Copy Cat

Version 2.0

This file will appear best if you have the font New York.

Contents:

Legalities and Distribution:

Copy Cat is Emailware. The only thing I ask if you receive Copy Cat and choose to keep and use it is that you email me. My email address is keithw@wam.umd.edu. I ask this for a couple of reasons. One, I'm simply curious how far it reaches out there. There's something cool about knowing that hundreds or thousands of people may be using my program. Two, I don't have an opportunity to test it on different CPUs and would be curious about any strange bugs that may arise.

The executable application may only be distributed free of charge. No monetary compensation can be traded for Copy Cat. Thank you for your cooperation.

Note:

Several of the diagrams in the documentation are despicable black and white graphics. Do not despair. All the windows and dialog boxes in Copy Cat are beautifully ordained in a myriad display of over 256 of your most favorite colors. Using reduced depth graphics in the documentation reduced the size of this file by several hundred kilobytes.

Introduction:

This is the second major artificial life simulation I have written. It would be more accurate to say that it is the third because updating Copy Cat to version 2.0 was practically a new undertaking as far as programming projects is concerned. I became addicted to artificial life about a year and a half ago when I found some books on the subject in a University bookstore. I have since studied the subject on my own time extensively. I have written an implementation of John Conway's Game of Life, a variant on Stuart Kauffman's Kauffman Model and a self-designed simulation called Bugs in which simple nonlearning creatures follow genetically determined rules in an attempt to find food, the most genetically adapted surviving and reproducing the most. In Copy Cat I decided to tackle a slightly more difficult and more specific problem than in Bugs. Bugs just optimizes search strategies such as how long to wander in a straight line without finding food before trying a different direction and what energy levels at which to reproduce. Copy Cat is a an attempt to evolve mimicry, a phenomenon known to exist in nature which is often used as evidence of evolution and natural selection.

Copy Cat was written on a Mac IIvx running system 7.1 using Think C 6.0 and on a Power Mac 7500 (601 processor) running system 7.5.3 using CodeWarrior 8 and CodeWarrior 9.

What is mimicry?:

Mimicry occurs when one organism looks like another in an attempt to gain some advantage of having that appearance without paying the cost that the original organism must pay for their appearance. To put it differently, say you have two populations of organisms (I am avoiding the word species because even though mimicry usually occurs in one species imitating another, it does occasionally happen in different populations of the same species). One population has a physical trait that they can benefit and one population does not possess that trait. If this trait has an influence on the organism's interactions with other organisms of whatever kind it may encounter, then the second population would benefit from this physical trait if they only looked like the first population. They don't actually have to possess the physical trait itself to benefit from it's advantages.

There are a couple different kinds of mimicry. The most hailed example is that of a two species both of which are potential prey to a certain species of predators. One of the populations of prey are poisonous, one is not. The nonpoisonous species looks nearly identical to the poisonous species and so it happens that the predators can't tell that the nonpoisonous species is actually edible. All the predators know is that everything looks poisonous and must therefore be avoided. This is the example that Copy Cat attempts to reproduce. Another examples of mimicry occurs within a single species some individuals are poisonous, some are not, the benefits of the species' poison are reaped by the entire species. Another example is animals that mimic plants. Walking Sticks are insects that when not moving, look strikingly like the branch on which they are walking. Sometimes plants mimic animals too. Several flowers mimic insects to seduce insects into "mating" with the flower for the purposes of spreading the flower's pollen. Another example is mimicking the predator of a would be predator. There is a caterpillar that looks frighteningly like a dangerous snake.

System Requirements:

Copy Cat is a fat binary application so on your Power Mac it will take full advantage of your RISC capabilities and on your 68k Mac it will still run without compatibility problems. While Copy Cat would make some use of a floating point unit, versions both compiled to use and to not use and fpu do fairly well, so, the 68k half Copy Cat does not require an fpu and should work on any computer that doesn't have one. Copy Cat pretty much requires 256 colors. Since the whole premise of the program is the evolving visual patterns on the backs of the prey, this range of color is pretty much a necessity to understand what is going on. Only sixteen colors are actual available to the prey, but they are chosen from the 256 color palette. It is also difficult to read a couple of variables and the statistics graphs without 256 colors. I haven't tried Copy Cat with system software earlier than System 7.1 but there should be no problem with System 6 so far as I can tell.

