# anywhere and here: zcreative, a toolkit for distributed control

Benedict R. Gaster
benedict.gaster@uwe.ac.uk
University of West of England
Bristol, UK

Nathan Renney
nathan.renney@uwe.ac.uk
University of West of England
Bristol, UK

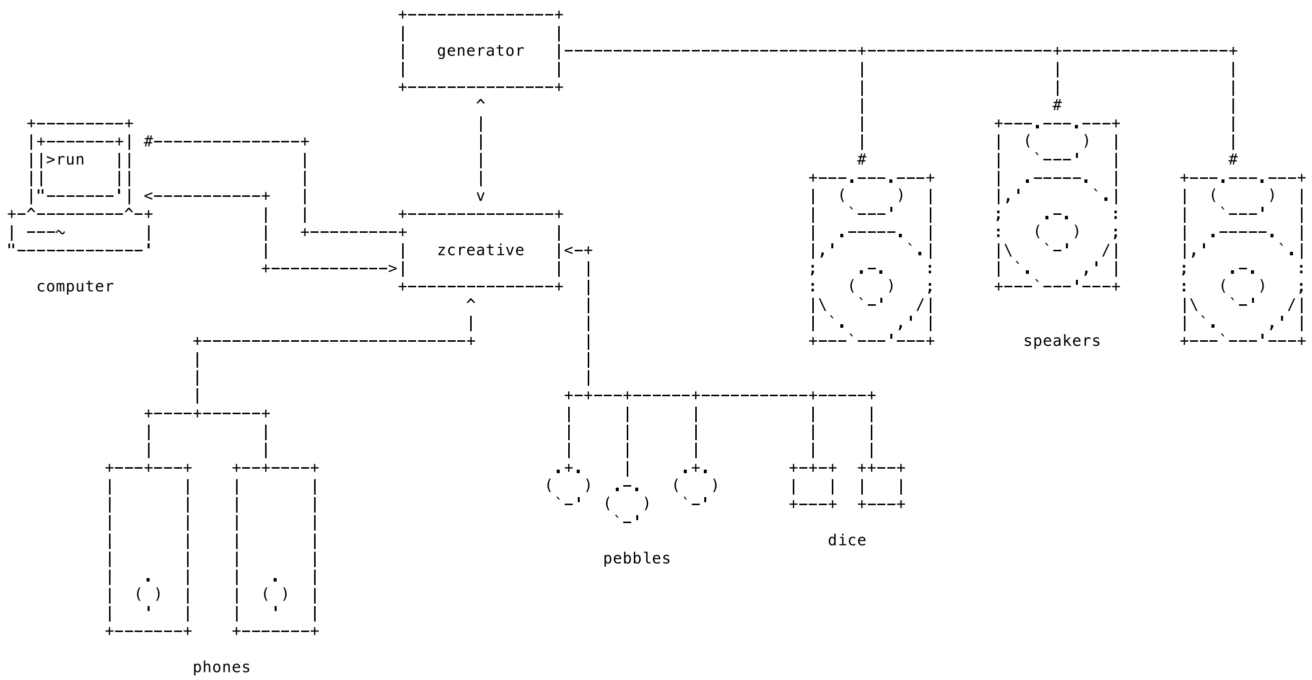


**Figure 1: Networked comms with *zcreative*.**

## Abstract

This paper is about sound and by extension the design of musical instruments through the window of shared play, memory, place, and materiality. It looks at NIME design within the context of storytelling. Exploring the design of a distributed musical controller and its use for performance and tabletop role-playing games that incorporate sound. What does it mean for a musical performance to be a community, or a game to create a story on the one hand and be an instrument on the other?

We play with this question through the design of a new musical instrument toolkit, *zcreative*, which is a set of autonomously connected controls, forming a network of concurrent controllers, that manipulate a whole.

*zcreative* is explored through three examples, considering what a NIME might be in a web of interactions. *FM for 8 phones* is a multi-channel spatial sound performance, where the listeners are also performers; *radical interaction* opens up a NIME themed podcast through community listening; and *pebble* is a sound driven tabletop role-playing game, where sound is first class and forms an additional creative axis for storytelling.


*NIME '26, June 23–26, 2026, London, UK*


## Keywords

Collaborative, DMI Design, Networked control, Agency, Design

## 1 Introduction

> What does your imaginary machine do and what does it sound like?
>
> *Kristina Andersen 2013*

In a 2013 lecture introducing her magic machine workshops, Kristina Andersen reflects on a device built by a young person, an imaginary machine that is designed for them to be "anywhere but here" [42]! The description that Andersen gives is at once melancholy, why does this teenager feel this way, while at the same time, when seen through the thoughts of a designer, inspiring. Invoking images of a younger viewer, memory and place brought into sharp focus with a short sentence. But why inspiring? Viewed from a design point of view, it is such a simple and clean idea, there are no preconceptions of what a machine is, what a device is, what a computer is, what the future is! Andersen lets the devices happen, instead of preconceiving a participant's wants and needs from or for a given contraption.

As Andersen herself notes, participants of her magic machine workshops are not alone in imagining what 'unreal' contraptions might do for us, children play in worlds dreamed up from their endless imagination, stories of myths, ghost stories, strange happenings passed down through generations and genres. Such fantasy demonstrates our desire and capability for magic. Andersen defines magical thinking, in this context, as the making of hypothetical things. Or contradictorily maybe we might think of magic as the unimaginable.

Wait a minute, why would a NIME reader and designer of musical instruments be reading about magic machines and moreover, roll playing games? How does this impact my daily thoughts about the next cool instrument I'm going to build? The short answer—design! Design is at the heart of all the devices we build and while there are a multitude of reasons to why we choose to build a particular sound making device, at some point they all go through a form of design process. This paper is about that process and what we can learn from it, in the context of storytelling through the lens of memory, place, and materiality. How can the development of new or old NIMEs be designed through a view of a shared history of time, place and the material?

Returning again to Andersen's lecture on her magic machines, she points to her reading of Karen Barad's notion of entanglement, emphasising how this leads to alternative thinking within the context of making [14], how the past interplays with the future, and how the future can, conceptually, impact ones reading of a shared history. How does this effect or relate to NIMEs? We argue that this highlights a reading of an instrument as an ongoing process; once it is crafted, the player must reinvent that instrument in their own way, with their own understanding, and it is here where the facets of Andersen's magic machines, Barad's agential realism, and more generally instrument design are at their most powerful.

Musing on the implications of Andersen's magical machines (cf. [11]) it seems only natural to consider a side of design and in particular instrument design, that sometimes feels neglected. What does it mean to tell stores in a time of digital overload, where often, we outsource our creativity and arguably our agency to the algorithms [52]? Garner expresses this notion in a thought that the art of storytelling has faded away as an oral tradition within the space of the British Isles, for example [28]. In this paper we propose a system, called *zcreative* for building networked instruments that can be utilized for building entanglements that can contribute to the notion of music or sound making as storytelling. An abstract overview of *zcreative* is given in Figure 1, capturing the entangled or networked nature of the system.

Wright [60] and Hollan and Stornettawith [30] argue that attempting to overcome the latency and bandwidth limitations of computer networks for live music is difficult! *zcreative* reacts to this limitation as a design constraint, leaning into its limits as an artist's bow, rather than trying to find an arrow to pierce through physical barriers. For accessibility a key design goal of *zcreative*, our proposed platform for realising network controllers, is for clients to be available on a variety of platforms, including the web [2], which raises a question over utilising OSC, a common approach for networked controllers, as it is built upon UDP, a protocol not accessible to modern browsers [6]. To address this issue we introduce *webosc*, a new bi-directional protocol layered on WebSockets [7] for client-server communication on the web, that supports transmission of OSC messages.

