

# **CREATING IMMERSIVE ART WITHOUT A PROGRAMMER: The first year for CANVAS, A Virtual Reality Environment for Museums**

## **Rose Marshack**

Krannert Art Museum  
University of Illinois  
Urbana-Champaign, Illinois USA  
rose2000@uiuc.edu

## **Nicholas Duchnowski**

University of Illinois  
Urbana-Champaign, Illinois USA  
duchnwsk@uiuc.edu

## **Abby Watt**

University of Illinois  
Urbana-Champaign, Illinois  
USA  
awatt2@uiuc.edu

## **Jennifer Jackson**

University of Illinois  
Urbana-Champaign, Illinois  
USA  
jmjacksn@uiuc.edu

## **Hank Kaczmariski**

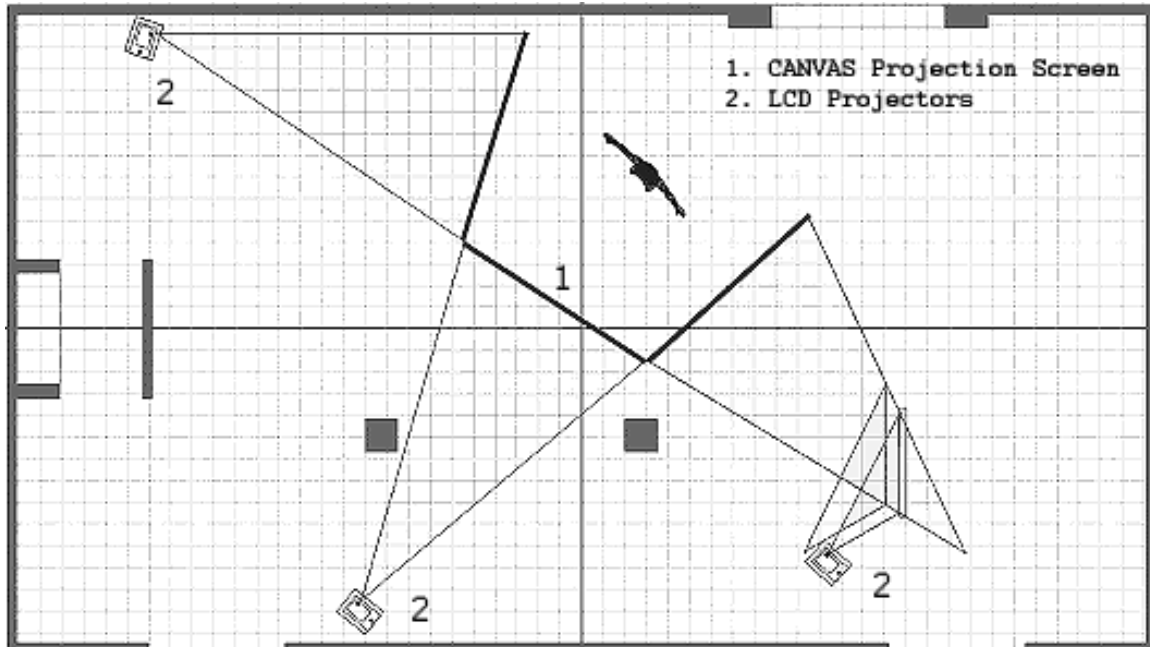
Beckman Institute  
University of Illinois  
Urbana-Champaign, Illinois  
USA  
hank@isl.uiuc.edu

## **ABSTRACT**

CANVAS, the Collaborative Advanced Navigation Virtual Art Studio [1], is a scalable, reconfigurable, display technology for modern art museums currently installed at the Krannert Art Museum at the University of Illinois as a permanent gallery fixture. The CANVAS is not a work unto itself but an environment that facilitates the creation and display of narrative art works consisting of virtual images in a multi-dimensional world. CANVAS's origins lie in CAVE™ immersive reality technology developed in the mid-1990s at the Electronic Visualization Laboratory (EVL) on the campus of the University of Illinois at Chicago. The CANVAS consists of a continuous seven-meter wide by two-meter tall screen surface mounted on a frame so as to create a three-walled display. The screen is back-projected by six consumer-grade LCD projectors fitted with circular polarizing light filters. Circular polarizing view glasses cause each eye to see a slightly different image, thus creating a 3D effect for the viewer. This is similar to the red and blue filtered glasses used to view 3D comic books, but allows for greater image fidelity than anaglyphic technology. Users navigate and interact with this multi-dimensional environment by use of a game pad or joystick.

