DYNAMIC PERFORMANCE SPACES FOR THEATRE PRODUCTION

by Jeff Burke

Published in *TD&T*, Vol. 38 No. 1 (Winter 2002)

Theatre Design & Technology, the journal for design and production professionals in the performing arts and entertainment industry, is published four times a year by United States Institute for Theatre Technology. For information about joining USITT or to purchase back issues of *TD&T*, please contact the USITT office:

USITT
6443 Ridings Rd. Ste 134
Syracuse, NY 13206
tel: 800-93-USITT (800-938-7488)
tel: 315-463-6463
fax: 315-463-6525
e-mail: info@office.usitt.org
web: www.usitt.org

Copyright 2002 United States Institute for Theatre Technology, Inc.
Advances in theatre technology provide amazing capabilities for both expressive subtlety and sophisticated spectacle. Often however, the more intricate the technical sequence, the less it can be adjusted during performance and the more mechanically consistent the actors' performances must become.

With more complex, media-rich stage design, it is difficult to maintain the control flexibility possible in simpler, manually operated sound and lighting systems. Though computer-controlled equipment can handle many hundreds of static cues and perform incredibly complex cross-fades between them, they usually do so according to pre-determined timing. Projection, live video, and motorized scenery tend to use even more rigid control structures; in the development of more complex theatre systems, repeatability and rehearsal editing features have taken precedence over flexible run-time control.

It is often difficult or impossible for an operator or designer to safely “reach into” a cue sequence during a performance and adjust cue parameters while it is active. The more complex the sequence, the less likely that this is possible. Of course onstage action still drives the progression through most of the lighting and sound design. When performers reach certain points in the script or in their movement, cues (usually with fixed timing) are started by the stage manager that adjust lighting, sound, or other stage elements according to a predetermined design. More subtly, if the performers’ pacing seems too slow or too fast, the stage manager might call these cues at a different tempo to try to bring the show back to the desired pace.

In these situations, digital technology is used to repetitively (and accurately) perform pre-determined tasks that have broad or subtle effects on the stage environment. This ignores the ability of digital technology to act as a common language, for digital technology to be a sophisticated bridge between many dynamically varying arenas: the physical world...
...theatre artists can choose to set up explicit systems of relationships between the physical world (as it can be measured by technology) and digitally controlled design elements.

of the space, the action of the actors, design elements, even connections and virtual spaces on the Internet—anything that can be measured, expressed, or controlled digitally. This bridge is present in traditional theatre technology, but simply connects the operator’s finger on a button to a pre-pre-programmed effect in the space. A broader approach is possible and brings both rich new tools and whole new processes for performance. The example explored here uses digital technology to connect sensing systems (e.g., a performer position tracking environment) to the extensive technical infrastructure of modern theatres, enabling designers to create direct real-time relationships between onstage action and the many parameters of digitally controlled design elements. A pragmatic benefit of such a system is that actors might not need to “hit their mark” exactly or cross the stage at the right speed in order to be properly lit. Simple modifications to lighting based on performer position could make complex cues more flexible and responsive to performers. Rather than estimating the time it takes a performer to cross the stage in rehearsal and programming a fade time based on that estimate, instruments can be dimmed directly in performance based on the actor’s position.

Once a system exists to gather performance data and control production equipment based on it, more sophisticated and unconventional relationships between performers and the stage environment become possible. For example, sound could be intensified based on the speed of movement of an actor during a certain section or lighting controlled by the distance from one tracked performer to another, regardless of where they are on stage.

Clearly, not all desired relationships between action and design can be expressed spatially, nor should they all be one-to-one, from the “actor steps here, light turns on” / “actor steps away, light fades out” genre of control. This is merely a practical starting point for experimentation. The deep shift in paradigm is that, in addition to designing traditional cues, theatre artists can choose to set up explicit systems of relationships between the physical world (as it can be measured by technology) and digitally controlled design elements.

Our research at the HyperMedia Studio in the UCLA School of Theater, Film and Television addresses the technological question of how to enable these relationships and the aesthetic considerations arising from their application to the stage. This involves exciting and quite challenging collaborations between theatre practitioners, engineers, and computer scientists. This paper describes our recent research in one particular area: systems that allow designers to experiment with connections between performer action and design in a large production setting. It does not include other very interesting possibilities for audience interaction, networked performances, database interfaces, and so on. Though digital technology enables new ways of escaping traditional forms, it also allows new types of design within those familiar boundaries—and this is what will be discussed here.