Bringing these somewhat disparate thoughts and technical notions together, this paper considers the following questions:

- RQ1. What would a sound toolkit look like for exploring Andersen's young person's design goal of *anywhere but here* or in our case *anywhere and here*?
- RQ2. How might we explore this toolkit to develop unconventional musical instruments?

This paper explores and reflects on a system that considers these two questions. To approach the first question we introduce *zcreative*, a toolkit for building shared and entangled controllers, that live within a distributed space, possibly across the globe, with the connecting plane hosted via the internet. Alternatively it can be a simple concurrent controller system used by one or more people within a locally shared space. Our proposed system is presented in Section 3.

We approach the second question by presenting three uses of *zcreative* in practice, as documented in Section 4. The first example is a networked spatial audio performance where the audience are also the performers and aspects of control for each channel are shared through a *zcreative* interface. The second example looks at radically breaking down the shared experience of listening to a podcast, itself about and by makers and musicians from the NIME community, providing a shared listening experience. In this context *zcreative* provides a controller interface that is abstracted from the listener, who interact with the experience through a webzine for the podcast. Finally, the third example explores the use of sound, via a dynamic *zcreative* interface, in the development of a table top role playing game (TTRPG) [4, 46], exploring both how sound itself might add to these creative storytelling games and how *zcreative* fits within this domain. This later example also demonstrates how autonomous embedded devices can be incorporated within *zcreative*'s control, motivating some of the design choices made when building the overall system.

Before diving into *zcreative*'s design and implementation, and the example use cases, Section 2, presents a selection of related work. Finally, Section 5, concludes with a discussion and pointers to future work.

To conclude the introduction we note that the networked nature of the paper sits closely with this year's conference theme, reaching beyond a more conventional view of what is a musical instrument, proposing to learn from and contribute to creative storytelling communities.

## 2 Background

Early HCI work by Hollan and Scott argue that it is possible to work with in the field of telecommunications and create systems that allow the same richness and variety of interaction, but with distance no longer an issue [30]. In this paper we continue to push the idea of "being there" or as they put it "beyond being there", enabling an entangled presence of time and place. Hwang and Marasco have explored a similar idea of distributed control, in their system MoNoDeC, that combines control with audience participation [31, 44]. MoNoDeC is built upon Collab [36], a system similar in scope to *zcreative*, although it differs slightly in that it is designed specifally for sharing data across the internet for multimedia collaboration. *zcreative* is designed for multiple forms of network bankbone communication, the internet being just one, of many, options. *zcreative* follows in the footsteps of a number of earlier web-based platforms for shared control, such as Soundworks, NexusHUB, and Rhizome [10, 53, 55].

O2 looks at inter-process communication of muscial devices through a new protocol that is similar to OSC, although with additional features such as syncronised clocks [24]. It differs to *zcreative* in that is takes a services rather than network address approach, and both client and final destination must support the O2 protocol. Similar the Σ protocol builds on OSC to build a more flexiable network protocol for muscial expression [25]. While similar in some aspects to *webosc* it not a system, in itself, for building shared control systems.

Mesh Garden, a creative-based musical game for participatory musical performance, takes a different approach to *zcreative*, instead exploring participatory music-making with smartphones using an audio sequencer game made up of a distributed smartphone speaker system [21]. Rather than presenting a general framework for building shared connected instruments, Mesh garden focuses in on a single application. Similarly TweetDreams is an instrument and musical composition that participants contrabute in music-making, creating tweets in real-time that are sonified and visualised [23].

An expanding application area for *zcreative* is within multi-channel speaker arrays and there is a long history within this domain, summarised by Taylor [59]. While not the original design focus of *zcreative*, approaches like MoNoDec and applications like *FM for 8 Phones* open up an entangled approach to multi-channel performances and instruments.

Robson explores the physical entanglement of listeners within a performance, incororating non-human influences through site and technology that informs the production process [48–50]. The distributed nature of *zcreative* naturally leads to shared interactions, and performances such as *FM for 8 Phones* (see Section 4.1) sit closely with entangled activities of composing and curating described by Robson.

Many works exist that involve wearables and external movement in terms of both sound generation and more narrowly in game design and in paritcular TTRPGs. See for example [17–19, 57]. More recently work in this domain has been explored within the context of GenerativeAI, including [16, 51].

There are many examples of work that explore the direct use of sound within the context of games and NIME design, although mostly not conbined within a distributed and shared gameplay [13, 27, 45, 56, 58] . Collin and Kapralos reflect that in video games, sound plays an important role, often less typical in interactice media, with an emphasis placed on the visual instead [22]. Storybit's and Jigsaw push aginst this, exploring interactive and immersive nattatives with micro:bits and paper and augmented reality and networked game play [33].

Working within the context of NIME design brings into sharp focus the evolving environmental crisis. The creation of NIME research often relies heavily on, and is tightly intertwined with, mass production and industrialisation [54]. As such throughout the development of the ideas presented in this paper we considered and applied ideas from Becker's recasting of computing through the lens of substainability [15]. Looking to this work for a more thoughtful path for development, *zcreative* is designed to minimize compute resources, when possible running locally on a laptop and private local network, rather than a cloud based solution, for example.

*zcreative* and the presented applications were built and designed with sustainability in mind, building on and inspired by recent work at NIME, for example, the circular maker practice of Anon et al [5] and Masu et al's reflections on the importanace of re-purposing old musicial instruments [37], and approaches such Rigaud's re-purposing of old devices [47]. Throughout the design and making process we considered a bricolage approach, trying to make use of what was around us and on hand, rather than looking for the newest thing [12]. This is most obvious in

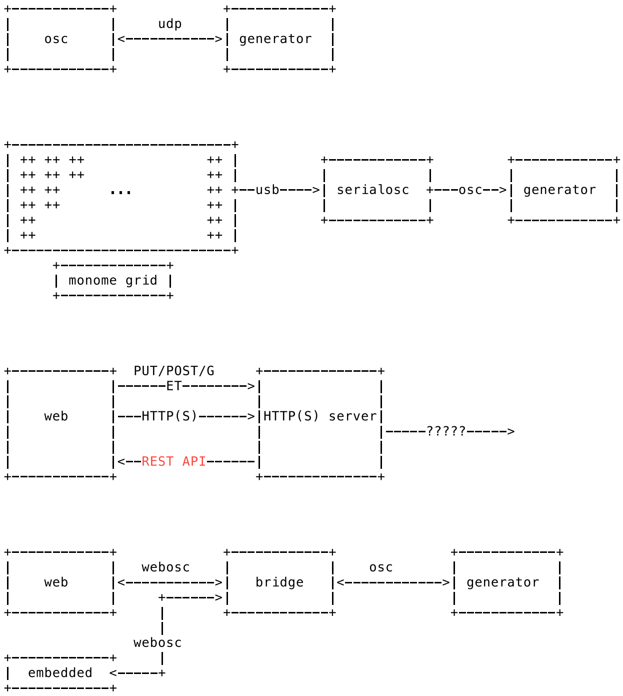


**Figure 2: Protocol(s) for sending control data.**

*pebble* (see Section 4.3), which incorporates found clay, old discarded paper, and re-purposed microcontrollers in the making of physical components for the game.

## 3 zcreative

In this section we introduce *zcreative* a musical platform designed for exploring distributed NIMEs. While the system lends itself to supporting unlimited participants, they are envisioned as controllers rather than sound generators, *zcreative* is what Magnusson calls the mapping engine, with the instrument model completed here with a sound generator written in MaxMSP[1] [35]. Clients can subscribe to the sound generator's audio streams, using Opus Codec [3], for example, although in many applications, as demonstrated in two of the three case studies, this is not necessary.

