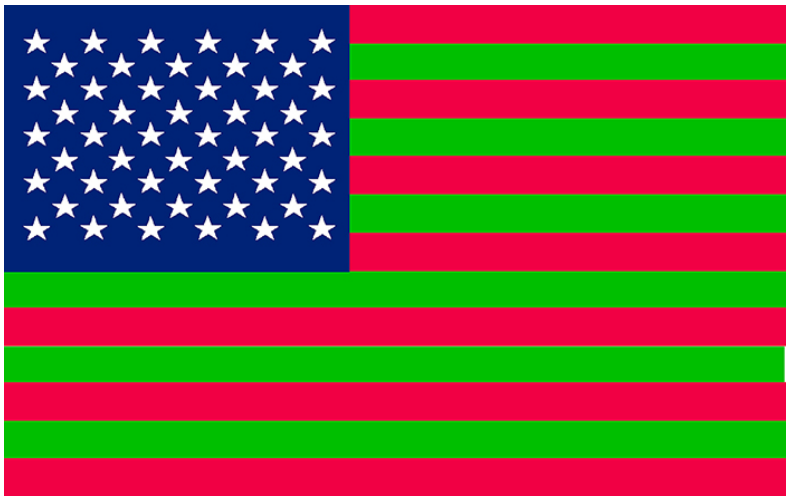
Jennifer, Noémie. “The Race to Save Computer-Based Art | Conservation Lab,” *The Creators Project*, September 6, 2016

The Creators Project

The Race to Save Computer-Based Art | Conservation Lab

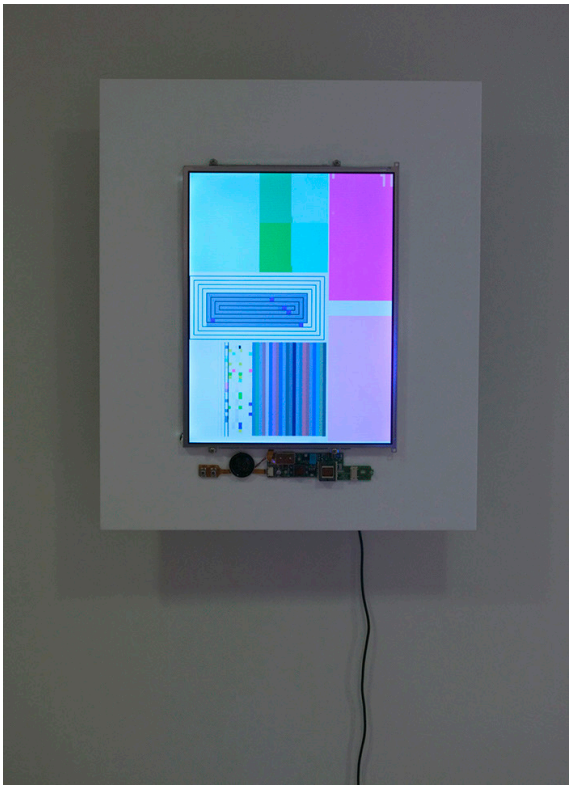
Noémie Jennifer — Sep 6



Viewing Java-based net artworks in modern browsers is becoming harder and harder. Pictured: Mark Napier, net.flag, 2002. Interactive networked code (Java applet with server database). Solomon R. Guggenheim Museum, New York 2002.17. © Mark Napier

Some of the most endangered works of art in the Guggenheim’s collection are also some of the youngest. The reason? They rely on computers—which is to say, they rely on the unreliable. Fitting those temperamental, short-lived machines within the museum framework is a difficult marriage, and conservators are left to mediate the conflicts. At a conference last year, Joanna Phillips, the Guggenheim’s conservator of time-based media, explained that the care of software-based work is particularly challenging because guidelines haven’t yet been clearly formulated, but are desperately needed. “These works are aging very rapidly. Intervention is urgent in many cases,” she warns. It’s a race without a roadmap.

In the hopes of fast-tracking technical research into the museum’s 22 computer-based works—and using them as case studies to establish best practices for the field—the Guggenheim has just secured funding for a two-year fellowship dedicated to the Conserving Computer-Based Art (CCBA) initiative. But who will be best for the job? “Should it be a conservator with some knowledge of coding, or a computer scientist with enough understanding of art and conservation?” wonders Phillips during a phone conversation with The Creators Project. “There is no precedent. We’re creating a new job description.”