Musicians, dancers, and choreographers were some of the first to explore “interactive technology” that connected performers (and often viewer-participants) with digitally controlled media in their environment. They have driven technical and aesthetic growth in digital art for many years, using new technology in “traditional” performances, creating new genres of performance using digital media, and offering critical comment on technology in their art and writings. Performance artists have likewise embraced and criticized technology in their work. Though some experimental theatre has incorporated similar systems, they are rarely found in “traditional” theatre.

In the United States, Mark Reaney’s work at the University of Kansas (Reaney 1999) and David Saltz’s at the University of Georgia (Saltz 2001) provide examples of larger productions using digital technology as more than just a new source of projection. These are fairly unique; theatre in its most familiar form, with a group of actors, a text, lights, sound, perhaps projection and scenery, and a defined playing and audience space, has remained largely on the periphery of new cross-disciplinary art arising out of “the digital.”

This research is related to other interactive stage and story environment work already taking place, with an important distinction. It does not attempt to produce stage systems that understand what the actors are doing, follow a script based on their movement and speech, or provide improvisational media response to human action. In this case, we accept simpler capabilities in return for reliability and very few limitations on production design. As with the work of Saltz and Lovell (Lovell and Mitchell 1995), it depends on human directors and designers to author relationships between onstage action and design elements.
Figure 1. Block diagram of interactive system for Macbett.

Not shown:
Support services running on production machines, backup equipment, hardware interfaces for external communication.
For a recent departmental production of Eugene Ionesco’s *Macbet*, we developed a complete system to allow control of theatrical lighting and sound based on performer position and movement. Our primary goal was to create a stable environment for experimentation, in the context of a real performance with real deadlines, that addressed the interests described above. Additionally, some technical goals guided the development:

- Traditional design tools, theatre systems, and protocols were to be accommodated wherever possible.
- Any systems created would communicate using TCP/IP over standard Ethernet networks.
- There was to be no central “brain” of the show—one machine upon which everything depended.

The last point represents a departure from traditional show control approaches. Different “data sources” provided performance information over the network to controllers for lighting and sound, but it was up to those individual controllers to request the information that they needed and decide how to use it.

A block diagram of the system is given in figure 1. The individual computers were connected to a switched fast Ethernet local area network (LAN) and used a simple publish/subscribe mechanism to communicate. A “broker” server on the network managed a list of all available types of data (position, velocity, etc.) and the addresses of the servers providing them. Individual controllers looked up (through the broker) the server for the data they needed and requested a “subscription” from that server, which then provided a continuous stream of the appropriate information, e.g. the position of a particular actor, until the controller unsubscribed. The following sections describe the individual components of the system in more detail.

**TECHNOLOGY**

**TRACKING SYSTEM and NETWORK POSITION SERVER**

The Martin Lighting Director (MLD), a commercial performer tracking system developed by Acoustic Positioning Research and marketed by Martin Professional was used to track the position of performers. The MLD uses four ultrasonic speakers mounted above the stage that pulse at different frequencies (figure 2). These pulses are received by modified wireless microphones worn by each performer (figure 3) and sent back to the base station over a wireless radio-frequency link. The time of flight of each pulse to the performer's microphone is used to triangulate its position on stage. We developed a network position server in Java that receives continuous updates of performer coordinates and instantaneous velocity from the MLD computer. That information is then served to clients (either in rectangular or spherical coordinates) using the publish/subscribe method described above.

The position server also provided other types of data based directly on performer position. For example, a common type of data used by many clients, including the lighting and sound control systems, were relative distances between a performer and a few common stage points, as well as between pairs of performers. The position server implemented these functions itself, so these often-used values are only calculated once throughout the entire network of controllers.

Other intelligent stage systems rely on computer vision or “video tracking” techniques that do not perform well under the drastic lighting shifts typical in live performance, preventing (for now) their extension into traditional theatre. The ultrasonic performer tracking system used here requires no such lighting restrictions. It also provides consistent identifi-
ocation of performers regardless of their costume and makeup, which is difficult or impossible to achieve with current computer vision technology; radar, or in-floor pressure sensors. The major disadvantage is that performers must wear a wireless microphone with an element that cannot be covered or obscured. Tracking human movement as a point in space is also limiting; it provides no gesture information like that available from computer vision systems.