*zcreative* provides a standard client server architecture, supporting two types of clients, either using a bidirectional or more limited unidirectional protocol. The protocol itself is implemented on top of an extended version of OSC [60, 61], called *webosc* that builds an OSC like layer on top of Websockets (described in Section 3.1). A high-level overview of *zcreative*'s architecture is shown in Figure 1.

The general, bidirectional protocol, enables clients to subscribe to controller or audio channels, in one of three different modes, read, write, and read/write, and additionally these channels can have multiple clients attached in any access mode. As *zcreative* is intended as a tool for exploring shared connections, the default mode of access is shared, although this is not a hard requirement and each channel can be configured with different access capabilities. While it is possible for clients to create and destroy channels, this is not something we have explored to date, rather a particular instance of *zcreative* is described statically, i.e. fixed, and expressed as a JSON configuration, loaded directly by the

[1] Our of Max should not way imply this is the only possible sound generator.

server when it is initialised. Currently, the static configuration nature of *zcreative* is deliberate and by choosing to enforce this limitation the focus is on applications that are constrained to a fixed centralised notion of control. We acknowledge that this limits the set of possible applications and in practice the is not a hard requirement of *zcreative*, however, we felt it interesting to focus on use cases, which are many, that fit this model of constraint.

While bidirectional control enables clients to both send and receive changes in the state of the system, e.g. when a control slider is moved either by the client or another remote user, some controls do not require the ability to receive state changes. This is of particular importance for embedded controllers, that might have actuator controls only. An example, of this might be a simple slider or button interface, and can be seen in action in the *pebble* game described in Section 4.3, where a small microcontroller and a 6-axis IMU are embedded within a handmade clay pebble, and connect via *zcreative*'s uni-directional protocol to send rotational information to the sound generator.

It is important to note that a networked server, as provided by *zcreative*, is always going to be limited in terms of latency induced by the physical constraints of the hardware, the amount of network bandwidth, and so on. To this end we decided to lean into this as a constraint and the system is not optimized for low-latency, i.e, action to sound [40, 41], focusing, instead on a robust implementation. Of course, the implementation is optimized and the server itself is implemented in the systems programming language Zig [1].

### 3.1 webosc

OSC is a protocol that enables musical controllers to communicate over a computer network, supporting control from remote devices [60, 61]. Unlike MIDI [38], OSC can encode and control almost anything, see for example Jones' complete model for controlling Ableton Live [32], and is applicable beyond just music applications. However, unlike MIDI, OSC has yet to be supported within the browser and is not directly accessible to the web developer, which given the wide accessibility of web technology can be seen as a major limitation.

Due to the web's focus on security and reliability there has been an understandable reluctance to expose Berkeley style sockets [34] to web applications. This lack of bi-directional communication between client and server, not withstanding stateless encodings offered by true REST APIs [26], limited web clients to communicating with servers through HTTP's PUT/GET/POST API, which in turn constrains the set of possible applications [2]. WebSockets were introduced to address this, built on top of HTTP itself they enable an existing HTTP connection to be upgraded to support bidirectional communication [7]. So if we are to support bidirectional OSC messages for web applications, then OSC's transport layer needs to instanciated using WebSockets. While OSC is transport independent its implementation, as originally, defined is to transmit over a socket, most often via UDP, and so does not map one-to-one onto WebSockets, which behaves similar to a standard socket where it is encoded over the original HTTP connection and thus happens at the application layer [8]. The implication here is that if we want to support OSC for web applications, without changes to browser APIs, then OSC needs to be formalised over WebSockets. Figure 3, shows this concept diagrammatically.

```
+------------+             +------------+              +------------+
|            | websocket   |            |     osc      |            |
|   webosc   |<----------->| zcreative  |------------->|   maxmsp   |
|            |             |            |              |            |
+------------+             +------------+              +------------+
```

**Figure 3: General communication network for *zcreative*.**

The idea of translating one protocol into another is as old as computing itself and *zcreative* is not novel in this regard. However, we argue that by conceiving *zcreative*'s server as a bridge between control client(s) and sound generator(s), in our case written in MaxMSP with OSC, it enables clients to utilise different technologies and address cases where network technology is limited, e.g. no UDP on the web. Formalising this design choice and the benefits it enables with integrating somewhat competing technologies enables a generalisation of OSC as a control technology within the application layer, rather than the more traditional transport layer. Figure 2, shows a progression in possible mappings for OSC in different scenarios. Example (A) recalls the traditional approach to transmitting OSC messages, while (B) highlights a non-standard approach, used by Monome to connect its Grid and Arc controllers to a computer sending closed OSC messages over a wired serial port to a small piece of software (serialosc), running on the host computer converting incoming signals to OSC messages that are forwarded over UDP to a receiving sound generator, such as Monome's own Norns computer. (C) shows a possible controller encoding over conventional HTTP, with the idea to encode state via REST style API, but limited by the fact that the server can only respond synchronously to client messages. Finally, (D) formalizes *webosc*, the idea introduced with *zcreative*. Inspired by Monome's *serialosc*, we propose that OSC messages are encoded on top of WebSockets, enabling bi-directional communication between client and a bridging server, the later of which is responsible for sending and receiving OSC messages from the wider network. The idea of a bridge is similar to that supported by the O2 protocol, while in practice a bridge is not required for all applications [24].

In practice a *webosc* client behaves much like any other OSC application, but there is an interesting design point that must be considered when using this protocol; does the client know and control where the OSC messages are destined to go on from the *bridge*? For example, in scenario (A) the client sends an OSC message directly to a chosen destination, i.e. it knows the IP address and port number, but in use case (D) is this the case? This comes down to if the *bridge* is transparent or not, thus behaving in a similar fashion to a network switch, or does it behave more like a router and thus enabling, at least conceptually, a remapping of the destination? We believe that both use cases are useful, for example a transparent *bridge* is simply there as a conduit for "switching" messages along the network, while a *bridge* that is visible in the pipeline can be seen as an actor that can play a semantic role in the overall system. *zcreative* is an instance of the latter, where it not only forwards on OSC messages to a sound generator, e.g. MaxMSP, but also maintains a stateful representation of the control system, that supports the ability for clients to subscribe to controls, with read only, write only, and read/write permissions. In theory, *zcreative* can support a wide selection of protocols for communicating with the client, however, for our current implementation OSC is used exclusively for communication with MaxMSP, which is highlighted in Figure 12, showing the communication architecture for the *pebble* game described in Section 4.3.

