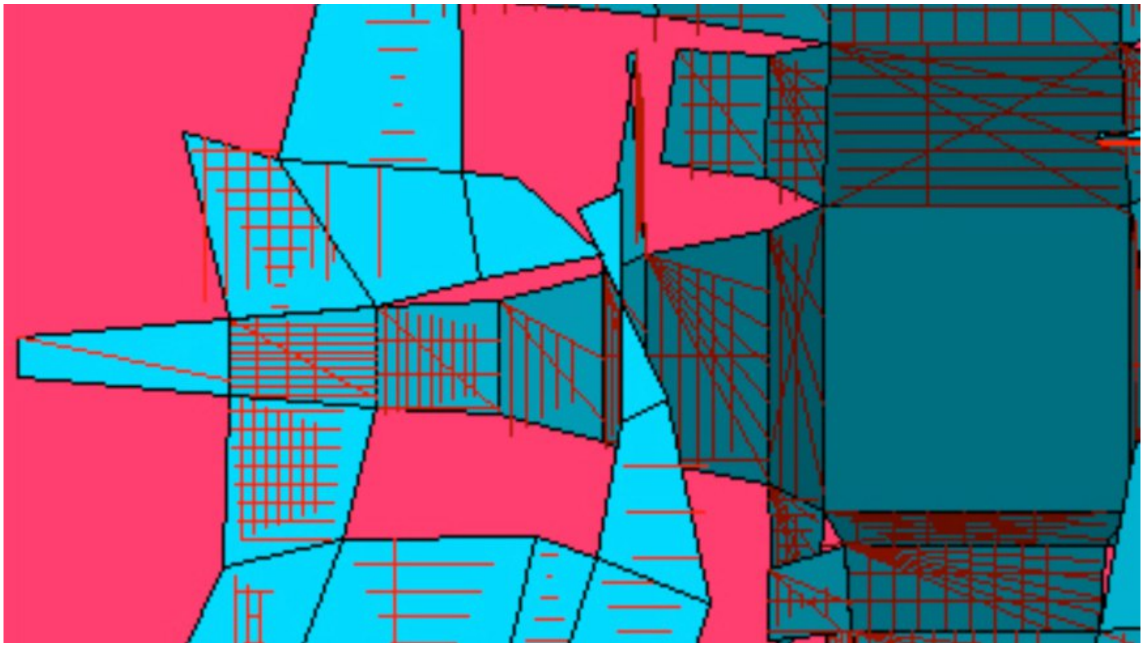


CHECKLIST • NOVEMBER 19, 2018

The Guggenheim Restores John F. Simon Jr.'s Early Web Artwork "Unfolding Object"

By Joanna Phillips, Deena Engel, Jonathan Farbowitz, Karl Toby Rosenberg

When artist John F. Simon, Jr. created his web artwork *Unfolding Object*, in 2002, the internet was still a novel medium for artists to explore. The piece, commissioned by the Guggenheim, enables visitors to create their own individual artwork online by unfolding the pages of a virtual "object"—a two-dimensional multi-faceted structure growing around a center square—click by click. Users can also see the traces of other people who have previously unfolded the same facets, represented by lines or "hash marks." Depending on the time of day, the colors of the object and the background change, so that two simultaneous users in different time zones are looking at different colors. Also, as commissioning curator Jon Ippolito wrote at the time the piece was created, "Facets that have already been unfolded darken with their distance from the central square, but a facet that has not yet been altered rewards its discoverer with a brightly colored figure untouched by previous markings."

Unfolding Object is one of three seminal, early web artworks in the museum's permanent collection. As with those other works, changes in web technology have made it difficult for contemporary users to access *Unfolding Object*. Continuous online accessibility is an important feature of web art, but contemporary web browsers stop supporting legacy technologies that were used to create early online artworks. As part of the Guggenheim's initiative to Conserve Computer-Based Art (CCBA), a cross-disciplinary team of Guggenheim conservation staff and New York University computer scientists is addressing these issues. In a previous Guggenheim blog post, the research team presented their analysis and complex restoration of Shu Lea Cheang's *Brandon* (1998-1999). Now, they have completed the restoration of *Unfolding Object*.

