

# Networked Music: Bridging Real and Virtual Space

John P. Young

Peabody Conservatory of Music  
Johns Hopkins University  
1 East Mt. Vernon Place  
Baltimore, MD 21202 USA  
jpyoung@peabody.jhu.edu

This paper describes an exploration of utilizing the World Wide Web for interactive music. The origin of this investigation was the intermedia work *Telemusic #1*, by Randall Packer, which combined live performers with live public participation via the Web. During the event, visitors to the site navigated through a virtual interface, and while manipulating elements, projected their actions in the form of triggered sounds into the physical space. Simultaneously, the live audio performance was streamed back out to the Internet participants. Thus, anyone could take part in the collective realization of the work and hear the musical results in real time. The underlying technology is, to our knowledge, the first standards-based implementation linking the Web with Cycling '74's MAX. Using only ECMAScript/JavaScript, Java, and the OTUDP external from UC Berkeley CNMAT, virtually any conceivable interaction with a Web page can send data to a MAX patch for processing. The code can also be readily adapted to work with Pd, jMAX, and other network-enabled applications.

## 1. Introduction

*Telemusic #1* originated from a long-held artistic vision of Randall Packer. Through the process of close collaboration and harmonization of compositional goals with technical feasibility, it became realized as an interactive musical work incorporating live performers, signal processing, and real-time participation via a public Web site. Making this concept a reality required the extension and integration of existing technologies, in ways that had not previously been documented. One primary objective for the software development was to adhere to open, established standards as much as possible. This mandate led to the use of Java and JavaScript (standardized as ECMAScript) for the core components, and the OpenSoundControl (OSC) objects from the UC Berkeley Center for New Music and Audio Technologies (CNMAT) for interfacing with Cycling '74 MAX. The resulting system enables anyone interacting with a Web site to send data over the Internet to a computer running MAX, which can then process the input as it would from any other source. In a sense, one can transform the Web into a musical instrument performed by an unseen ensemble of limitless proportions.

## 2. Context

The eventual goal of this research will be to dissolve boundaries between real and virtual space, by creating a persistent, distributed interactive world of sound and imagery. As development advances, it will naturally intersect to an increasing degree with efforts in the pursuit of 'virtual reality'. A survey of the literature in this field reveals a strong emphasis on

representationalism—modeling the physical properties of our tangible world as realistically as possible. Much effort has been devoted to generating reasonable simulacra, in the expectation that immersion can best be achieved by recreating the familiar, with less interest demonstrated in constructing a compelling environment based on alternative principles. This may reflect a 'scientific' emphasis on analyzing and reproducing external features for judgement by comparison, as opposed to 'artistic' considerations of reinterpretation and internal consistency. Rather than focusing so much on anthropomorphic avatars, recognizable visual environments, and spatially flawless acoustic cues, perhaps more investigation of compositional issues is warranted. Surely there are other ways of delivering a compelling experience to a participant other than emulating his/her natural milieu.

Most graphical virtual worlds to date have been designed as isolated entities, to be experienced by one or a few physically collocated individuals. Groundbreaking work in this area that laid the foundation for extending these experiences to a shared context should be acknowledged. 'Sculpting 3D Worlds with Music' (Greuel 1996) and related efforts by Fakespace Music on rendering real-time visual analogues to sound informed the parallel development of the dynamic audio-driven visualizations one can enjoy in any recent music-playing software. These visualizations can be extraordinarily compelling in the collective experience of a dance club—a virtual equivalent can't be far behind. *Coney Island* (Bargar et al. 2000) was a milestone in integrating virtual scenery with responsiveness to audience participation. Taking a leap from this expensive demonstration of technology in

the direction of more populist counterparts might lead us to the current crop of Massively Multiplayer Online Role-Playing Games (MMORPGs). These are the culmination of well-established communal constructs born on the Internet in the form of Multi-User Dungeons (MUDs), and their related brethren. Originally text-based, the trajectory of these shared realities has been towards ever-greater explicit visual detail, to the extent of even scanning one's likeness into the game, molding an avatar as an unmistakable extension of oneself. There is every reason to believe that these trends will continue, defining the state-of-the-art in consensual virtual reality. These game environments, though still aspiring to traditional, albeit fantastic, notions of realism, encompass many useful paradigms for less ambitious endeavors:

- Bidirectional communication between each interactor and the virtual space
- Independent and persistent existence of the virtual realm
- Consistent, perceptible rules governing interaction and feedback
- Aspects of emergent behavior—reaction of the virtual world incorporates, but is not limited to, direct manipulation
- Potential for coordinated collaboration with other interactors without requiring external channels of communication
- Evolution—the shared environment changes as a result of the sum total of interactions

These conditions describe many aspects of our perception of physical reality, but need not be implemented as a literal reflection thereof, with all the complexity that would imply. Taking as a point of departure the observation that music is deeply meaningful though fundamentally abstract, the features above can potentially be incorporated into a virtual environment arising from the same conceptual basis as our relationship to music.