REGION MAPPING SERVER

In many situations, exact performer coordinates were unimportant. Instead, the designers or director simply wanted to use the actor’s presence in (or absence from) a particular region onstage. For example, in a particular scene, one might intuitively define the regions “upstage left,” “centerstage,” and “near the guillotine.” A simple way to visualize these areas would be to shade them on a groundplan. That sketch might then be translated by hand to a region expressed numerically in a digital controller. An interactive relationship would link a lighting or sound shift to a performer’s presence in the region, e.g., “if Banco moves ‘near the guillotine,’ raise the lights on it by ten percent.” We wanted to eliminate the intermediate step of hand translation, using a computer graphic sketch to define the region directly.

The MLD software itself has some facility for drawing shapes on a groundplan that trigger events or adjust lighting output when the performer moves into or out of them. A major limitation of this region-based control is the “hard” edge of each region. If the tracked performer is on the boundary of a region, their movement and sensor jitter can cause the coordinates to bounce in and out of the region, triggering responses from the production control systems. This was one motivation for our choice to allow regions with “soft” edges. In this case, a performer might be in the regions “center” and “downstage,” but if she moves a little to the left, she would still be “center” but less so, and would remain the same amount “downstage.”

A region mapping tool was developed to allow the use of graphical sketches on a groundplan for control of production equipment (figure 4). The tool loads a user-created Adobe Photoshop image containing sketches of each region, one per layer. It separates and stores the image’s individual layers in memory, then subscribes to position data from the network position server and maps the tracking coordinate system onto the image. The server maps each new position sample (for every performer) into the image, and the opacity of the corresponding image pixel in every layer is made available over the network to other clients. Clients can use these “presence” values in any manner. For example, the lighting system described later might dynamically map high opacity values to brightness, smoothly fading up an instrument’s intensity as a performer moves “through” a gradient of decreasing transparency in the Photoshop image. Different mappings can be loaded on-the-fly by switching the Photoshop image with a network command.

INTERPRETED POSITION SERVER

Additionally, an interpreted position server maintained several short-term snapshots of each tracker’s movement and used them to calculate statistics providing some insight into how the performer or prop was moving. These statistics ranged from the familiar—average speed, distance traveled, average direction—to more interesting quantities like the “roundness,” “directness,” and “stillness” of movement. These were continuously calculated for each performer and then made available for subscription on the network. The server subscribes to position data for every tracker, generating statistics for each performer over a number of different time intervals ranging from forty milliseconds to two seconds.

DYNAMIC SOUND AND LIGHTING CONTROL

We wanted to experiment with control over specific details of the cues themselves—lighting and sound intensity, transition speed, choice of media, effect parameters, etc. This required controllers that could relate sensor data to production system outputs in real-time. A balance between performance, runtime configurability, and ease of use was desired. The pragmatic goal of having practical lighting and sound control ready for the production was also very important and led to two different approaches for lighting and sound. The dynamic lighting system (DLS) software was developed from scratch to allow relationship-based control of DMX lighting through a simple scripting language. This approach provided the maximum amount of flexibility but required a considerable development effort and resulted in a relatively unrefined user interface.

Rather than defining static looks, cues in the DLS define relationships between sensor data and lighting control parameters using user-scripted expressions. The controllable parameters are familiar and include intensity, fade in/out times, and wait in/out times, but these values can be updated on-the-fly by evaluating user-defined formulas dependent on performance data. Additionally, the dynamic system intercepts the DMX output of the traditional console, allowing it to modify as well as override existing stage looks based on performer movement data. For example, it can lower the intensity of every light in a cue output from the traditional console by five to fifteen percent depending on how far an actor is from downstage. This also allowed the traditional console (an ETC Obsession II) to be used normally during the show, its control over the house dimmer system remaining intact for any cues that were not designed to be affected by real-time performance data.

In contrast to lighting, projection, and scenery, a commercial package for the real-time playback and processing of live and recorded sound already exists. Max/MSP is a graphical programming language for the Macintosh with a robust toolkit for real-time audio signal processing. Sound for Macbeth was designed and implemented in Max by David Beaudry, a doctoral student in the Music Department at UCLA. An interface to the different performance data servers available on the network was written for Max, allowing data from
the sensing, interpretation, and mapping systems to be used for control over sound playback, volume, reverb, and any other available processing tools. Max can play back (and record) digital audio files and address up to 512 discrete outputs using commodity hardware, making multichannel sound output very straightforward. Beaudry used one Macintosh running Max for all “traditional” cues and another for all dynamic cues that were dependent on performance data.