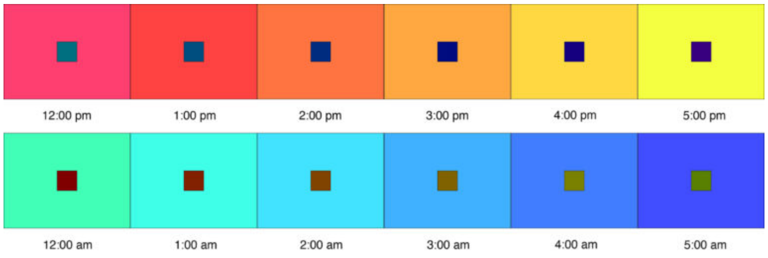
JOHN F. SIMON JR. INTERACTS WITH UNFOLDING OBJECT

John F. Simon Jr. discusses his work *Unfolding Object* (2002), which was recently restored as part of the Guggenheim's Conserving Computer-Based Art initiative.

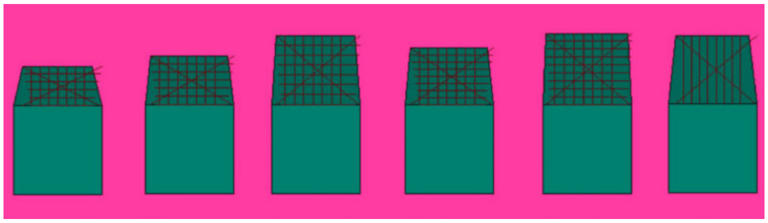
For the composition of *Unfolding Object*, Simon drew his inspiration from various sources, including the writing of theoretical physicist David Bohm. In conversation with Guggenheim staff, Simon explained that Bohm "talked about an unfolding universe . . . reality is only where your attention is." *Unfolding Object* models this concept of an unfolding universe; the artwork "is the perfect record of the places, the decisions you made. So it's a model of reality in that way. Everybody's different, but some of the reality is shared. But then you get to a place where it's just your own thing." Simon also referred to Paul Klee and drawings by Vasily Kandinsky that inspired the appearance of his web artwork.

At a time when the internet's potential for audience engagement and interactivity was just being realized, Simon embraced those new possibilities. According to Ippolito, "Rather than isolating the viewer in a cul-de-sac of individual experience, *Unfolding Object* places the beholder in a communal, although virtual, space. For this is an object, and a space, shared by the users whose fingerprints are left on the pages they unfold."

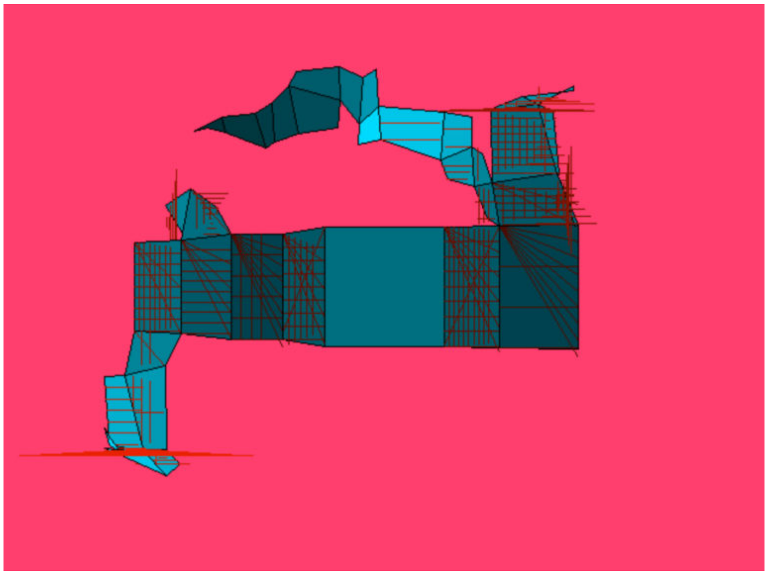
A skilled programmer, Simon coded the artwork himself using Java applet technology, which was common at the time, but is obsolete today. Interestingly, the artist decided to base *Unfolding Object* on a pre-written Java applet called "Crystal Applet" that programmer Andy Gill had freely distributed online. Simon modified this applet to handle all of *Unfolding Object*'s behaviors. The applet draws the graphical elements onscreen and responds to visitors' clicks by unfolding new pages. The quadrant of a page that the user clicks on determines the direction to which the next page unfolds, and every page unfolds with a randomly designed shape each time, making every unfolded object different. The applet also selects the object and background colors depending on the internal clock of the user's computer.



