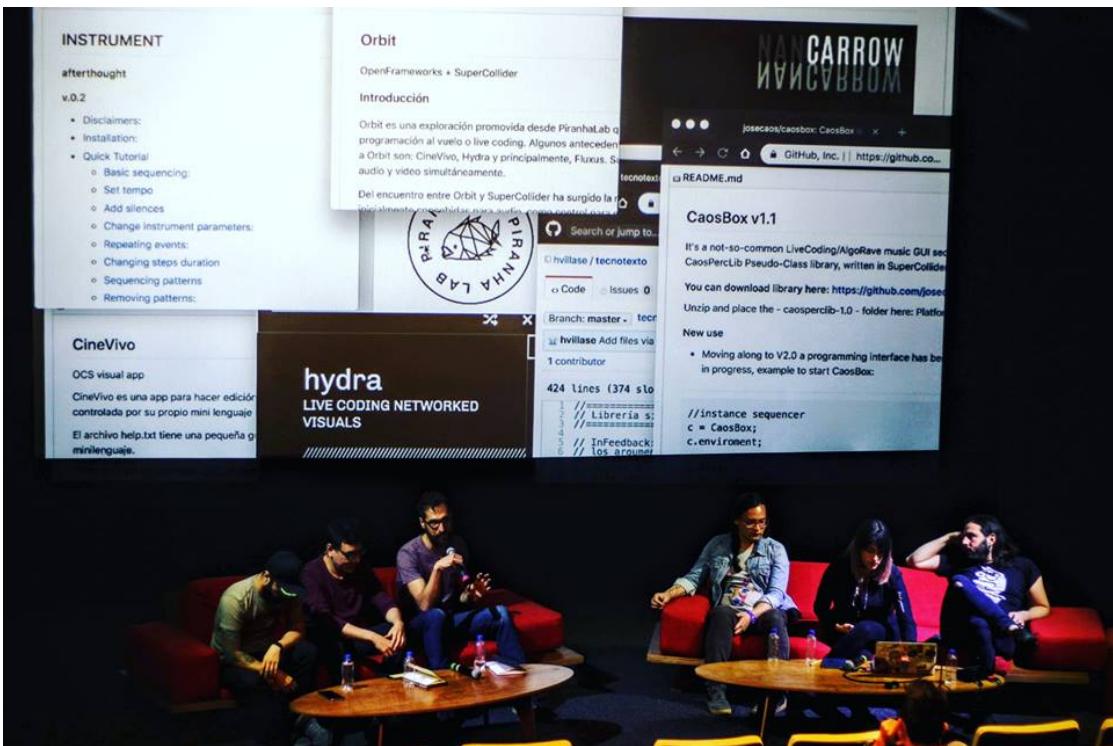


Figura 5: Conversatorio organizado por PiranhaLab y el Laboratorio de Tecnologías Compartidas en el CCD. 2019.

proyecto. El artículo *Panorama* (?) hace referencia de manera extensa al ecosistema de espacios y propuestas que también inciden en *Tres Estudios Abiertos*. Espacios inmersivos que estuvieron activos de manera simultánea.