THE IMPORTANCE OF HAVING BUGS

A Sculptor's Perspective on 3D Modeling Software

Michael O'Rourke
Pratt Institute

The by-products of process have often been a source of creativity for artists. In Chinese brush painting, the artist pays close attention to, accepts and builds upon the slight irregularities of the ink stroke as the brush glides across the paper. In much of African art, the broad, flat blade of the adze leaves a mark which is intentionally incorporated into the final look of the piece. In Modern art of the West, the turbulent brush strokes of Van Gogh, the blocky wood cutting of Brancusi, the scratched surfaces of Moore, the forged and hammered steel of Smith — all attest to the artists' desire to work with, and receive inspiration from, the process.

It is in the nature of the processes which I've chosen to cite here that they introduce a level of unpredictability into the visual results that you will get. When Henry Moore filed and chipped away at his plaster models, the process of filing and chipping left marks which, while Moore could have predicted their broad patterns, he could not have predicted in detail *exactly* the configuration of each scratch and chip. Jackson Pollock was extremely adept at controlling his drip-painting technique, but beyond the overall pattern, breadth and density of his stroke, he could not have predicted precisely where each individual drop of paint was going to hit the canvas. This unpredictability was deliberately sought by the artists, cultivated, honed and eventually used to great effect.

But the sort of cultivatable, process-derived unpredictability that Pollock had with his paint or Moore with his plaster tools is almost wholly lacking in today's' 3D computer modeling software. Computer graphic researchers have painstakingly and successfully developed a wide array of modeling tools which allow the user to very precisely develop a desired form. Realizing that precision doesn't satisfy all possible modeling needs, they have stressed the importance of randomness in the modeling process, as witnessed by the development of fractal and stochastic modeling procedures. But the process-based unpredictability that artists have valued for years is not captured fully either by techniques based on precision or by techniques based on randomness. Pollock's brush stroke is both, and simultaneously, precise and random. It is the simultaneous combination of control and lack of control which has historically been so fertile to the artist — perhaps because it echoes the control and absence of control of our human existence, the order and chaos of all existence.

In my own experience as an artist working with three dimensional computer modeling software, the closest I have come to finding modeling tools which incorporate this kind of simultaneous control/lack of control has been when I have stumbled upon certain software bugs. Odd as such a statement may sound, I have seen this happen more than once and it raises some interesting questions. What is it about these bugs that has made them so useful as artistic tools for sculpting? Could the positive aspects of these bugs be deliberately replicated to produce a different type of, perhaps a better, software modeling tool?

Programmers tend to want their code to work a certain way. When it produces unexpected results, and especially when it produces inconsistent and unpredictable results, they wince, label these results "bugs", and do their best to ferret them out. But unexpected and unpredictable results may be precisely what the artist seeks. If the results are completely unpredictable, of course, they tend to be useless. But if they are some nice combination of the somewhat (whatever that means) predictable and the somewhat unpredictable, they may capture very effectively the creative flexibility the artist enjoys with many traditional artistic tools.

surface irregularities had the serendipitous effect of giving the final form a very natural, lifeform-like look — as if the form were some unknown and as yet unclassified organism. This was very much the feeling I

Within this simple exterior form, we see the much more complex "smoothed" form. Notice the large spike at the top of the crescent's hollow and the smaller spike at the bottom. Notice also the ripples on the outer, convex

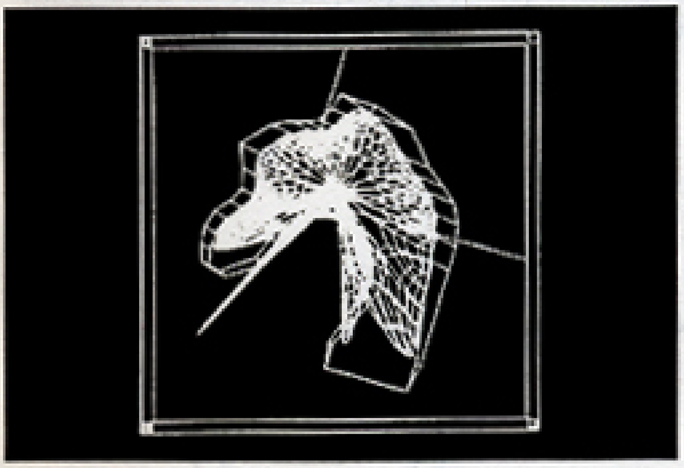

Figure 1. The simple initial exterior form, along with the complex, irregular form caused by the bug.