**PRODUCTION IMPLEMENTATION**

The UCLA Department of Theater’s subscription-series production of Eugene Ionesco’s *Macbett* in the Spring of 2001 was the first artistic piece to use these systems in performance. *Macbett* was produced in a largely traditional manner, following the production process typical of large shows at UCLA. It was directed and designed by graduate students, advised by faculty, managed by department staff, with undergraduate students forming the cast and crew. Like other efforts at the HyperMedia Studio, it also involved the collaboration of students from computer science and electrical engineering, who helped to develop the “interactive systems” concurrently with the production process (figure 5).

For the production of *Macbett*, a total of five performers were equipped with position trackers; another two were placed in staffs carried by the witches. A network of four workstations provided performance data services for the show. One

![Figure 5. A computer science student works in an early test environment on the stage of UCLA’s Freud Playhouse.](image-url)
managed the MLD itself, while the others ran the software described in previous sections: the network position server, interpreted position server, region mapping server, and various support servers. Two additional machines were used for sound control: one provided “traditional” playback, while the other handled “interactive” cues. A final computer ran the dynamic lighting system.

In cases where the “risk” of allowing performers to affect cues is acceptable, the designer can establish relationships directly rather than approximating them with pre-timed crossfades.

**EXAMPLE INTERACTIVE SEQUENCES**

Many of the cues in *Macbett* were executed as traditional timed cues. Early in the conceptual development of the show, the director decided that certain performers would be given control over the theatre space in scenes where their characters were in some way “supernatural.” For example, Ionesco’s two witches influence the environment whenever they appear. In most cases, this control was tied to a character and, if that character appeared more than once, he or she maintained similar control over the environment in each appearance.

A classic insertion of the absurd by Ionesco, the Butterfly Hunter, was to move through a pastoral sound environment that becomes progressively more detailed as he chases his invisible quarry onto the stage. To accomplish this effect, sound designer David Beaudry used the region mapping tool to draw smooth position-based fades of overlapping audio tracks. Different layers in the image controlled lighting effects in the same scene, fading in new lighting instruments as the actor moved into certain positions. The performer learned the relationship between his position and the design and could be confident that, as he responded to each night’s audience and performance, the design would be consistent in its relation to his action.

When the character Macol appears in the final act, a single dynamic cue lights his initial dialogue from a platform at one edge of the stage through his cross to the center to kill Macbett. The stage manager starts the cue when Macol appears on the platform and the operator’s “go” on the ETC Obsession II simultaneously triggers both the console’s own cue and the dynamic system. The console establishes the scene and platform light for his initial dialogue. When he finally leaves the platform, the dynamic system takes over, opening a path of light before him with each step, while capturing and dimming the rest of the console’s cue as he moves inward. By the time Macol is in the center, he is alone with Macbett in downlight brought in based on his position. A set of simultaneously active dynamic cues handle this sequence based on Macol’s distance from centerstage, matching his timing every night. After Macbett is killed, the operator takes the next light cue on the console, fading out dynamic control and bringing in the next cue.

These sequences can be achieved (or at least approximated) without interactivity. Yet connecting elements of the stage design directly to performers—mapping position or movement to the digital parameters of certain cues—eliminates a layer of mediation between action and design. In cases where the “risk” of allowing performers to affect cues is acceptable, the designer can establish relationships directly rather than approximating them with pre-timed crossfades. This has fascinating implications for theatre and location-based entertainment.

By necessity, spatial relationships defined most of our “dynamic cues,” but new sensing technologies will someday unobtrusively follow human gesture and speech as easily as we tracked position and movement. These will provide information on how performers move and speak, not just where they are.

Using the interpreted position server, this potential could be explored even in *Macbett*. The clearest example is found in the primary agents of the supernatural, the two witches, who also appear as Lady Macbett and her Lady-in-Waiting. Each witch was to have her own type of control over the environment through her staff (figure 6). The first conjured thunder and lighting by raising the staff quickly in the air—the quicker and stronger the thrust, the more powerful the lightning strike—while the second witch swirled her staff to create ripples of darkness, color shifts, and the sound of whirring wind proportional to the speed of her staff.

In addition to being based on how the props moved and not where they were, these cues had to remain “dormant” when the witches moved normally with their staffs. A tracker was embedded in the foam head created for each of the staffs by costume designer Ivan Marquez and a combination of interpreted position information for both the staff and the witch herself was necessary to achieve the effects. Lightning was created by controlling strobe and intensity attributes of automated lighting fixtures while triggering a proportionally amplified sound of thunder in Max. Color shifts combined with slightly randomized intensity control achieved the lighting effect for the second witch. At the same time, the whirling of her staff also faded in and panned the sound of the wind. These relationships were activated at the beginning of each scene where the witches appeared, allowing the actresses to conjure them up at any point.

