

University of Texas Rio Grande Valley

ScholarWorks @ UTRGV

School of Music Faculty Publications and
Presentations

College of Fine Arts

2021

Collab-Hub: A system for creating telematic musical performance environments for Max and web-based instruments

Nick Hwang

Anthony T. Marasco

The University of Texas Rio Grande Valley

Eric Sheffield

Follow this and additional works at: https://scholarworks.utrgv.edu/mus_fac



Part of the [Music Commons](#)

Recommended Citation

Hwang, Nick, Anthony T. Marasco, and Eric Sheffield. 2021. "Collab-Hub: A System for Creating Telematic Musical Performance Environments for Max and Web-Based Instruments." In , 2. Barcelona, Spain.

This Conference Proceeding is brought to you for free and open access by the College of Fine Arts at ScholarWorks @ UTRGV. It has been accepted for inclusion in School of Music Faculty Publications and Presentations by an authorized administrator of ScholarWorks @ UTRGV. For more information, please contact justin.white@utrgv.edu, william.flores01@utrgv.edu.

Collab-Hub: A system for creating telematic musical performance environments for Max and web-based instruments

Nick Hwang
The Media Arts & Game
Development Program
University of Wisconsin -
Whitewater
hwangn@uww.edu

Anthony T. Marasco
School of Music
The University of Texas Rio
Grande Valley
anthony.marasco@utrgv.edu

Eric Sheffield
Department of Music and
Theater Arts
SUNY Broome
sheffielder@sunybroome.edu

ABSTRACT

Collab-Hub is a networking tool for sharing data across the internet for multimedia collaboration. Utilizing the Max graphical coding environment and its integrated Node.js functionality, Collab-Hub allows for multiple users to send data easily and remotely so they can collaborate with that data in real-time.

In this talk, we discuss the design of the Collab-Hub framework and provide examples of using it to create scalable and reconfigurable interaction layouts between Max patches and web-based instruments/interfaces. An analysis of pertinent historical precedents highlights the advantages Collab-Hub provides to artists who have little to no web development experience or to those who may be new to creating telematic and remote performance environments. A showcase of works created with Collab-Hub demonstrates the wide variety of artistic endeavors made possible through the framework.

1. INTRODUCTION

Collab-Hub¹ was born out of a desire to create local and wide area networked musical performances that incorporate audience participation and multi-user collaboration simultaneously. While a number of web-centric frameworks exist for simplifying the communication process between users in Distributed and Networked Musical Performances (notably NexusHub by Jesse Allison² and Soundworks by Sébastien Robaszekiewicz and Norbert Schnell[2]), we chose to center our efforts on brining these concepts to a wider audience of musicians who many not have experience with JavaScript but are eager to telematically collaborate with fellow electroacoustic performers. Our interests in expanding the device set available for networked pieces that utilize the We-

¹More information on the system along with a tutorial and example interfaces can be found at <https://github.com/rioter00/Collab-Hub/blob/master/docs/index.md>

²<https://github.com/nexus-js/nexusHub>



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** owner/author(s).

Web Audio Conference WAC-2021, June 21–23, 2021, Barcelona, Spain.

© 2021 Copyright held by the owner/author(s).



Figure 1: The Collab-Hub logo

bAudio API as well as our backgrounds as avid designers of software instruments led us to explore methods of integrating web techniques used to share control data over a network with the user-friendly audiovisual and DSP capabilities of Cycling74's popular Max software. Users who develop instruments within the Max environment can now easily control a fellow performer's patch and relinquish control of their own instrument's parameters through the addition of our Collab-Hub abstractions and bpatcher modules. Through on-the-fly modifications, the routing of control data throughout a patch or to any connected web-based instruments can be manipulated mid-performance, allowing for endless customization and improvisation. Beyond remote collaboration, Collab-Hub can also be used in other contexts regardless of locality, serving as a vital performance tool for laptop orchestras, multi-channel speaker arrays, video games, net art, multi-site installations, Internet of Musical Things devices[1], and video art.

2. FEATURES OF COLLAB-HUB

During the design phase, we focused on creating a system that would allow users to easily set-up and modify their preferred methods of sharing data and collaborating amongst each other in performance. If desired, the types of data and the routing topologies can be easily altered on the fly in client Max patches, embracing that program's penchant for allowing quick-and-easy editing of a software instrument while in use.