Figure 2. Several forms, all generated with the same bug, combined together.

The first time I encountered this sort of creative modeling bug was while making a 3D computer animation at the New York Institute of Technology in 1985-86. Within the proprietary NYIT 3D modeling software there was a function for "smoothing" a form. This function would round off the edges of a three dimensional form through a series of beveling processes. For example, if a cube were repeatedly "smoothed", it would eventually take on an approximately spherical shape.However, the procedure, I discovered, had a bug. If the original form had an extreme concavity — for example, the inside of a crescent moon — the beveling process produced unexpected spikes and ripples, rather than a gentle, overall smoothing. These unexpected

was after in my animation, and I diligently "studied" this bug.

I found that, as with an ink brush or a broad-bladed adze, it was never entirely possible to tell exactly where or how large the surface irregularities created by this bug would be. But, with practice, I also found that I could predict and control approximately where and what sort of result I would get.

Figures 1 and 2 illustrate this process and its result. In Figure 1, we see a wireframe rendering of both the original form and the "smoothed" result. The original form is the simple hatchet-shaped form in the center of the image. It was modeled by drawing a deliberately provocative crescent shaped curve and then sweeping that curve back into space.

side of the form. Figure 2 shows a shaded rendering of several similar forms which were generated with this technique and then combined to form a larger object. This final configuration, which would have been impossible without my bug, became the central motif in my animation.

An interesting side note to this account is the fact that, when I pointed out this bug to the NYIT programmers, they promptly "fixed" it. At my request, however, they left two versions of the procedure on the machine — the fixed version, which everyone else used, and the original, buggy version, which I continued to use.

A more recent example of the importance of this kind of bug arose in a series of sculptural forms which I modeled for the artist, Frank Stella. Stella was interested in the forms that smoke takes on when it drifts and floats in the air. The challenge was to model something which would be build-able (whether as a fabricated or cast object) but which retained the fluidity and was that if you scaled the control points and then un-did the scaling operation, the points did *not* return to their original positions as they should have. The location of each control point, after undoing the scale operation, was offset by some unpredictable amount. This Still, within these controllable parameters, there remained the fact that when the scale operation was undone, one never knew precisely what the form would be.

This was a *great* bug! The unpredictability of the transformation effects allowed us

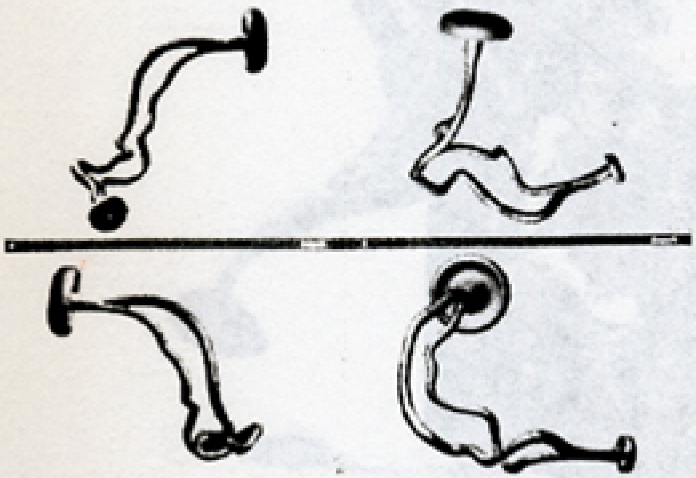

Figure 3. The simple tubular form for one of Frank Stella's "Smoke Sculptures".