The CANVAS operates using three consumer-grade computers running Syzygy [2], an open-source grid operating system created and developed by the Integrated Systems Laboratory of the Beckman Institute for Advanced Science and Technology at the University of Illinois at Urbana-Champaign. A multitude of applications are available to run under Syzygy – from original works of art (such as *360* by Nan Goggin, Joseph Squier, Richard Valentin and Jonas Downey), to mathematical visualizations, to biological models, to psychological experiments, to gaming environments (such as *Quake*). The creation of these types of applications, however, required knowledge of programming using C++, OpenGL, and the Syzygy infrastructure. Non-programming artists who wished to use the CANVAS as their medium were forced to employ programmers to carry out their visions – an arrangement that seemed optimal to neither the artist nor the programmer.

Our proposed solution to this problem was KAMScript, a program developed as a joint project between the Krannert Art Museum and the Beckman Institute, which seeks to provide artists with a means to create art for the CANVAS without the use of a programmer.



**Figure 1.1:** The layout of the CANVAS at the Krannert Art Museum showing (1) the three-walled projection screen and (2) the set of six LCD projectors. Illustration by Camille Goudeseune.

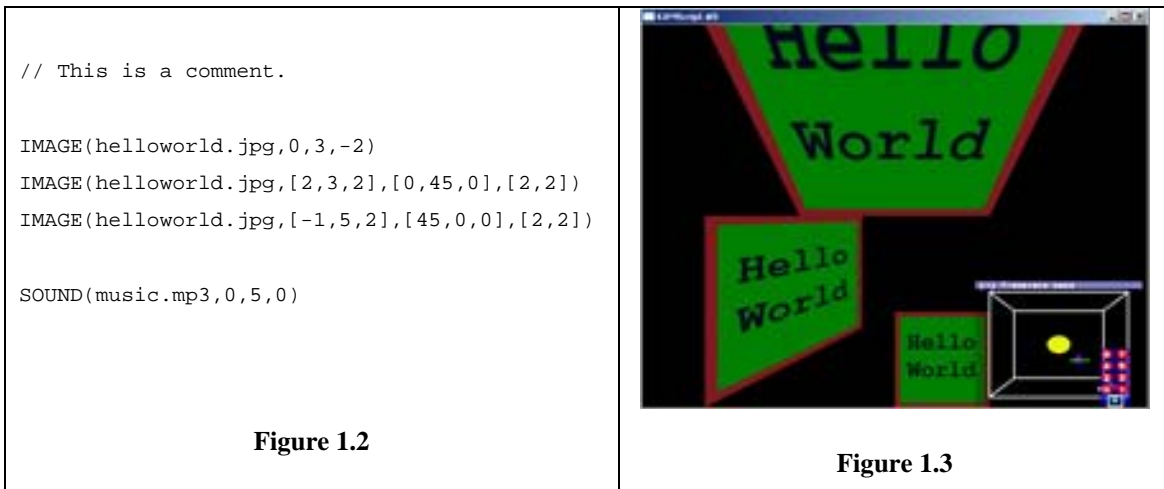
## AN INTRODUCTION TO KAMSCRIPT

KAMScript is a program that allows people with no programming experience to create three-dimensional virtual environments that can be displayed on the CANVAS. Users simply create a text file that describes the location, orientation, and size of 2D images (in JPEG file format) and 3D models (in WaveFront \*.OBJ format). KAMScript reads this text file, along with the supporting data files, and renders the scene. With KAMScript, sounds (in the form of MP3 files) can be inserted into the virtual world at defined locations. Primitive volumes - such as boxes, ellipsoids, tori, cones and teapots - can be created in any RGB color. Images can be texture-mapped onto 2D and 3D shapes. Lists of images can be displayed in flipbook-style animation. KAMScript also allows for the animation of these visual props. Objects can spin, orbit points, or follow polygonal paths.