### 3. Inception

*Telemusic #1*, a collaborative intermedia work by Randall Packer with Steve Bradley and myself, was the driving force behind development of a distributed input environment functioning in parallel with a live performance. The physical aspects of the piece took place on Friday, November 3<sup>rd</sup>, 2000 at the Sonic Circuits VIII International Festival of Electronic Music and Art in St. Paul, MN, USA. A Web site, [www.telemusic.org](http://www.telemusic.org), was constructed for the event, presenting an interface for visitors worldwide to participate in real time. The primary controls were implemented in Macromedia Flash. Visitors to the site could define a 'telematic identity' and navigate through

a virtual space. By manipulating various elements, they projected their actions into the physical space in the form of triggered samples, filter parameters, and other live sonic results, according to Packer's compositional design. In addition, an active audio feed was streamed back to Web participants using RealAudio, providing a stereo rendering of the performance in progress. Thus it was possible to interact with the Web site and hear concrete audible feedback of one's actions in a matter of seconds. The Walker Art Center, one of the sponsors of *Telemusic #1*, also cooperated by invisibly re-coding their home page, mapping various links so that visitors to their Web site became unwitting participants in the piece. The resulting work was a dynamic balance of precomposed elements and improvisation, as the input received from the Internet was inherently unpredictable. By most accounts, it was a successful premiere, both technically and artistically.

### 4. Related projects

Previous attempts to interface standard Web technologies with interactive music have involved substantial limitations. *Variations for WWW* (Yamagishi and Setoh 1998), was a breakthrough for its time. However, it only worked with form-submitted input, substantially limiting the flexibility and reactivity of the user interface, and was based on David Zicarelli's W protocol and W server, which are no longer supported or available. JSyn (Burk 1998) is an excellent tool for producing interactive audio entirely within a local browser environment. This software was extended with TransJam (Burk 2000), a server-based module enabling communication between multiple JSyn users. However, using the system requires installation of the JSyn plugin on each client, the TransJam server component is not available to the public, both are closed-source projects, and they do not include any facility for communicating with other common software applications such as MAX. 'Piano Master Classes via the Internet' (Young 1999) introduced musically sophisticated error-correction for near-real-time collaboration over the Internet. However, the musical elements of the system were entirely based in MAX, and could not be integrated with a browser-based interface.

The research supporting *Telemusic #1* aimed to pick up where these predecessors left off, enabling a Web-based interactive interface to control the musical capabilities of MAX/MSP in either a client/server or peer-to-peer fashion, using sophisticated error correction for transmitting live control data over the Internet. In addition, full access to the relevant source code will be available under the GNU Public License (GPL), allowing freedom to manipulate the system as long as any modifications or enhancements are submitted back into the project for all to share.

Others have made forays into the territory of musically collaborative virtual space. Enterprises such as Rocket Network enable participants to essentially use central servers as a convenient conduit for sharing their material out of real time. Such services underscore the impossibility of simultaneous ‘jamming’ over the network in the traditional sense, without confronting the potential for manipulating audio in a time-insensitive context. *Cathedral* (De Ritis 1999) combines periodic live performances with Web-based interaction, but the experience is unidirectional—each participant inhabits their own bubble, with no ability to influence the performance as witnessed by others. Interestingly, the Virtual Worlds Group at Microsoft Research may have progressed the farthest in this direction with their prototype, MusicWorld:

a graphical collaborative musical performance environment where avatars join together to explore a live, on-line ‘album’ of multiple songs. Within each soundscape, avatars can mix sounds and compose their own musical patterns that are shared in real-time. Even though all of the avatars can affect the music individually, all the changes they make are instantly updated across the network, guaranteeing that the music is truly collaborative and heard the same way for all participants. (Microsoft Research 2000)

MusicWorld exhibits important characteristics of a distributed musical environment, but is limited by its emphasis on literal representations and incompatibility with non-Windows platforms.

All of these efforts illustrate worthwhile objectives, but none seem to aspire to the larger goal of creating a persistent virtual space where anyone can collaboratively interact within a mutually composed and experienced environment for exploring the interface between sound and image. There is no inherent obstacle to achieving such a goal. This paper represents a first step on that path.

## 5. Technical Description

The current software implementation is very flexible, allowing transmission of arbitrary data to any IP-addressable client in response to any JavaScript event. Common browser event triggers include page loading (onLoad), clicks (onClick), rollovers (onMouseOver), etc. *Telemusic #1* also made extensive use of the Macromedia Flash FSCommand(), a built-in method for addressing JavaScript that allows custom functions to be executed based on seven possible object behaviors in a Flash movie: Press, Release, Rollover, Rollout, DragOver, DragOut, and KeyPress. Any plug-in

similarly JavaScript-enabled could be successfully used with this system. Once caught by a JavaScript event handler, action data is passed to an embedded Java applet. The applet then forwards that data over the network via TCP/IP to a Java application running on the host Web server. The server application reformats the data into OSC so it can be received within MAX, at which point the data can be freely manipulated as control information. In its current form, this data flow is one-way, from browser to MAX only. It would not be difficult to send a response back from MAX to the client’s embedded Java applet, allowing bidirectional communication and greater interactive potential.