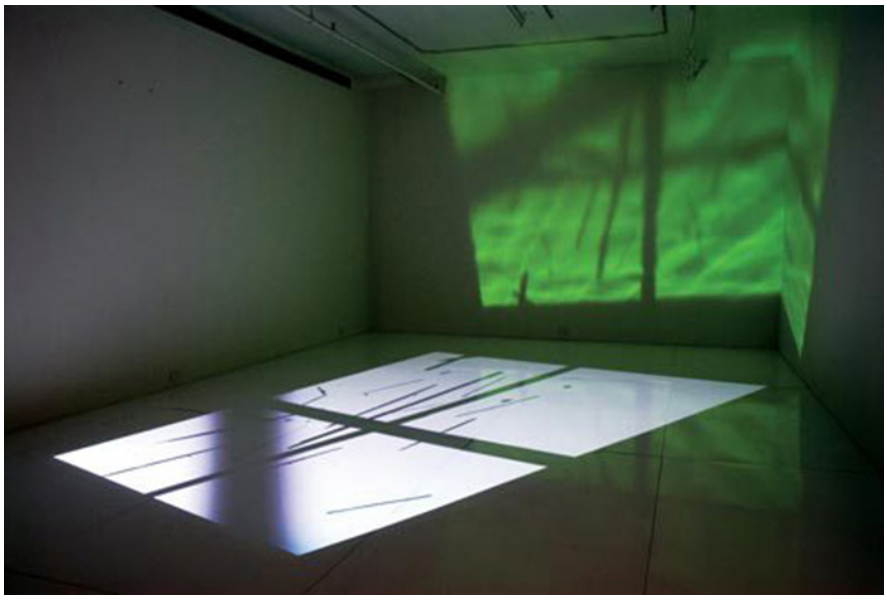
John F. Simon, Jr., Color Panel v1.0, 1999. Apple Powerbook 280C, software, and acrylic plastic, 13 1/2 x 10 1/2 x 2 1/2 in. Installation view: Seeing Double: Emulation in Theory and Practice, Solomon R. Guggenheim Museum, New York, March 19–May 16, 2004. Photo: David Heald © Solomon R. Guggenheim Museum

New, experimental, and interdisciplinary: Those are the three buzzwords of the CCBA initiative. Ambitious could be another: “We need to conduct a survey of all the works, create backups, identify which are high- and low-risk, and really launch an in-depth research and development phase,” notes Phillips. In this case, R&D will involve close examination of key works in their native environments, with their original hardware, and careful documentation of their “behaviors” through video recordings and detailed documentation (eg. the red triangle hovers over the blue square for 10 seconds, then jumps to the top right corner).

NYU computer science student Jiwon Shin during her summer internship with the Guggenheim Conservation Department. Photo: Joanna Phillips © Solomon R. Guggenheim Museum

Another critical step is to analyze the source code. “It’s simply beyond our human capacity to observe open-ended durations or detect whether an image sequence is randomized or programmed,” explains

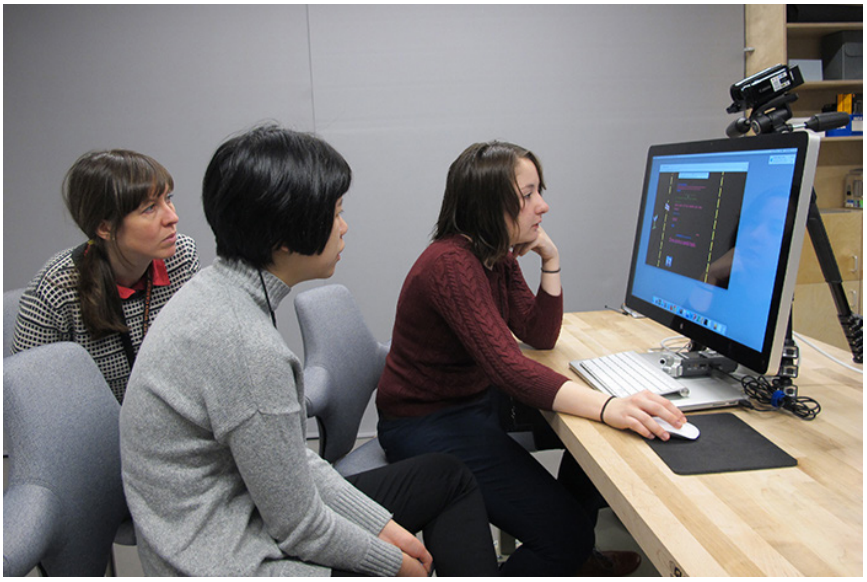
Phillips in her conference talk. “Code analysis reveals that information, providing that you’re fluent in the programming language.” Cue the arrival of the computer scientists: Since 2014, while preparing for CCBA initiative, the Guggenheim has partnered with NYU’s Prof. Deena Engel of Courant Institute’s Dept. of Computer Science and worked with CS students who are interested in art and old programming languages. Line by line, the students translate the code into information that is meaningful to conservators and curators.



Paul Chan, 6th Light, 2007. Flash animation projection, silent, 14 min, dimensions variable. Solomon R. Guggenheim Museum, New York. Courtesy the artist and Greene Naftali, New York © Paul Chan

“We’re trying to build a bridge between computer science analysis and conservation assessment,” Phillips tells us, adding that the collaboration required a learning curve on both ends. “We realized we have to make them understand what our questions are first. In turn, we try to understand things they find that we hadn’t even thought about.”