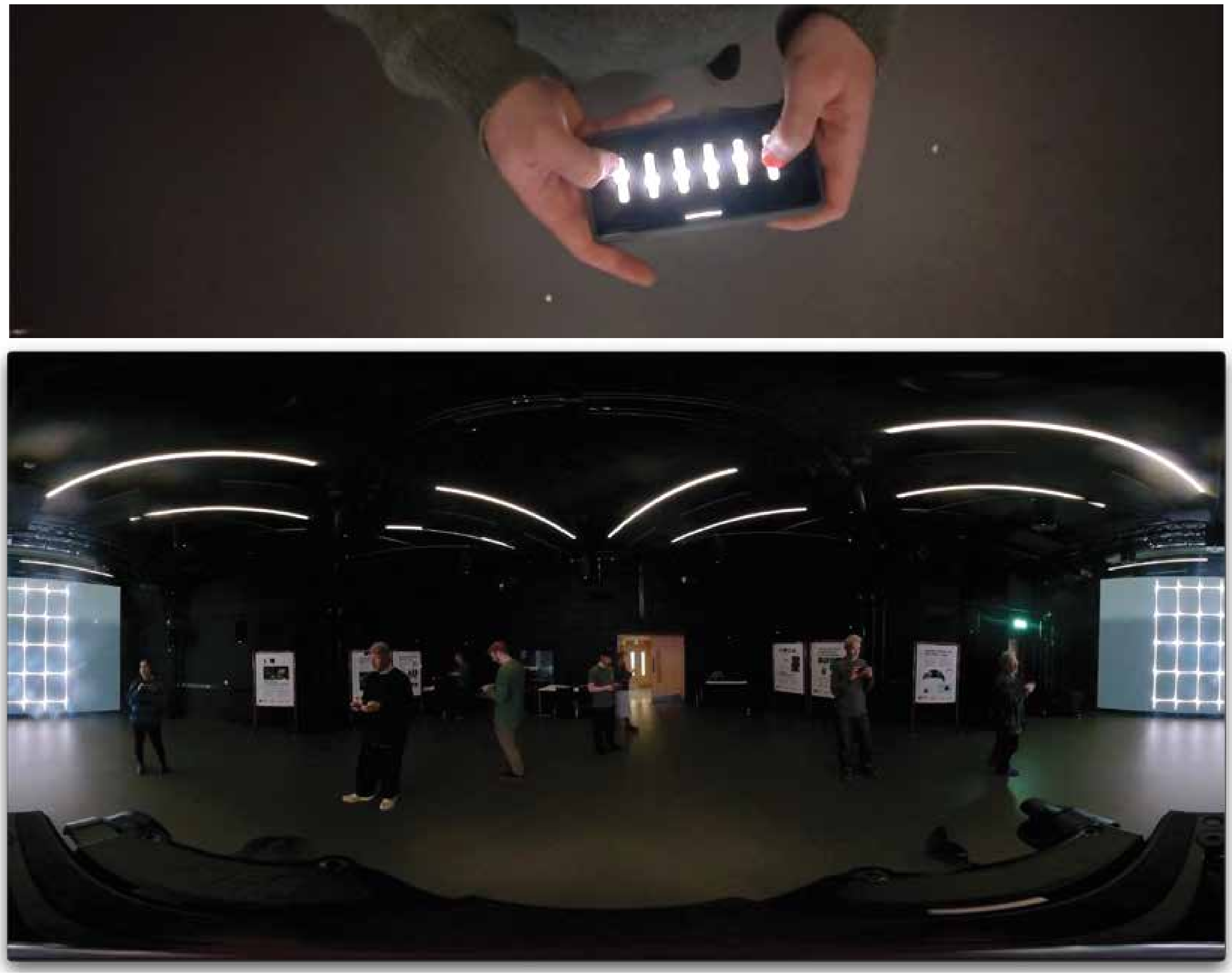

**Figure 4: Stills from FM for 8 phones performance.**

## 4 shared control

To put *zcreative* in context, building on the foundations laid out above, we developed three applications of networked control taken in part from our own artistic practice that is embroiled with our interest in computer science, networking and control, in conjunction with an ongoing interest in what defines a musical instrument.

These examples show the practical use of a system such as *zcreative*, while at the same time casting light into the world of networked systems, when bent beyond the physical limitations of latency and space. Musical storytelling can be one of shared connections, where materiality reaches within the immediate physical space and beyond, into the virtual.

### 4.1 FM for 8 phones

As part of a larger research project we were lucky to be invited to create a performance at The Bridge Studios[2], an interdisciplinary research lab and a state of the art creative space for physical-meta production. The Performance Space provides a $28m^2$ LED production-quality volume wall and 24 speaker (plus 4 subs)

```
+-----+
|in1 | ------+
+-----+      |
             |                              +---------+
            ++------------+                 | cycle~  |
            | frequency   +---------+       +---------+
+-----+     +-------------+         |            |
|in2 +-------+                      |            |
+-----+      |                      |            |
            ++------------+         |            |
            |     q       +---------+------------+
            +-------------+         |
+-----+                             |
|in3 |-------+                      |
+-----+      |                     ++----------------------+
            ++------------+        |                      |
            |    gain     |----+   |abl.device.autofilter~|
            +-------------+    |   |                      |
                               |   ++----------------------+
                               |    |
                               |    |
                               |   ++----------------------+
                               |   |                      |
                               |   |  abl.device.reverb~  |
                               |   |                      |
                               |   ++----------------------+
                               |    |
                               |    |
                               +---++--------+
                                   |  gain~  |
                                   +---------+
```

**Figure 5: Abstract representation of participant's controls.**

[2]https://www.uwe.ac.uk/research/centres-and-groups/cate-research-overview/the-bridge-studios.

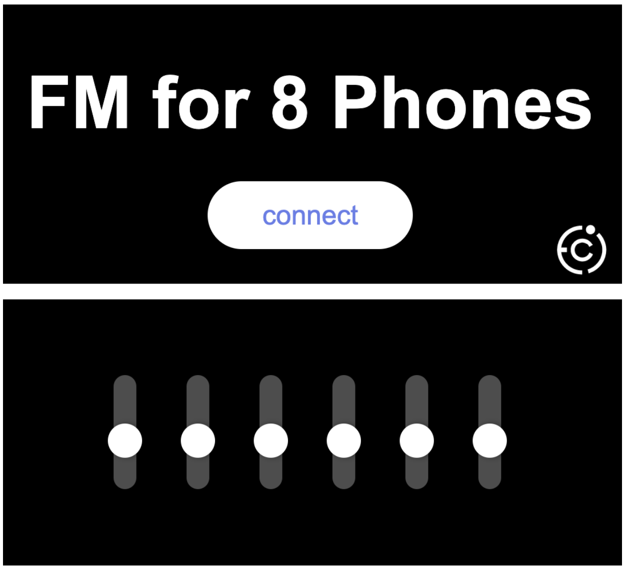


**Figure 6: FM for 8 phones application interface.**

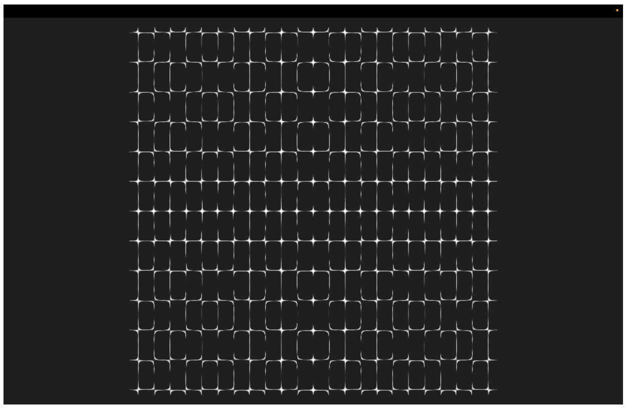

**Figure 7: FM for 8 phones performance visualization.**

Martin Audio surround sound system, based on a Dante network[3]. Inspired by the network nature of both *zcreative* and a Dante based sound system, along with the community foundations of The Bridge Studios we designed *FM for 8 phones*, a piece based on John Chowning's 1978 Stria [43]. Expanding Chowning's idea to 16 channels of audio and introducing an entangled set of dynamic controls for 8 performers, who are also the listeners.

The intention of the piece is to blur the line between the performer and the listener, where the 8 performers are also the audience, entangling their shared interaction through real time control of sound features for 2 channels (i.e. speakers) each. Control is provided through a simple webapp connected to an instance of *zcreative*'s standard networked architecture, as shown in Figure 3.