The data path may seem rather convoluted, but is necessitated by the intentional containment of browser capability for security reasons. Flash doesn’t include any means of external communication other than passing an FSCommand() to JavaScript. JavaScript has no networking functions of its own, so must call methods from a Java applet in order to connect with anything outside its window. Java applets are constrained by the well-known ‘sandbox’, which prohibits establishing a network connection with any machine but the one from which the applet was loaded—the host Web server. Once the data is received by the Java ‘listener’ application running on the Web server, it is finally free of restrictions and can be reformatted for any supported protocol and sent to any machine on the Internet. The only way to circumvent all these steps would be to create a browser plug-in specifically designed to violate the well-established browser security model in return for a modest increase in expediency.

One caveat with the software is that it is currently limited to using the OSC protocol developed at UC Berkeley CNMAT to communicate with MAX. Despite its moniker, OpenSoundControl is not entirely open. Much of the general specification has been published, but source code to the specific implementation used by the MAX external objects is not available, and substantial portions of the software fall under restrictive copyright of the UC Regents. However, OSC, and OTUDP, its companion object for UDP-based network communication, are the only networking externals currently supported for the MAX environment, and they function quite well in this capacity. OSC will continue to be supported, but interoperability will be extended to the open-source projects Pd by Miller Puckette and jMax from IRCAM as well, for those who wish to have complete end-to-end transparency in their software.

This system was designed for maximum cross-platform compatibility. It has been successfully tested using Netscape Navigator under Windows, MacOS, and Linux, and Microsoft Internet Explorer (IE) under

Windows. IE under MacOS does not work because IE:Mac unfortunately does not enable any communication between JavaScript and an embedded Java applet. The server application has been successfully tested under Windows, Linux, and MacOS X. Because most of the software is written in Java, it is quite portable, and every effort will be made to ensure full compatibility with common operating systems and browsers.

## 6. Future Development

My collaboration with Randall Packer continues in the form of his next intermedia work, *Telemusic #2*. This piece will build technologically on its predecessor by interpreting network traffic statistics gathered in real time to generate an organically textured musical atmosphere as a canvas for interaction. The software required will likely be based on two open-source projects: the Multi Router Traffic Grapher (Oetiker 2000), a tool for aggregating Simple Network Management Protocol (SNMP) queries, and Peep (Gilfix and Couch 2000), a framework for auralizing

dynamic network conditions. Looking beyond the horizon, Packer sees *Telemusic #3* adding musically responsive visualizations to the environment to create a more immersive intermedia experience.

## 7. Acknowledgements

This research owes a great debt of gratitude to my friend and collaborator Randall Packer, whose wonderfully impractical ideas pushed me beyond the limits I would have otherwise set for myself.

## 8. Conclusion

The system described here has been designed for flexibility and extensibility, offering a framework which can support many potential applications in addition to those already noted. I sincerely hope that development and sharing of this software can inspire further creative use of the Web as a means of participating in live interactive works. The latest version and documentation can be found at <http://www.netmuse.org>.

## References

- Burk, P. 1998. JSyn – a real-time synthesis API for Java. *Proceedings of the International Computer Music Conference*. pp. 252-5. Ann Arbor, MI: ICMA.
- . 2000. Jammin' on the web – a new client/server architecture for multi-user musical performance. *Proceedings of the International Computer Music Conference*. pp. 117-20. Berlin, Germany: ICMA.
- Bargar, R., F. Dechelle, I. Choi, A. Betts, C. Goudeseune, N. Schnell, and O. Warusfel. 2000. *Coney Island: Combining jMax, Spat and VSS for acoustic integration of spatial and temporal models in a virtual reality installation*. *Proceedings of the International Computer Music Conference*. pp. 149-53. Berlin, Germany: ICMA.
- De Ritis, A. 1999. *Cathedral: an interactive work for the Web*. *Proceedings of the International Computer Music Conference*. pp. 224-7. Beijing, China: ICMA.
- Gilfix, M., and A. Couch. 2000. Peep (the network auralizer): monitoring your network with sound. *Proceedings of the 14<sup>th</sup> Systems Administration Conference*. New Orleans, LA: USENIX.
- Greuel, C., M. Bolas, N. Bolas, and J. McDowall. Sculpting 3D worlds with music; advanced texturing techniques. *Proceedings of the SPIE—The International Society for Optical Engineering* 2653: 306-15. San Jose, CA: SPIE.
- Microsoft Research. 2000. MusicWorld. <<http://www.vworlds.org/music/>>. 1 May 2001.
- Oetiker, T. 2000. MRTG—the multi router traffic grapher. <<http://ee-staff.ethz.ch/~oetiker/webtools/mrtg/paper/>>. 1 May 2001.
- Yamagishi, S., and K. Setoh. 1998. *Variations for WWW: network music by MAX and the WWW*. *Proceedings of the International Computer Music Conference*. pp. 510-13. Ann Arbor, MI: ICMA.
- Young, J., and I. Fujinaga. 1999. Piano master classes via the Internet. *Proceedings of the International Computer Music Conference*. pp. 135-7. Beijing, China: ICMA.