All commands for a KAMScript virtual environment are typed into a text file. Visual elements are placed, rotated and scaled along the x-, y- and z-axes by varying the parameters of their commands. There exist additional commands for animation, varying opacity, and placement of light sources. KAMScript essentially provides access to various OpenGL and Syzygy functions (the graphics and programming libraries used to create applications for the CANVAS), but does not require users to have full knowledge of what these functions are or how they are implemented.

There are many other tools already in existence for the creation of 3D environments and virtual worlds. Many of them have a steep learning curve or are prohibitively priced for both authoring and run-time environments. KAMScript was designed to be easy to use and quick to learn. To be cost-effective for sharing artworks and facilitating collaboration among educational and arts-based communities, both the program itself and the platform upon which it runs, Syzygy, are free and open-source. KAMScript can be downloaded as an executable file and also as a run-time environment with its own text editor. These downloads can be run in stand-alone mode on personal computers, so artists do not need to have access to the CANVAS to begin creating their virtual artworks.

*An example of a KAMScript text file and resulting world:*



**Figure 1.2:** The first line places the image named “helloworld.jpg” at  $x=0$ ,  $y=3$ ,  $z=-2$ . The second line adds rotation and scaling as the second and third sets of parameters.

**Figure 1.3:** The resulting image is shown using the standalone player.

## WORKS CREATED WITH KAMSCRIPT

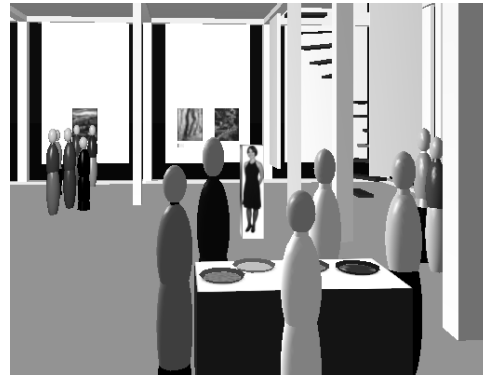
KAMScript was introduced in 2006 and is currently being used by art professors, students and the interested public. It was first used by Design faculty at the University of Illinois at Urbana-Champaign interested in new ways of portraying temporality in graphic novels. Thus the first KAMScript application, *Circuit Breaker*, was created. *Circuit Breaker* contains eight images of pages of a graphic novel. To traverse the novel, viewers must navigate along the z-axis, where events occurring later in time are portrayed further away in space.

### Icons

In October of 2006, there was an open-call to the Champaign-Urbana and Internet community to submit works made with KAMScript answering the question, “What is an Icon?” Several students responded with entries. One of these student-entries was titled *What is an Icon? (Jeopardy)*, by Abby Watt. The initial

layout of this piece is a large Jeopardy board. Flying through the board at various locations reveals thirty Jeopardy clues, the answer to all of which are, “What is an Icon?”

Another entry was a piece titled *Gallery*, by William Van Hagey. In this KAMScript application, the author creates a virtual art gallery opening. Colored boxes shape the architecture of the building. Images of paintings are hung on its walls, and spheres and ellipsoids form people milling about to look at them. There is even a table where cheese and crackers are served – and eaten! This impressive virtual world was created by a law professor who had no programming experience and was a first-time user of KAMScript.



**Figure 1.4:** *The KAMScript piece Gallery depicts people attending a gallery opening.*

## **A History of New**

In February of 2007, the *A History of New* exhibit at the Krannert Art Museum placed the CANVAS in the context of media/support history, and served as a beginning for thinking about what types of future art could be created on the CANVAS. Four KAMScript applications – worlds devised of crayon, pencil, construction paper and sumi-e – were displayed on the CANVAS. Each of these worlds placed two-dimensional cutouts of “primitive” media into a three-dimensional space. Flat, crayon-drawn trees were placed in a valley surrounded by mountains while fish jumped in the pond. This was inspired by David Pape’s CAVE™ application *CrayoLand* [4], but is markedly different in that it was written as a simple text file displayed in KAMScript, not programmed in computer code needing to be compiled into an executable.

## **TESTING KAMSCRIPT**

Since its introduction, users of KAMScript have been offering feedback regarding their experiences with the program. Many users were comfortable with a two-dimensional coordinate system, yet many were unfamiliar with the idea of the z-axis. The time-consuming nature of placing a large amount of images in space (which is what many artists wished to do) also became apparent. The (x, y, z) position of each image had to be calculated and hand-coded. Something as simple as arranging several images in a circular formation required a calculator, algebra and a lot of patience.