When sending data to the server, Collab-Hub requires all users to pre-pended all data messages with one of three labels: Control, Event, and Chat. Data sent with the Control label

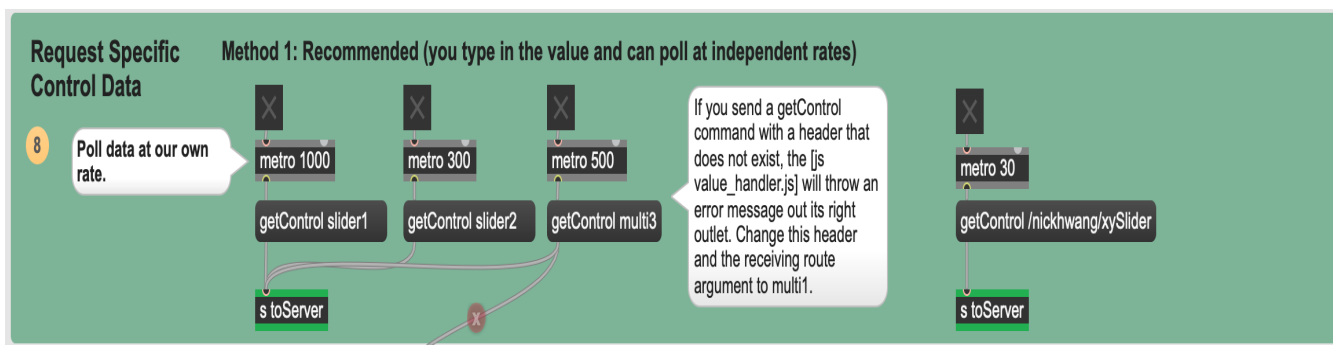


Figure 2: Example of Max patch for requesting specific data from the Collab-Hub server in performance.

is meant to be used for continuous control of an instrument’s parameters. An Event is an instantaneous occurrence, such as a button press, a toggled boolean state change, a signifier to start or stop a performance or playback action, or an indication of a section change in a scored piece. Data sent with the Chat label is designed for communicating directly with specific or all connected users through text messages.

2.1 Basic Data Routing

All connected clients can receive a collection of every message that has arrived at the server if they’d like, allowing them to monitor all activity occurring amongst collaborators during a performance. To tailor their specific Max patch or web instruments to be controlled by one of more specific collaborators, a user may choose to only grab select Control and Event-labeled messages from the server and map that data onto various parameters of their Max or web instruments. To simplify this routing process, users must add a Header string immediately following the Control or Event label and before the message’s data value (only applicable for Control-labeled messages).

2.2 Customizable Routing Topologies

Collab-Hub offers different methods for designing routing topologies between collaborators in a telematic performance. Users are not limited to committing to single routing topology model in a performance and can design works that switch between each of these models at a whim. When designing our routing topology methods, we felt that it was crucial to focus more on how we wanted clients to be able to interact with each other instead of simply adapting a traditional client-server communication scheme and sticking to that one model. With Collab-Hub’s routing models, users can send any data to any people privately and/or they can send data designed to be grabbed (“observed”) by anyone and everyone. This led to our two core transmission modes - Public and Private - the former of which follows the observer model of communication previously described, while the latter is similar to other existing networking direct communication protocols (such as OSC).

On a more macro-level of organization, users can take part in a performance set inside of a segregated Namespace. Additionally, users within a Namespace can further be divided up into separate Rooms. These advanced layers of routing can be helpful for ensembles (relegated to their own unique Namespace on the server) who wish to design a series of different telematic works, each using specific instruments and

a specific sub-set of performers (with each set of performers placed into specific Rooms). Using the ‘Private’ communication model, users can direct messages using a Target flag followed by either another user’s name or a room.

3. WORKS FEATURING COLLAB-HUB

Collab-Hub has been used to create a number of networked multimedia projects. These projects include:

- *SHP of THESEUS*— a composition by the authors and performers Anna Weisling and Jeff Herriot—which features a series of labeled graphical images that are sequentially presented to each performer. When performers receive a “sound”-labeled image, they interpret their image and contribute sonically. When performers receive a “control”-labeled image, they interpret their image and send control data to the performers’ instruments, affecting the gestures performed by others.³
- *Rhumb-Line*—a multimedia web-streamed installation by Margaret Schedel, Robert Cosgrove, and Brian Smith—which allows users to perform telematically physical wooden frog sculptures and hear/view their sonic output through a live audiovisual web stream.⁴
- Experiments in real-time interaction through networked musical performances undertaken by Marc Ainger and Federico Cámara Halac of the Advanced Computing Center for the Arts and Design at Ohio State University.⁵

4. REFERENCES

- [1] A. T. Marasco and J. Allison. Connecting web audio to cyber-hacked instruments in performance. In *Proceedings of the International Web Audio Conference, WAC ’19*, Trondheim, December 2019. Norwegian University of Science and Technology.
- [2] S. Robaszkiewicz and N. Schnell. Soundworks - a playground for artists and developers to create collaborative mobile web performances. In S. Goldszmidt, N. Schnell, V. Saiz, and B. Matuszewski, editors, *Proceedings of the International Web Audio Conference, WAC ’15*, Paris, France, January 2015. IRCAM.

³<https://www.youtube.com/watch?v=BEIwRtYClOQ>

⁴<https://hwang-schedel-collab.herokuapp.com/>

⁵<https://accad.osu.edu/research-gallery/networked-performance>