On the Guggenheim server that hosts the artwork, a program called a Java servlet communicates with the applet to collect and store data about the paths visitors take— the succession of their clicks that take them onto a specific quadrant of each unfolded page. The servlet has kept a constantly updated record of every unfolding since the piece was first released to the public in 2002. This record is accessed and updated every time a new visitor unfolds a page: For every previous visit, one horizontal line is drawn on a page; for ten previous visits, one vertical line is drawn; for every hundred previous visits, one left-to-right diagonal line; and for every thousand visits, one right-to-left diagonal line.



As described above, the artwork also displays its history of use through the brightness level of newly unfolded pages. When a visitor clicks to unfold, the program checks to see how many other visitors have taken the same path. Using this information about previous clicks, a series of conditions in the code determine whether the new page will be lighter or darker than the previous one. If others have followed the same path, the newly unfolded pages darken in color. When visitors find a less-traveled path, the new pages will keep brightening until they find a path that has never been visited. If the visitor continues unfolding along this new path, the pages will gradually darken until they reach black.



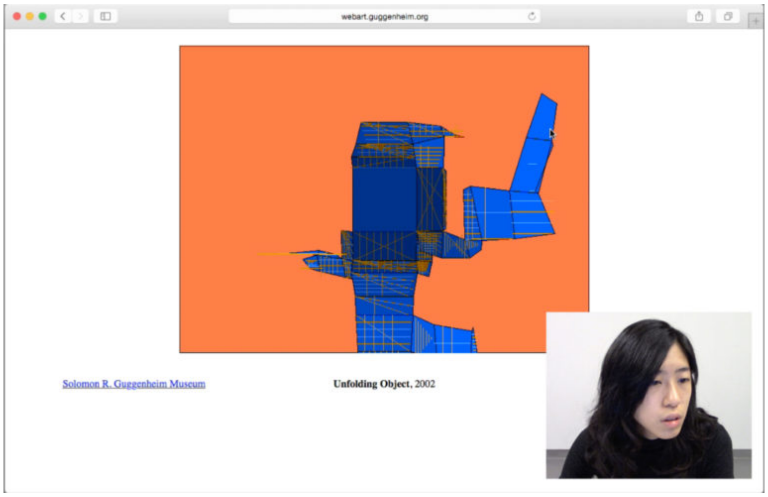
Before *Unfolding Object*'s restoration was launched in December 2017, the artwork was enrolled as a CCBA case study. Over the course of several semesters, NYU computer science students analyzed and documented the artwork's source code and its resulting aesthetic and functional behaviors. Looking for an appropriate treatment strategy, the interdisciplinary team of NYU computer scientists and Guggenheim conservation staff considered and tested different emulation options, exploring ways to run the original software on newer computer systems by simulating the older environment. However, it was found that emulation as a treatment strategy could not handle the task of constantly collecting and accessing the user data and did not result in an accurate representation of all of the original artwork's behaviors.

In dialogue with the artist, it was determined that a code migration from the obsolete Java code to the contemporary programming language JavaScript was necessary to fully restore the artwork's functionality and appearance. From an art-conservation perspective, such a code intervention is considered a rather invasive treatment. To apply conservation ethics and practice guidelines and preserve as much of the original code as possible, a new migration strategy, "code resituation," was developed and tested through extensive prototyping by the students. All of the programming work of the restoration was later carried out by former NYU computer science student Karl Toby Rosenberg (class of 2017), supported by Jonathan Farbowitz, the Guggenheim's Fellow for the Conservation of Computer-Based Art, and supervised by NYU's Deena Engel and the Guggenheim's Senior Conservator of Time-Based Media, Joanna Phillips.

By the time *Unfolding Object*'s restoration began, most contemporary web browsers had discontinued their support of Java applets. In Google Chrome, the artwork was invisible. In other browsers, such as Apple's Safari, visitors saw an "unsupported plug-in" message.

The code migration required the careful translation of the piece's logic and appearance to JavaScript while preserving as much of the original source code as possible. By applying the strategy of code resituation—copying and reanimating the original code where possible—much of the artist's code for the logic and behavior of the piece required little or no modification. One artwork behavior, however, could not be addressed without changing the original algorithm: Over the years, *Unfolding Object*'s increasing amount of visitors had led to a behavioral change of the darkening of unfolded pages. Rather gradual in the beginning of the work's life, the darkening had turned into a rapid process, resulting in a deep black object within a few clicks. In discussion with the artist, it was determined that the rate of the darkening should be "slowed down" by changing the multiplication factor that determines the brightness gradient in a non-linear way. The revised algorithm darkens the color more gradually for the first few steps, and in later steps it multiplies by a factor of 0.94 to move towards black slowly.