Setup for each participant requires connecting their phone to the local private network, opening their browser of choice and the provided URL[4]. In practice this is reasonably straightforward with only minimal support required.

As can be seen from the interface shown in Figure 6 the participant is confronted with a simple welcome screen (top image), that asks them to connect to the performance. Once connected an interface with 6 unlabeled sliders is presented, representing pairs of 3 controls for two randomly allocated channels. Each pair enables the performer to control the gain, frequency cutoff and Q value for that channel, as shown in Figure 5. Wrapping this patch with Max's multi-channel *mc.poly~*, with 16 channels, and combining with nodes *funnel*, which preappends the channel index, and *setvalue* enables each participant's control inputs to be routed to the corresponding instance.

A participant's control message for each input is sent via *webosc* from the app to a shared instance of *zcreative*, which forwards on the corresponding OSC message, with the slider value, to MAX, which is matched and routed as described above. When a new participant initialises connection with *zcreative* 2 channels are randomly allocated and the corresponding controls are sent to the connecting app, and the uniquely assigned ID provides a mapping from incoming values to outgoing OSC messages. In this application no additional temporal state is required.

Originally it was conceived that the interface of the digital instrument would be visualised during the performance, enabling some visual feedback of a performers controller commands. However after early prototyping it was found that obscuring the interface from the audience could induce a deeper listening experience. This is not to say that visual feedback was not a key element of the performance, in fact, The Immersion Studio's large LED array sits at the front overlooking the whole room, providing a visual dimension to the speaker array.

For the piece the large LED array was utilised, with different standing wave patterns[5] built from one of the channel's frequencies, picked at random, and then modulated by the carrier's modulator of other channels. A snapshot image is shown in Figure 7. While subtle, choosing the carrier at random and changing this selection at the same rate as the channel sequencer, means that it is possible to see the entangled interaction of the performers, as well as hearing them.

Stills from the first performance of FM for 8 phones can be seen in Figure 4, the top image shows a POV recording of one listener/performer interacting with the 6 controls, while the second is a flattened 360 image of the space with the participants. 360 degree and POV video recordings of a performance at The Bridge can be found at [Link] and [Link], respectively. A stereo recording of the performance is included as part of the paper's supplemental material.

## 4.2 radical interaction

*radical interaction* is a podcast series that is inspired through the lens of research and interaction [29]. The first series focuses on NIME researchers and is a set of eight interviews with broad participant interests, ranging from PhD students, established researchers, to well known artists. The second series, still in the early stages of development, considers programming language design and interaction, and we do not focus on it here. The first series explores three themes, shown in Figure 10, but in general, allowed each interview to consider paritcular interests of the participants[6].

To keep with the spirit of the podcast and its intention to break with podcast norms, we explored the idea of a podcast as

[3]https://www.getdante.com/meet-dante/what-is-dante/.
[4]*zcreative* runs on a secure private network and it is not connected to the internet, users must accept a locally signed certificate.
[5]This element of the patch was inspired by a similar one found on the Cycling74's website by @ringtone.
[6]For review we have avoided giving to much detail about participants, to keep it as anonymous as possible.


**Figure 8: One "side" of radical interaction zine.**

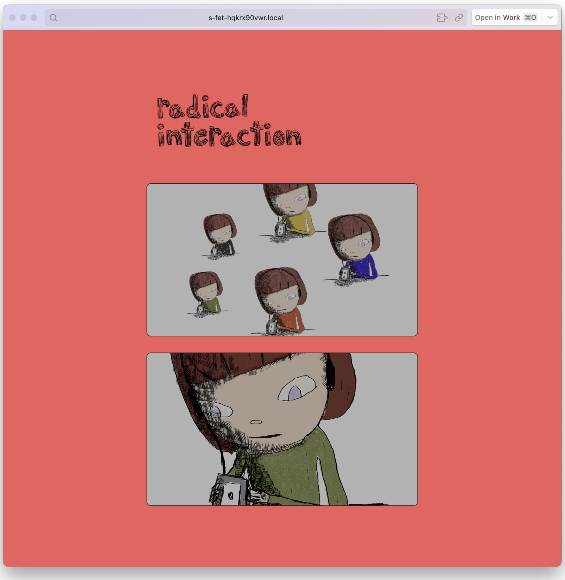


**Figure 9: radical interaction landing page.**

zine, delivered both in traditional paper format and digitally, and the listening experience itself can be individual or shared. While the paper zine is fun and interesting in its ownright, we forgo discussion here as it is not directly related to the presentation of *zcreative*, instead focusing on the digital version, which is implemented on top of *zcreative*.

The digital zine is delivered as a webapp that works in two modes; mode one presents the podcast through a community listening approach, using *zcreative* to manage the shared listening experience; while mode two presents a conventional linear listening, where each episode is presented as an embedded page that provides the same text as the paper zine, along with the ability to listen to that episode as audio. On opening the webapp the user is presented with the landing page shown in Figure 9. Clicking on the bottom image, i.e. the single listener, takes the user through a reasonably conventional listening experience for a podcast, where each episode can be played and text providing context read. However, it is worth noting that while the listening experience can be linear, the podcast itself is presented in the form of the zine, which in paper form folds out to a single double-sided A2 page that contains descriptions of each podcast and so on. The digital form is presented as an "infinite" canvas, with links embedded for playing episode's audio. An annomised one side of the zine is shown in Figure 8.

Selecting the community listening path, top image in Figure 9, connects the digital zine to *zcreative*, allocating a UUID and a randomly allocated pitch that is used for podcast snippets. The user is presented with a virtual representation of the paper zine, where moving around the canvas allows different elements of the podcast to play, while "avoiding" a particular episode casues random elements of any (or all) of the podcasts to play. For a given user they hear their pitch at the forefront of all played backed sounds. Unlike the linear listening experience they can also hear others that are present. Depending on where another user is on the canvas, i.e. are they hovering over a specific episode[7], the user may hear only clips playing at their allocated pitch, but as others join the shared reading and listening experience their allocated pitches are introduced. Each additional pitch is placed

[7]Clicking on a specific episode locks it in, until another click is recorded.

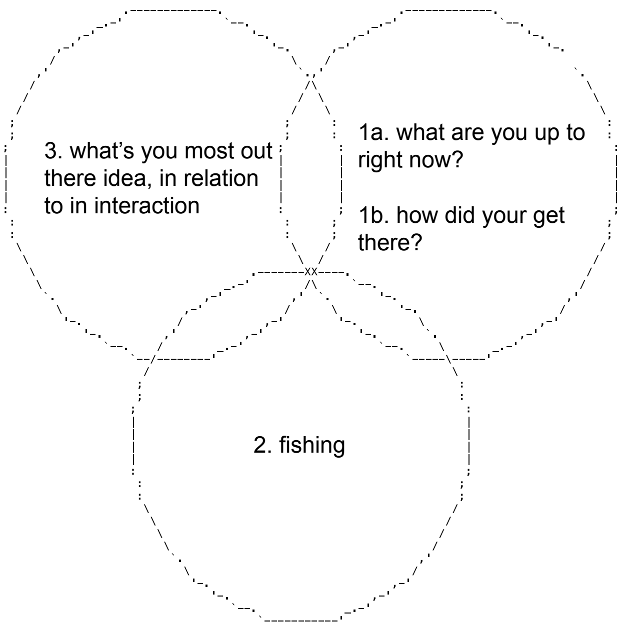


**Figure 10: radical interaction themes.**

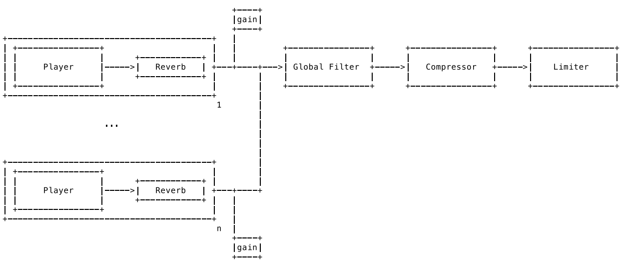


**Figure 11: radical interaction audio pipeline.**

in a 3D space by simple use of gain and reverb to give a more ghostly sense of community presence and shared listening.

