

and check out a program that lets you draw simple fractals in BASIC. Being computer phobic, I was defiant at first, and preferred to keep drawing animation cells by hand. But in the span of a few hours I became fascinated with this new way of describing an image, and a transformation of that image. With this program (which looked like alphabet soup to me), I could (very clumsily) edit the first in a string of four numbers in the midst of the soup to be slightly different than before, run the program, take a frame of the computer screen with my movie camera, and repeat the process over and over again. The changes in the images implied that the developed film would reveal a complex organism which gracefully writhed in harmonic motion, like some underwater sea creature. As I began to contract the flu late into the night, and as my thumb began to form a blister from repeatedly clicking the camera, I began to recognize the significance of the forms and motions that were unfolding in the as yet undeveloped film: simple, linear changes in one small number in the series beget a cascade of complex hierarchical movements in the form, due to the recursive nature of the fractal. The code was terse and the transformation was simpleminded, but the generated form was chaotic, complex and surprisingly organic.

### Tree Epiphany

Soon, I figured out how to create tree shapes using recursive algorithms. My excitement was like that reported by Richard Dawkins when he first brought his biomorphs to life for his *Blind Watchmaker* software<sup>8</sup>. I began to appreciate the vastness of the family tree of trees (a “genetic space”), and had to take a walk in the night air. I began to watch the bare tree branches (it was winter) as they glided past my vision while I walked down the street. I saw the trees in a new way. I saw them growing—I saw them as grown things. This led to a trend of looking at all things as grown from generative forces - interacting with other things as they take shape. It was a new eye on things as procedural things, not designed or created, but evolved through a dynamism.

“Procedural modeling” was a refreshing new approach to many in the computergraphics world who were growing tired of chrome spheres and texturemapped teapots, and worlds constructed with building blocks. Some computer art purists have suggested that the computer, being a dynamic information processing machine, is best at modeling dynamic phenomena: growth, thought, evolution, language, rather than building upon past imaging technologies and artforms. Artist Ed Zajec sees processing techniques such as recursion as a way of expressing with as much potential as the technique of perspective had in the Renaissance<sup>9</sup>. Recursion is a motor which aids in the emergence of form, its generation, its development through time. It also appeals to the musically-inclined.

### Composing Seeds

Recursion means you can artfully compose a seed (a few numbers, an equation, a geometrical transformation, a musical

motif, etc.) and an environment within which that seed will grow, and then let it grow. The grown result can then be evaluated aesthetically and then the seed can be re-designed, in an attempt to create more interesting or expressive grown forms and motions. It’s usually difficult to relate the change in the seed with the change in the grown thing, because the “embryology”—the expression from *genotype* to *phenotype*, may be complex and unpredictable<sup>10</sup>. But after a while of doing this, one can acquire a skill at predicting the results of making such changes to the seed. My explorations in this activity led to an obvious approach to crafting families of images and animations: to use *genetic algorithms* (the technique of modeling the dynamics of Darwinian evolution in software). Genetic algorithms are optimizing schemes which use the genetic operators of reproduction, mating, mutating, and dying, to evolve a population of *genotypes* (the parameters which determine the characteristics of a family of images, for instance), to improve what the *genotypes* represent: the *phenotypes* (the images themselves).

The genetic algorithm became a tool to help me breed these images and motions, where my aesthetic evaluations became a “survival of the fittest”. This kind of technique is similar to what Karl Sims had developed, using genetic programming (expressions in the LISP language which can mate and mutate) for image breeding and animation scripting. It suggests a possibly whole new kind of artmaking process, which Sims called, “interactive evolution”<sup>11</sup>.

### Evolving Animals

I had also used genetic algorithms in an experiment for evolving populations of simple, articulated stick figures rendered in a virtual 3D world with gravity, friction, and momentum<sup>12</sup>. These were presented at the latest Artificial Life gathering at MIT. Variety in these populations was determined by a genotype (a set of genes controlling parameters for anatomies and motions among the body parts). These very simple shapes evolved seemingly purposeful, and often amusing motions, as the population evolved, driven by fitness pressures for locomotion. They grew better proportioned extremities and rhythmic gaits—and many unexpected solutions to the general task of locomotion.

Another set of creatures I had developed existed in a virtual underwater world. Their evolutionary fitness pressures were for covering more distances in water, in any way they happened to chance upon. In these populations, starting from random primordial beginnings, later generations produced longer limbs and wider, more coordinated strokes.

At the MIT conference, Karl Sims presented a family of 3D land and undersea creatures composed of blocks, whose behaviors and anatomies had evolved purposeful behavior through competitive situations, using the genetic programming technique<sup>13</sup>. These objects incorporated a physically-based model and embryological process determining morphology and mo-