Along these lines, KAMScript was tested on a Freshman Discovery Course at the University of Illinois at Urbana-Champaign – a group unfamiliar with creating any kind of artistic content, much less in 3D. Having just finished high school, they understood the concept of the three-dimensional coordinate system. The problem they faced, however, seemed to lie with “getting images into the computer.” Instructing them to use digital cameras did not help the situation, since this yielded images that were simply too large. Images used in KAMScript should generally be 400 pixels square or smaller. Anything larger tends dramatically increase load and render time without noticeable gains in display quality.

Some users not only experienced difficulty with creating KAMScript applications; they also had trouble running KAMScript on the CANVAS. On the testbed we set up in the museum (allowing any visitor access to the innards of the CANVAS), KAMScript had to be run from a command prompt. Many users, though

quite accustomed to graphical computer interfaces, were completely unfamiliar with such a text-based command-line interface.

## **KAMSCRIPT VARIATIONS**

KAMScript, being open source and flexible, can be altered by programmers to suit specific needs. In some cases, the KAMScript core was modified for use in projects displayed on the CANVAS. Two of them were *Trees* and *MixTAPEStry*.

### **Trees**

This piece, by John Jennings and Damien Duffy [3], arranges the timeline of a human life in a circular formation, much like the rings in the trunk of a tree. Experiences are illustrated in two-dimensional comic panels, arranged in a spiral. The beginning of life occurs at the center, and, as the main character increases in age, the images spiral outwards. To experience a “flashback,” the viewer must navigate to an inner circle.

To aid in the creation of this piece, KAMScript was modified to place images using a spherical coordinate system, rather than a rectangular coordinate system. That is, instead of defining an (x, y, z) position for each image, an angle of rotation, angle of inclination and radial distance determined image location. This feature made the spiral arrangement much easier to accomplish, and may be incorporated into the full version of KAMScript in the future.

### **MiXTAPEStry**

MiXTAPEStry was a collaborative virtual-reality experience between the University of Illinois at Urbana-Champaign and Duke University, featuring hip-hop dancers, rappers and graffiti. In the CANVAS, 24 hand-illustrated portraits stood vertical facing the audience. At Duke University, some 2000 miles away, students danced around the room in front of cameras. The data from their movement was sent over the Internet via SuperCollider (a free Macintosh program) and caused the portraits in the CANVAS to bounce forward and backward along the z-axis. The more the people danced, the more the portraits bounced.

KAMScript was used during this event to place images in virtual space. The program was modified so that it could receive input data from the Internet and use that data to drive animation.

## **CONCLUSION**

KAMScript seeks to provide the elements, building blocks and methods necessary for successful immersive digital art. It is intended to be an easy to learn, quick to run, free to use, open-source tool for the arts community. As more options and improved capabilities are added, it should remain a program that is robust and powerful, yet easy to use. Preliminary tests at the University of Illinois at Urbana-Champaign have yielded a greater understanding of the needs of artists outside the actual authoring language. These tests will soon be expanded to include invited artists from around the globe in the hope that it will enrich their artwork as well as improve the quality and usefulness of KAMScript.

## **ACKNOWLEDGEMENTS**

We would like to thank Benjamin Schaeffer, creator of Syzygy; Jim Crowell, Camille Goudeseune, Jim Farrar, Debbie Carrier of the Integrated Systems Laboratory for providing support for Syzygy, and Kathleen Harleman, Director of Krannert Art Museum for guidance.

**KAMScript Website:** <http://www.canvas.uiuc.edu/KAMScript>

### **References**

[1] CANVAS – A Virtual Reality Environment For Museums; Hank Kaczmarski, Kathleen Harleman, University of Illinois. EVA2005, Florence

[2] Application Framework For CANVAS, The Virtual Reality Environment For Museums; Hank Kaczmarski, et al; EVA2006, London

[3] Virtual Unreality and the Shape of Time: Virtual Comics, Postmodern Self-Referentiality, and the Fourth Dimension; John Jennings, Damian Duffy, Rose Marshack, University of Illinois. ICAF2006, Washington, D.C.

[4] <http://www.evl.uic.edu/pape/projects/crayoland/>