*zcreative* provides the backbone of the shared listening experience tracking user pitches, current position and selection, and providing simple instructions to the frontend webapp of what pitch and particular set of samples to play. The webapp itself is a single page application, using HTML Canvas and SVGs to draw and represent the digital version of the zine, along with *Tone.js* [9] for handing the audio playback through Web Audio. The audio pipeline for each player, i.e. user, along with the global mix is shown Figure 11.

Unlike *FM for 8 phones* and the *pebble* game, described in the following Section 4.3, *zcreative* does not maintain controller information, as the zine does not have controllers in that sense, but state is required to keep track of concurrent users and their associated pitches. Pitches are allocated within a half octave, up and down, from a fixed zero playback pitch, i.e. not timestreched, and allocations are within that range. In practice, once there are above about twenty four online concurrent users the amount of pitch variation can become less discernible, however, through the use of per-player reverb it is still possible, with careful listening to determine a larger number of online users.

The *radical interaction* podcast zine can be listened to at the following URL:

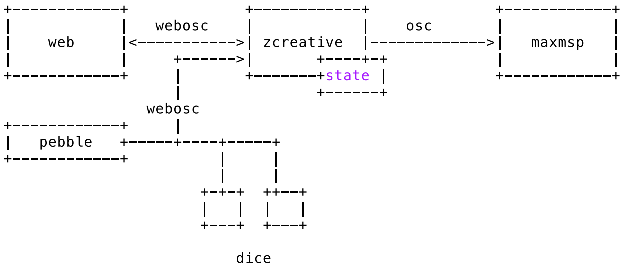


**Figure 12: Communication network for *pebble*.**

**Website:** https://radicalinteraction.org

## 4.3 pebble

*pebble* is a world-building game exploring sound as an aid for storytelling. Player's search for a way home by exploring locations where rocks abound.

A large inspiration came from the community of single page role playing games found on platforms such as itch.io, which has and continues to be an amazing source of creativity.

The original aim with *pebble* was to build a single page role playing game, which isn't a lot of space, so we focused on the following design goals:

- Sound should play a key role in playing the game;
- It should be easy to learn and accessible;
- It should encourage player creativity;
- It should be tight and have an ending.

Additionally we felt that a TTRPG was a great place to explore *zcreative* from an embedded device view point, in particular the community aspect was already manifest in the shared storytelling elements of role playing games and thus focused on *zcreative*'s centralized notion of control, rather than its implicit support for distributed interactions.

In *pebble* 3-6 players create lost explorers, discover strange lands, cursed people, exploring deadly landscapes of different locations. With fellow travelers they discuss how to survive and, when everything has been explored, choose a route home. A key element of the game's design is iteration, where players explore different routes home with similar game play. The use of iteration or loops in game play enables learning once and applying multiple times, thus reducing the likelihood that players experience cognitive overload [20].

Throughout *pebble* sound is used as a creative tool for storytelling, providing prompts for the players to expand the emerging environment. As the game progresses players interact with six "pebbles", connected via *webosc* to *zcreative*, that enable continuous control of different sounds. The sets of accessed sounds change with each iteration of route exploration. These interconnections are shown diagrammatically in Figure 12. The A3 design and text for *pebble* is included with the paper's supplemental material.

The heart of *pebble*'s game mechanics is a pair of nested loops, where players explore a route home, as shown in Figure 13. Each route is envisioned through first selecting one of six cards, each a themed drawing that is intended to invoke a sense of place and time, chosen by rolling a six-sided die and placed on the map. The cards can be seen in the bottom left of Figure 14. At this

```
                              Try a different route
                        +------------------------------+
                        |                              |
                        |                              |
                        v                              |
+----------------+    +----------------+    +--------+-------+    +----------------+
|                |    |                |    |                |    |                |
|     Setup      |--->|Explore a route |--->|   Scrapbook    |--->|  Pick a route  |
|                |    |                |    |                |    |                |
+----------------+    +-----------+----+    +----------------+    +----------------+
                          ^       |
                          |       |   Listen to sounds of
                          |       |   pebbles and interact with
                          +-------+   6 things
```

**Figure 13: *pebble*'s game loop.**


**Figure 14: Physical elements of *pebble* game.**

point one or two players are chosen to perform with one or more pebbles, which in turn causes the connected sound generator to produce sounds that creates a feeling of the weird, eerie, and a sense of three dimensions, with respect to the selected card. Each set of sounds is bespoke for the chosen card and there are multiple sets for each card, that are picked at random for each iteration of the game.

Sound set selection must be tied to each card and thus must be dynamically communicated to the sound generator, which itself is a simple Max patch for playing and controlling samples. Originally, we developed a button pebble that one of the players would click to indicate the card that had been pressed, however, after playing the game with a number of different groups the general consensus was the conventional affordances of a pebble were lost when it was cast as a button. To work around this constraint an alternative way of selecting the corresponding sound set was necessary and we considered two possible options; embed a NFC tag in each card and a NFC reader in each pebble, requiring the card to be swiped or tapped on a pebble; or embedding a communication device within the dice themselves, allowing rolling the device to select both the card and the corresponding sample set.

Using NFC tags introduces a new interaction into the game, while in the most part keeping with the intended affordances of a pebble, but a downside is the implementation cost. Adding a tag to each card would be straightforward, however, adding an NFC reader to each pebble would increase the complexity of the embedded technology within each pebble. It was fairly clear that to create smart dice an embedded IMU could be used, something we are already using for the pebbles themselves, however, with a new size constraint. A single dice is generally quite small and a custom PCB would be required and potentially worse still sourcing what would need to be a tiny lipo battery is not trivial. Luckily, role playing games are not the niche business of yesterday and in today's post COVID gaming world there already exits GoDice6[8], which are Bluetooth enabled dice, shown in the middle left picture in Figure 14. Utilizing GoDice's API it was straightforward to implement a driver that translates die roll notifications to *webosc* messages that are transmitted to *zcreative*, just like any other supported control signal. The dice, as per the pebbles themselves, communicate with *zcreative* using the unidirectional protocol, and the server itself was extended to support a new (die) controller type.

## 5 Conclusion

*zcreative* and its example use cases presented in this paper provide one response to our original research questions, framed and inspired by Kristina Andersen's magic machines. *zcreative* is a toolkit for exploring musical machines that have a shared and networked heart, providing a framework for exploring story telling and shared agency in a time of digital overload.

The three applications introduced in the previous section scratch only the surface of what might be possible with *zcreative*. Shown clearly by the work of Hwang and Marasco with MoNoDeC, a multichannel audio system that inspires to incorporate listeners and performers in an entanglment, similar in spirit to FM for 8 phones, that connects through the use of mobile phones and speakers, there are a multitude of application domains to explore with *zcreative*. Of particular, interest is the connected nature of these systems to contribute to shared story telling, through exploring its use for new interactions within spatial audio and how it might expand narrative story telling in TTRPGs. We plan to begin a wider exploration of *zcreative*'s use in the design of TTRPGS by placing it in the hands of game makers and players, asking them what they might use an entangled and networked tool for sound exploration in their own or future games.