Testing: Guacamole Server
Darker lines similar to color of planes appear when new plane is unfolded
Sometimes disappear while plane is unfolding, after several clicks, or after switching tabs in the browser
Slight delay for plane to start unfolding after clicking
Always on PST
Reflected in colors

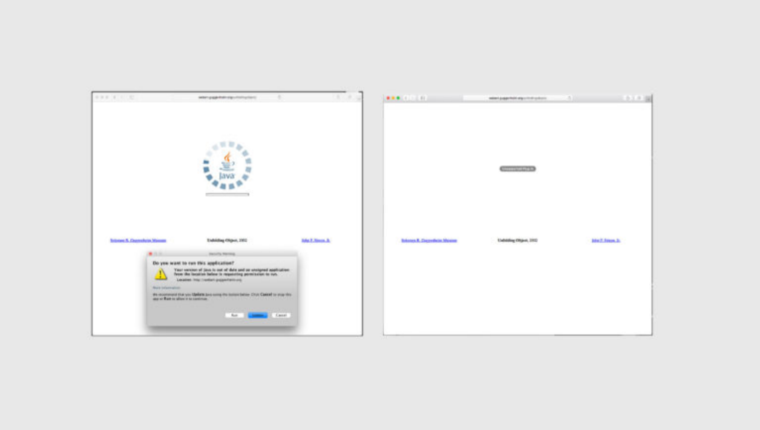


Solomon R. Guggenheim Museum

Unfolding Object, 2002

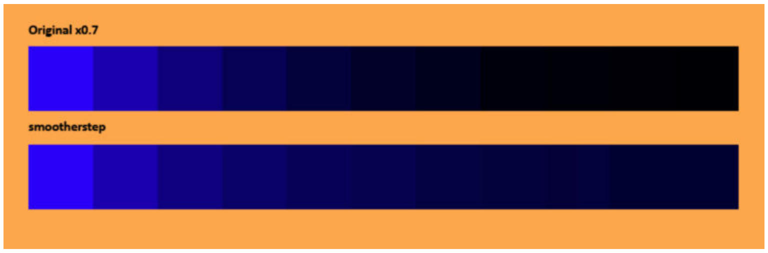
An unexpected challenge in this restoration involved recreating the original look of how shapes and lines are depicted in *Unfolding Object*. The original artwork features a sharp image where each pixel can be seen individually, resulting in staircase-like steps in the pixelation of diagonal lines. However, JavaScript's commonly used graphics library for drawing shapes and lines, Canvas 2D, anti-aliases (smooths out) the images it produces, resulting in a blurred impression of the pixelated shapes. An alternative graphics solution had to be found to recreate the original aesthetic. To accurately compare how different graphics solutions impacted the rendering of *Unfolding Object,* a program was written that allowed differently configured versions of the restoration to be run side by side.