These cues required the performers to be aware of their new capabilities on stage and to work with the director to explore how they could be most effectively used. Though the cues were designed and tested before rehearsal, the actors required time in the performance space with active influence over lighting and sound to discover how they could work within the newly responsive environment.
Even with this technology, some brushstrokes still happen by accident. And sometimes, accidents show the technology’s potential most clearly. An unexpected embellishment (a bug that we made no attempt to eliminate) was the tiny wisp of wind that accompanied certain small gestures of the staff by the actress portraying the second witch. As she learned to punctuate her speech with these gestures in a way that could not be done with traditional cues, a real interplay emerged between how she moved and her control over the environment. Her process would not have been the same if there had been an operator trying to follow every move. Instead, she discovered an unmediated relationship with the world of the play that she could use in her performance.

Her discovery, however small, is the beginning of an answer to those who argue that we are wasting our time to use new technology for results that, in many cases, can appear the same to an audience when achieved in a more traditional manner. Because of a new relationship between her character and the onstage world, her process and her performance changed, affecting the experience of the ensemble and of the audience.

CONCLUSION

The defining characteristic of the digital era is the potential that it brings for “real-time” interconnection between anything that can be measured, expressed, or controlled digitally. The systems described here illustrate only one type of interconnection: a direct interactive relationship between the performer and the onstage world. There are many others to be explored in performance. As new sensing technologies develop, the speech and gesture of the performer (or the audience member) can become “inputs” that are interconnected with traditional outputs, such as lighting, sound, video projection, and moving scenery. Or they can feed into a “memory,” perhaps implemented as some sort of database, to be recalled later on and used to adapt another performance to what happened before.

These new technologies, like any others, remain a tool. They should be used when appropriate and discarded when unnecessary. But their sole capability is not the automation of existing forms of stage control. Digital technology opens up new possibilities for design and performance process driven by a “systems aesthetic,” one the encourages us to explore dynamic relationships between each element of the world onstage.

Jeff Burke is a research fellow at the HyperMedia Studio in the UCLA School of Theater, Film and Television. His research interests are in the application of new technology to the performing arts and division by zero. He can be reached at jburke@hypermedia.ucla.edu.

TECHNICAL NOTES

Network communication used a custom UDP protocol over switched fast Ethernet. The network position server was written in Java 1.2.2 with some platform-specific objects. The region mapping server, dynamic lighting system, show location server (not described in this paper), and interpreted position server were written as multi-threaded Win32 applications in Visual C++ 6.0 using /n software’s IPWorks toolkit. The
broker and debugging interfaces were written in Visual Basic. Expression evaluation in the dynamic lighting server used the UCalt FastMath library; the server also used an Artistic Licence DMX-Dongle II for DMX input and output and a MIDIator MIDI to serial interface. The software was run on a collection of Pentium II single and multiprocessor systems running Windows NT or Windows 2000. The Max/MSP sound system used MOTU 2408 multi-channel interfaces.

ACKNOWLEDGEMENTS

Professors Fabian Wagmister and John Villasenor provided invaluable advice and assistance. Will Bauer of Acoustic Positioning Research helped us at every step of the way. Thanks to the Macbett production team, especially director Adam Shive, David Beaudry, David Miller, Ivan Marquez, Maiko Nezu, Michelle Magaldi, Katie Mitchell, Ben Story, and Gavin Haag, and the theatre staff, especially Jeff Wachtel, Jonathan Burke, and Jeff Segal. Undergraduate students involved in software development were Stephen Lin, John Cheung, A. J. Ross, and Genesan Kim. Partial funding was from the University of California Digital Media Innovations program.

SOURCES CITED


END NOTES

1. Show control systems have long used safety switches and other sensors to control cue timing. The approach suggested here is somewhat broader; it advocates using sensor information to control any cue parameter that the designer wishes to link to action. Two books are recommended for their technical overviews of modern theatre control and effects systems: John Huntington’s Control Systems for Live Entertainment, Second Edition and Graham Walne’s Effects for the Theatre.
2. The MLD can provide “soft” edges but not directly from the drawing tool for the regions. It actually has an extensive set of features for drawing mappings between position data and lighting output. The region mapping server described here is an effort to provide a network-accessible tool that integrates “softness” into the region creation step itself.
3. The transparency, or “alpha channel” of the image, is then used to designate soft edges in the Photoshop image.