Computer scientists can tell conservators, for example, whether the speed of a moving object is coded into the work, or if it’s dependent on the processing speed of the computer. When the file for Paul Chan’s Flash animation 6th Light, which originally ran on a 2006 gaming computer, was tested on a contemporary PC, the 14-minute piece was over in 10 minutes. Since the original computer has now broken down, the museum is “exploring different possibilities of migrating the Flash animation to a new environment without affecting the artist-intended playback speed. Considerations include emulating the 2006 computer with its specific processing speed, and more invasive methods, such as adjusting the frame rate of the animation to compensate for contemporary processing speeds.” They can then assess which test environment most closely reproduces the behaviors of the work in its native environment—which is why that earlier documentation is so crucial. Performing those trials will surely require the help of programmers, but ultimately, the collection caretakers in charge, such as conservators and curators (in coordination with the artists, when available), are the only ones who can make decisions about what can change, taking into consideration all of the rules and ethics of the field. “That cannot be outsourced,” remarks Phillips.

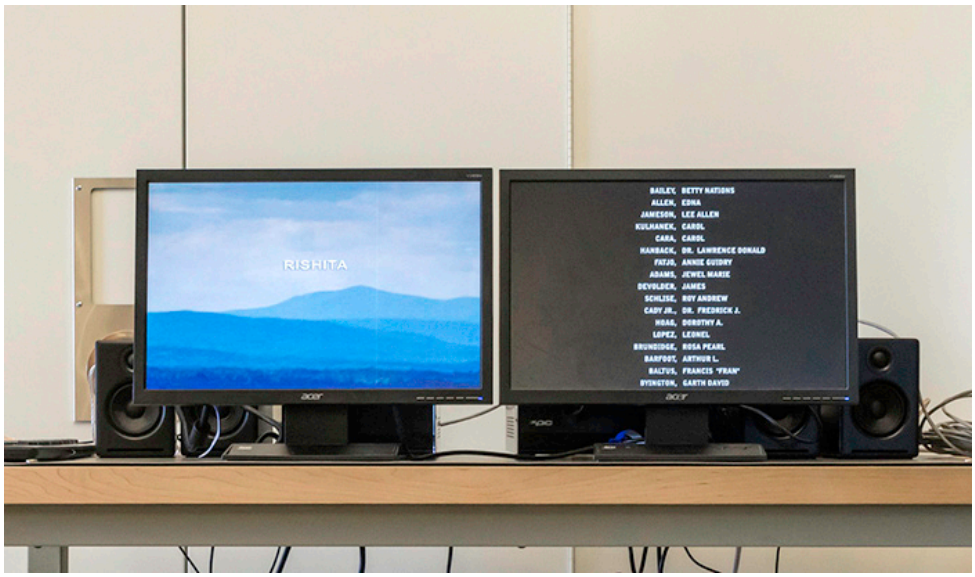


Recording of a narrated screen navigation of Shue Lea Cheang's net artwork, Brandon, (1998-99) in the Guggenheim conservation lab. Guggenheim conservator Joanna Phillips (left) invites NYU computer science students Jillian Zhong (middle) and Emma Dickson (right) to narrate through the screen navigation in order to capture the complex interactive functions of the work, its overall site structure and any compromised elements such as broken links or blocked Java applets. Photo: Brian Castriota © Solomon R. Guggenheim Museum

In many cases, it seems that the preservation of these works will likely involve emulation of outdated technology and migration onto newer platforms. That simply cannot be done, however, with Jason Rhoades' Sepia Movie or John F. Simon Jr.'s Color Panel, wherein the hardware is a unique, structural component of the work. "The museum may choose to create 'exhibition copies' that are identified as such in an exhibition context and approved by the artist, but we can't recreate the work itself, if the dedicated custom hardware fails. Ethically, it's not possible," notes the Guggenheim conservator. Works that are wholly dependent on hardware, then, seem to be particularly at-risk.

Yet even when software dependency is the only restriction, further ethical quandaries can easily arise: While Siebren Versteeg's Untitled Film II, which he created in Macromedia Director with Lingo, could probably be recreated with a contemporary language that preserves the work's look and feel, "that would be eliminating the trace of the artist's hand," cautions Phillips. Versteeg has explained that Lingo was the first language he ever used, and he stuck by it in the creation of several pieces. By the time he made the work in 2006, it was already obsolete, but it remained his preferred mode of expression. Phillips, who was originally trained as a paintings conservator, likens Versteeg's preference for Lingo to that of a painter for a certain pigment—and in the world of conservation, stripping an artwork of such an essential trait is unthinkable.

So what can possibly be done within those limitations? The answer—like so much else in this nascent specialty—is TBD.



Siebren Versteeg, Untitled Film II, 2006. 2 internet connected computers output to LCD screens, real-time obituary listings, real-time birth announcements, dimensions variable. Installation view: Guggenheim Media Conservation Lab, for inspection purposes. Photo: Kris McKay © Solomon R. Guggenheim Museum.

To learn more about the care of computer-based art, watch Joanna Phillips discuss the main issues here.