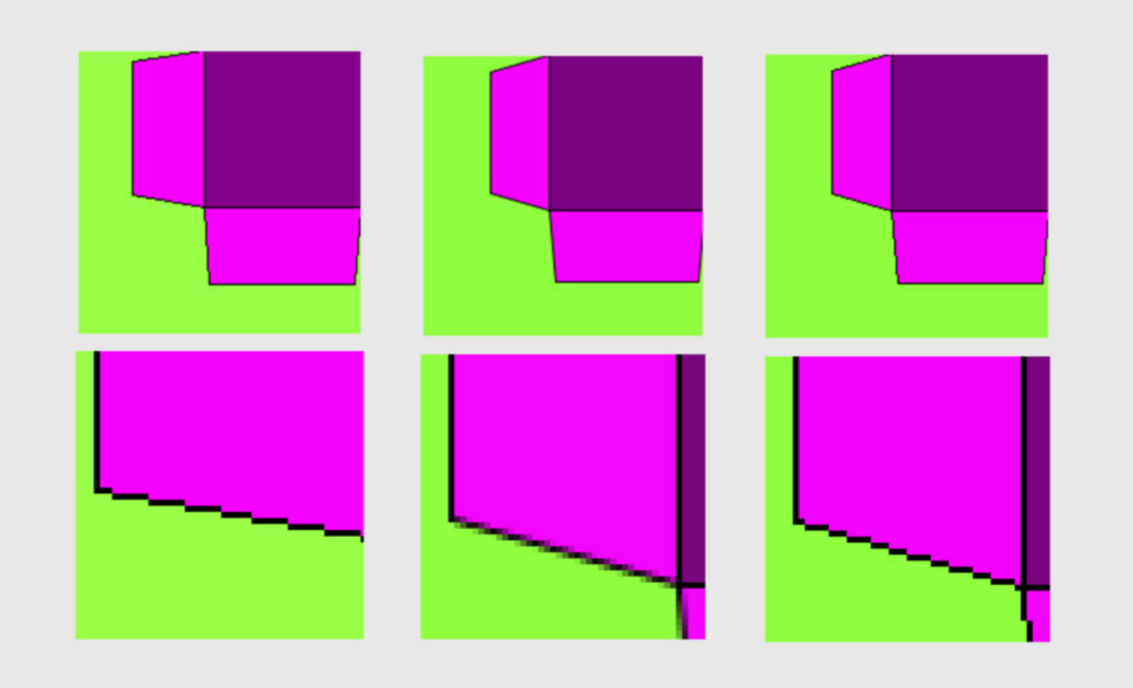
As a result of this testing, the team decided to use WebGL, a JavaScript library for programming custom graphics rendering systems. The fine-grained control it offered made it better suited to reproduce the pixelated effects needed for this restoration. The graphics code was designed with the naming conventions of *Unfolding Object*'s original Java graphics library, so although the underlying graphics code changed, the artist's original source code for setting-up drawing of shapes and lines and specifying colors was supported and required little to no modification, barring some syntactical differences between Java and JavaScript. Special care was taken to create detailed documentation of the original and new code with the intention of simplifying the maintenance and study of the artwork in the future.



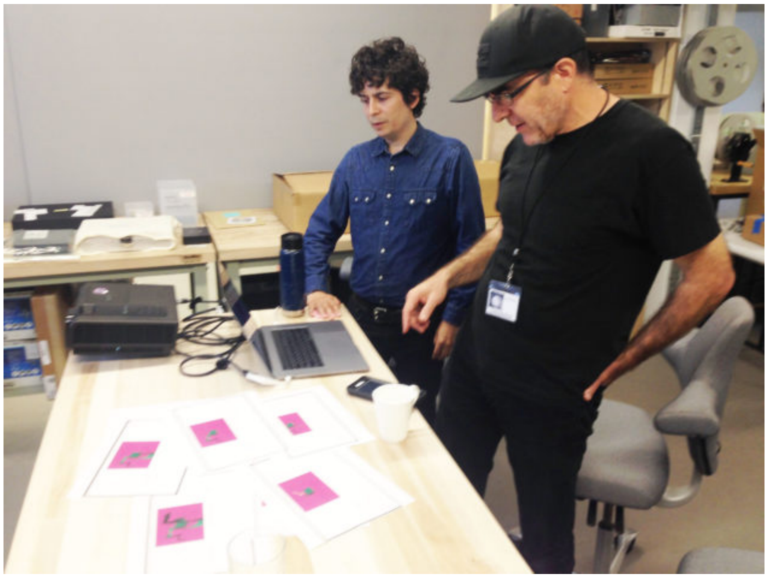
Do you want to run this application?

Your version of Java is out of date and an unsigned application from the location below is requesting permission to run.





In an interview conducted at the Guggenheim's media conservation lab, Simon discussed the myriad of ways that visitors had interacted with and enjoyed *Unfolding Object* in the past, creating their own unique shapes and patterns. The CCBA team's work on Simon's piece restores this interactivity and allows a continued history of visits to accumulate within *Unfolding Object*. New audiences can now explore this pioneering web art commission by visiting the work's original URL, unfoldingobject.guggenheim.org.



CCBA, CONSERVATION, TECHNOLOGY

CHECKLIST

JOIN TODAY →

Accessibility
Careers
E-News
For Families
Internships

Box Office
Contact Us
For Members
Host an Event
Location Shoots

Buy Tickets
Donate
For Educato
Image Licen
Press Room

CONNECT WITH US

FROM THE COLLECTION

Friedrich Vordemberge-Gildewart
Composition No. 96
1935

SEE ANOTHER

NEW YORK
Solomon R. Guggenheim Museum

VENICE
Peggy Guggenheim Collection

BILBAO
Guggenheim Museum Bilbao

ABU DHABI
Guggenheim Abu Dhabi

TERMS AND CONDITIONS PRIVACY POLICY USE OF COOKIES