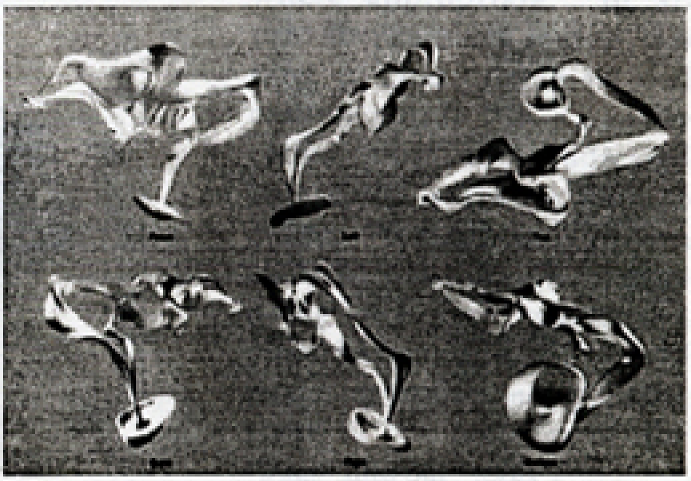

Figure 4. The same form twisted and deformed by the critical software bug.

irregularity of smoke.

The starting point for each "smoke sculpture", as we called them, was a set of photographs of a particular smoke configuration. Using these as a guide, a simple tubular extrusion was developed which followed the principle lines of the smoke. An example of this sort of structure can be seen in Figure 3, which shows one of the tubular structures as seen from four different points of view.

It was at this juncture that the critical software bug came into play. One of the modeling tools of the package we used for this work allowed the user to change the shape of a surface by transforming a network of the surface's control points. The bug, it turned out, resulted in a very irregular and unexpected deformation of the surface. Sometimes the final surface looked very similar to what the original surface had been, sometimes it acquired a bump or swell here or there, sometimes it ended up turning completely inside-out and going through itself.

What made this bug useful was that, while it was extremely unpredictable, it was not *entirely* unpredictable. There was a direct relationship between the magnitude of the original scale operation and the severity of the final deformation after de-scaling. It was also possible to restrict the surface area affected by carefully choosing the size and location of the network of control points.

to capture the feel of the irregular and unpredictable deforming effects that air currents have on smoke. Un-scaling the control points had the effect of twisting and yanking them about in much the same way that particles of smoke might be pulled and pushed by air. At the same time, since the bug's effect was directly related to how *much* the points were scaled and where, one could exercise some control over the degree and location of deformation. Like a stick of charcoal, an ink brush or a plaster chisel, this tool afforded a delightfully creative unpredictability, while at the same time allowing control of the broader effects of the modeling changes.

Figure 5. Frank Stella, Zimming, *1992. Stainless steel, bronze, and aluminum. 16"x13"x12".*

An example of this scaling deformation bug can be seen in Figure 4. The form we see here, represented from six different views, began as the same tubular structure we saw in Figure 3. Figure 4 shows the result of repeatedly using the bug just described to deform the tubular structure until it had a much more fluid, "smoky" feel to it. Figure 5 shows one from the series of Mr. Stella's sculptures which incorporated this sort of deformed "smoke sculpture" form. The form in the lower right of this photo, protruding from the bottom of the sculpture, is the same form we see illustrated in its digital incarnation in Figure 4.

The kind of bug I have described here can be extremely useful to an artist. It can, in fact, cease to be a "bug" in the artist's mind, and become a "tool". I believe that this kind of tool could be, and should be, deliberately developed by computer graphics programmers and researchers. Currently available modeling tools tend to be either too precise or too random to suit the needs of many artists. There is a need for tools which, like the bugs I have described, have a bit of a mind of their own, but still remain controllable by the artist in a broad sense.

The artistic endeavor is an inquiry, an attempt to understand what has not yet been understood. One of the ways artists do this is to abandon control and allow the process, within constraints, to generate unforeseen shapes, marks or colors. The artist then reacts to these, attempting to find order and sense in them, modifying them if possible, repeating them where appropriate. In 3D modeling software to date, this kind of controllable lack of control is rare. It is usually called "bug" and eradicated before the software gets to market. We may learn a lot from studying our bugs. And in so doing, we may produce a powerful, and different, type of modeling tool by incorporating some of the features of our "bugs" into our "tools".

Michael O'Rourke is a sculptor and animator who works with computer graphics. He is also an Associate Professor at Pratt Institute, Brooklyn,, NY. His book, Principles of 3D Computer Graphics, *will be published by Design Books in the Fall of 1994. His studio address is: 44 Tompkins Place, Brooklyn, NY 11231.*

Layout by Chuen Choi