[8]https://particula-tech.com/pages/godice

Clearly *zcreative* conceptually, does not have to be tied to sound control interfaces, and it could be interesting to consider its uses from a music control metaphor approach, similar to that of McNamara et el [39]. This aligns with our interest in TTRPGs and it seems reasonable to develop *zcreative* both for expanded audio lead storytelling, along with more general use for introducing dynamically connected materials, enhancing game play.

## 6 Ethical Standards

This work was ethically approved by our university's Faculty Research Ethics Committee. AI was not used in the creation of this paper.

## Acknowledgments

The inspiration for *zcreative* in part came from a workshop the $1^{st}$ author attended at NIME'25 run by Nick Hwang and Anthony Marasco, which introduced their system MoNoDeC, a multichannel audio system that uses audience mobile phones and IoT-hardware-driven speakers as point sources for configurable and dynamic immersive audio speakers and audience performance interfaces [44]. *zcreative* has different goals from MoNoDeC, although similar in many ways to its larger parent Collab-Hub [36], but I can say that had I not attended the NIME workshop, then the idea for *zcreative* and its possible applications would have stayed dormant for longer, if not forever.

FM for 8 phones was preformed and recorded at The Bridge Studios, an AHRC-funded Creative Research Collaboration Facility at the University of the West of England [grant reference: AH/X010201/1].

The physical pebbles were kindly made in found clay by Sonny Lightfoot.

## References

[1] [n. d.]. Zig Programming Language. https://ziglang.org/.
[2] 2025. HTTP. *Wikipedia* (Nov. 2025).
[3] 2025. Opus Codec. https://opus-codec.org/.
[4] 2025. Tabletop Role-Playing Game. *Wikipedia* (Oct. 2025).
[5] 2025. Title Redacted for Peer Review. *Journal of Redacted Studies* (2025). Author's previous work.
[6] 2025. Web Browser. *Wikipedia* (Nov. 2025).
[7] 2025. WebSocket. *Wikipedia* (Oct. 2025).
[8] 2025. X.225 : Information Technology – Open Systems Interconnection – Connection-oriented Session Protocol: Protocol Specification. https://www.itu.int/rec/T-REC-X.225-199511-I/en.
[9] 2026. Tone.Js. https://tonejs.github.io/.
[10] Jesse Allison, Yemin Oh, and Benjamin Taylor. 2013. NEXUS: Collaborative Performance for the Masses, Handling Instrument Interface Distribution through the Web. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2013)*. NIME'13, Daejeon + Seoul, Korea Republic.
[11] Kristina Andersen and Ron Wakkary. 2019. The Magic Machine Workshops: Making Personal Design Knowledge. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland Uk, 1–13. https://doi.org/10.1145/3290605.3300342
[12] Jack Armitage and Andrew McPherson. 2019. Bricolage in a Hybrid Digital Lutherie Context: A Workshop Study. In *Proceedings of the 14th International Audio Mostly Conference: A Journey in Sound (AM '19)*. Association for Computing Machinery, New York, NY, USA, 82–89. https://doi.org/10.1145/3356590.3356604
[13] Jan Banas, Razvan Paisa, Iakovos Vogiatzoglou, Francesco Grani, and Stefania Serafin. 2016. Design And Evaluation Of A Gesture Driven Wave Field Synthesis Auditory Game. In *NIME'16*. Zenodo. https://doi.org/10.5281/ZENODO.1175972
[14] Karen Barad. 2007. *Meeting the Universe Halfway: Quantum Physics and the Entanglement of Matter and Meaning.* Duke University Press, Durham, NC.
[15] Christoph Becker. 2023. *Insolvent: How to Reorient Computing for Just Sustainability.* The MIT Press, Cambridge.
[16] Josiah D. Boucher, Gillian Smith, and Yunus Doğan Telliel. 2025. How Do You Want to View This? Generative AI, Creative Ethos, and TTRPG Hobbyists. *Proc. ACM Hum.-Comput. Interact.* 9, 6 (Oct. 2025), GAMES035:995–GAMES035:1020. https://doi.org/10.1145/3748630
[17] Oğuz Turan Buruk, Ismet Melih Özbeyli, and Oğuzhan Özcan. 2017. Augmented Table-Top Role-Playing Game with Movement-Based Gameplay and Arm-Worn Devices. In *Proceedings of the 2017 ACM Conference Companion Publication on Designing Interactive Systems (DIS '17 Companion)*. Association for Computing Machinery, New York, NY, USA, 289–292. https://doi.org/10.1145/3064857.3079176
[18] Oğuz Turan Buruk, Ismet Melih Özbeyli, and Oğuzhan Özcan. 2017. WEARPG: Movement-Based Tabletop Role-Playing Game with Arm-Worn Devices and an Augmented Die. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17 Extended Abstracts)*. Association for Computing Machinery, New York, NY, USA, 639–646. https://doi.org/10.1145/3130859.3130868
[19] Oğuz Turan Buruk and Oğuzhan Özcan. 2018. Extracting Design Guidelines for Wearables and Movement in Tabletop Role-Playing Games via a Research Through Design Process. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3174087
[20] Hobbs Caroline. 2024. Designing Games for How We Learn.
[21] Tate Carson. 2019. Mesh Garden: A Creative-Based Musical Game for Participatory Musical Performance. (June 2019). https://doi.org/10.5281/ZENODO.3672986
[22] Karen Collins and Bill Kapralos. 2012. Beyond the Screen What We Can Learn about Game Design from Audio-Based Games. In *2nd Annual International Conference on Computer Games, Multimedia and Allied Technology (CGAT 2012)*. Global Science & Technology Forum (GSTF). https://doi.org/10.5176/2251-1679_CGAT06
[23] Luke Dahl, Jorge Herrera, and Carr Wilkerson. 2011. TweetDreams : Making Music with the Audience and the World Using Real-time Twitter Data. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. Zenodo, 272–275. https://doi.org/10.5281/zenodo.1177991
[24] Roger B. Dannenberg. 2021. Communication for Real-Time Music Systems: An Overview of O2. *Computer Music Journal* 45, 4 (Dec. 2021), 7–19. https://doi.org/10.1162/comj_a_00620
[25] Roger B. Dannenberg. 2022. Scalable and Easy-to-Use NIME Networking. In *International Conference on New Interfaces for Musical Expression*. https://doi.org/10.21428/92fbeb44.afe64be9
[26] Roy Thomas Fielding. 2000. *Architectural Styles and the Design of Network-Based Software Architectures.* University of California, Irvine.
[27] Johnny Friberg and Dan Gärdenfors. 2004. Audio Games: New Perspectives on Game Audio. In *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE '04)*. Association for Computing Machinery, New York, NY, USA, 148–154. https://doi.org/10.1145/1067343.1067361
[28] Alan Garner and Travis Elborough. 2006. *The Stone Book Quartet.* Harper Perennial, London.
[29] Benedict R. Gaster, Nathan Renney, Courtney Reed, Rodrigo Constanzo, Jasmine Butt, Maisie Palmer, Laddy Cadavid Hinojosa, Kathy Hinde, Sarah Angliss, Atau Tanaka, and Tom mitchell. 2026. radical interaction: a podcast of interaction exchange. In *NIME 2026*. London, UK.
[30] Jim Hollan and Scott Stornetta. 1992. Beyond Being There. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '92)*. Association for Computing Machinery, New York, NY, USA, 119–125. https://doi.org/10.1145/142750.142769
[31] Nick Hwang and Anthony T. Marasco. 2024. A Networking Framework and Mobile Node Controller for Sound Diffusion with Audience Participation. *Computer Music Journal* 48, 1 (March 2024), 51–64. https://doi.org/10.1162/comj_a_00712 arXiv:https://direct.mit.edu/comj/article-pdf/48/1/51/2512364/comj_a_00712.pdf
[32] Daniel Jones. 2023. AbletonOSC: A Unified Control API for Ableton Live. (May 2023). https://doi.org/10.5281/ZENODO.11189233
[33] Khushbu Kshirsagar, Marcelo Worsley, and Maryam Azmandian. 2025. StoryBit Boards: Crafting Interactive Narratives in Paper-Based Games with Micro:Bits. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY Companion '25)*. Association for Computing Machinery, New York, NY, USA, 302–304. https://doi.org/10.1145/3744736.3749319
[34] Samuel J. Leffler, William N. Joy, and Robert S. Fabry. 1983. *4.2bsd Networking Implementation Notes (Revised July, 1983).* Technical Report. University of California at Berkeley, USA.
[35] Thor Magnusson. 2010. Designing Constraints: Composing and Performing with Digital Musical Systems. *Computer Music Journal* 34, 4 (Dec. 2010), 62–73. https://doi.org/10.1162/COMJ_a_00026
[36] Anthony T Marasco, Nick Hwang, and Eric Sheffield. 2021. Collab-Hub: A system for creating collaborative telematic music performances with web-based instruments and creative development platforms. In *Proceedings of the International Web Audio Conference (WAC '21)*, Luis Joglar-Ongay, Xavier Serra, Frederic Font, Philip Tovstogan, Ariane Stolfi, Albin A. Correya, Antonio Ramires, Dmitry Bogdanov, Angel Faraldo, and Xavier Favory (Eds.). UPF, Barcelona, Spain.
[37] Raul Masu, Fabio Morreale, and Alexander Jensenius. 2023. The O in NIME: Reflecting on the Importance of Reusing and Repurposing Old Musical Instruments. In *New Interfaces for Musical Expression (NIME'23)*.
[38] Sam McGuire. 2014. *Modern MIDI: Sequencing and Performing Using Traditional and Mobile Tools.* Focal Press, Burlington, MA.
[39] Tace McNamara, Jon McCormack, and Maria Teresa Llano. 2025. Mixer Metaphors: Audio Interfaces for Non-Musical Applications. In *New Interfaces*

*for Musical Expression (NIME'25)*. https://doi.org/10.5281/ZENODO.15698879
[40] Andrew McPherson and Victor Zappi. 2015. An Environment for Submillisecond-Latency Audio and Sensor Processing on BeagleBone Black. In *Audio Engineering Society Convention 138*. Audio Engineering Society.
[41] Andrew P. McPherson, Robert H. Jack, and Giulio Moro. 2016. Action-Sound Latency: Are Our Tools Fast Enough?. In *New Interfaces for Musical Expression (NIME'16)*. Griffith University, Brisbane, Australia.
[42] MedeaTV. 2012. Kristina Andersen: Making Magic Machines or Practising at the Unknown.
[43] Matteo Meneghini. 2007. An Analysis of the Compositional Techniques in John Chowning's *Stria*. *Computer Music Journal* 31, 3 (Sept. 2007), 26–37. https://doi.org/10.1162/comj.2007.31.3.26
[44] Hwang Nick and Marasco RenAnthonyney. 2025. MoNoDeC: The Mobile Node Controller Platform. In *NIME 2025*. Zenodo. https://doi.org/10.5281/ZENODO.15698773
[45] Jonas Oxenbøll Petersen and Kim Halskov. 2025. Sound-O-Matic: A Tool for Designing and Implementing 3D Soundscapes. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference (DIS '25)*. Association for Computing Machinery, New York, NY, USA, 711–723. https://doi.org/10.1145/3715336.3735408
[46] Jon Peterson. 2024. *Playing at the World, 2E, Volume 1: The Invention of Dungeons & Dragons*. The MIT Press. https://doi.org/10.7551/mitpress/15384.001.0001
[47] Clara Rigaud. 2025. Zombitron: Towards a Toolbox for Repurposing Obsolete Smartphones into New Interactive Systems. In *LIMITS 2025: 11th Workshop on Computing within Limits*. online. https://doi.org/10.48550/ARXIV.2508.06354
[48] Nicole Robson, Andrew McPherson, and Nick Bryan-Kinns. 2022. Being With The Waves: An Ultrasonic Art Installation Enabling Rich Interaction Without Sensors. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. https://doi.org/10.21428/92fbeb44.376bc758
[49] Nicole Robson, Andrew McPherson, and Nick Bryan-Kinns. 2025. Listening Together-Apart: On the Social Mediation of Sound Installation Listening. In *Proceedings of the 20th International Audio Mostly Conference (Am '25)*. Association for Computing Machinery, New York, NY, USA, 164–177. https://doi.org/10.1145/3771594.3771610
[50] Nicole Robson, Andrew McPherson, and Nick Bryan-Kinns. 2025. Negotiating Entanglements in the Composition and Curation of an Ultrasonic Art Installation. (June 2025). https://doi.org/10.5281/ZENODO.15698992
[51] Ernie Roby-Tomić, Samuel Valkeapää, Ánde Somby, and Lars Ailo Bongo. 2025. Storycrafting With Constraints: Sámi Storytelling and Generative AI Workflows. In *Companion Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY Companion '25)*. Association for Computing Machinery, New York, NY, USA, 161–168. https://doi.org/10.1145/3744736.3749342
[52] Michael Salu. 2023. The Collective Storyteller. (March 2023).
[53] Norbert Schnell and Sébastien Robaszkiewicz. 2015. Soundworks – A Playground for Artists and Developers to Create Collaborative Mobile Web Performances. In *'Proceedings of the Web Audio Conference (WAC'15)*. Paris, France.
[54] Yann Seznec. 2022. Music Within Limits. In *2022 International Conference on ICT for Sustainability (ICT4S)*. 172–178. https://doi.org/10.1109/ICT4S55073.2022.00028
[55] Tim Shaw, Sébastien Piquemal, and John Bowers. 2015. Fields: An Exploration into the Use of Mobile Devices as a Medium for Sound Diffusion. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME 2015)*. The School of Music and the Center for Computation and Technology (CCT), Louisiana State University, Baton Rouge, Louisiana, USA, 281–284.
[56] Scott Smallwood. 2017. Locus Sono: A Listening Game For Nime. In *New Interfaces for Musical Expression (NIME'17)*. Zenodo. https://doi.org/10.5281/ZENODO.1176330
[57] Miriam Sturdee, Mafalda Gamboa, and Michael Heron. 2023. TTRPG UX: Requirements & Beyond. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems (CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, 1–9. https://doi.org/10.1145/3544549.3582737
[58] Kentaro Takahira, Yue Yu, Takanori Fujiwara, Ryo Suzuki, and Huamin Qu. 2025. InSituTale: Enhancing Augmented Data Storytelling with Physical Objects. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3746059.3747678
[59] Benjamin Taylor. 2017. A History Of The Audience As A Speaker Array. In *New Interfaces for Musical Expression (NIME'17)*. Zenodo. https://doi.org/10.5281/ZENODO.1176324
[60] Matthew Wright. 2005. Open Sound Control: An Enabling Technology for Musical Networking. *Organised Sound* 10, 3 (Dec. 2005), 193–200. https://doi.org/10.1017/S1355771805000932
[61] Matthew Wright, Adrian Freed, and Ali Momeni. 2003. Opensound Control: State Of The Art 2003. In *New Interfaces for Musical Expression (NIME'03)*. Zenodo. https://doi.org/10.5281/ZENODO.1176574