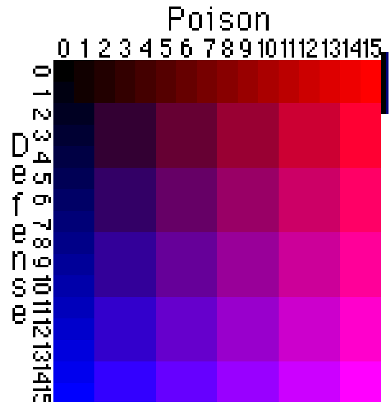
The General Overview:

Copy Cat occurs in a two-dimensional toroidal (sides of the rectangular world wrap together. Walk off one side, appear on the other) world. There are two things in the world of Copy Cat: prey and predators. All reproduction is asexual. A prey or predator is cloned and mutated slightly when a new organism is needed. Prey and predators move forward with a speed ranging from 1 to 20 pixels per turn (that's the number of pixels they move with the zoom set to x8), and rotate in 22.5 degree increments (16 possible directions). Prey and predators see a 180 degree view of their world out to a genetically determined distance.

What You See:

Depending on your zoom and the size of your world you will be looking at some fraction of the landscape. The general background is neutral gray on which you will see the two different kinds of objects: prey and predators. Prey are bright colored circles with a small circular head. The heads have three purposes. To show the direction the prey is pointing, the defensive capability, and the poisonousness of the prey. The more defensive a prey is, the stronger the blue in the head will be, the more poisonous a prey is, the stronger the red in the head will be. If equally magenta colors emerge, then the defense and poison are probably pretty equal and the brightness of the head will designate the value of those two genes. Predators are lozenge shaped, also with circular heads. The color of the predator's body shows how many memories of poisonous prey it has accumulated. Black prey have no memories, lighter shades of gray denote more memories, and white is the most memories possible, six. This variation in color does not change with the Maximum Number of Memories variable (see below). The color scale is unchanging. If you change the Maximum

Number of Memories, the predators will simply never become the lightest shades of gray or white. The color of the head of a predator shows its fatality. Brighter shades of red designate stronger fatality values. It is important to note that predators cannot see the color of a prey's head. The prey have colored heads simply so you can tell their defensive and poisonous capabilities by glancing at the world. The same goes for the color of the predators' heads, prey cannot see it. The prey head color can be decoded using this chart:



The big blocks of identical colors result from the color palette. It would be impossible to define a palette with 256 differently mixed shades because then only shades of magenta would be available to the program, and all colors are necessary for the patterns and other aspects of the program.

The Menu Bar and Menu Options:

The menu bar and menu options are as follows:

Apple:
About Copy Cat: You know the routine.

File:
New: Begin a new simulation using the settings in the Settings dialog box.
Pause: Yep, you guessed it. The simulation only pauses between timesteps so you will have to wait for all the creatures to be updated before the simulation pauses.
Step: Advance one timestep per creature. Only available when the simulation is paused.
Quit: This is the bonus question folks.

Edit:
Not Used.

Simulation:

Kill Prey: Randomly kills a number of prey you set in the Kill Prey dialog box:
- Kill a Specific Number: Kill a flat number of prey.
- Kill a Percentage: Kill a percentage of the prey population.

Kill Predators: Randomly kills a number of predators you set in the Kill Predators dialog box:
- Kill Specific Number: Kill a flat number of predators.
- Kill a Percentage: Kill a percentage of the predator population.

Make Prey: Makes the number of prey you set according to the criteria you choose in the Make Prey dialog box:
- Unrelated: Makes random prey using the same method for creating the initial prey.
- Exact Clones: Makes identical clones of the various prey.
- Mutated Clones: Makes mutated clones of the various prey using the mutation rate set in the Settings dialog box.
- Make a Specific Number: Make a flat number of prey.
- Make a Percentage: Increase the prey population by a percentage.

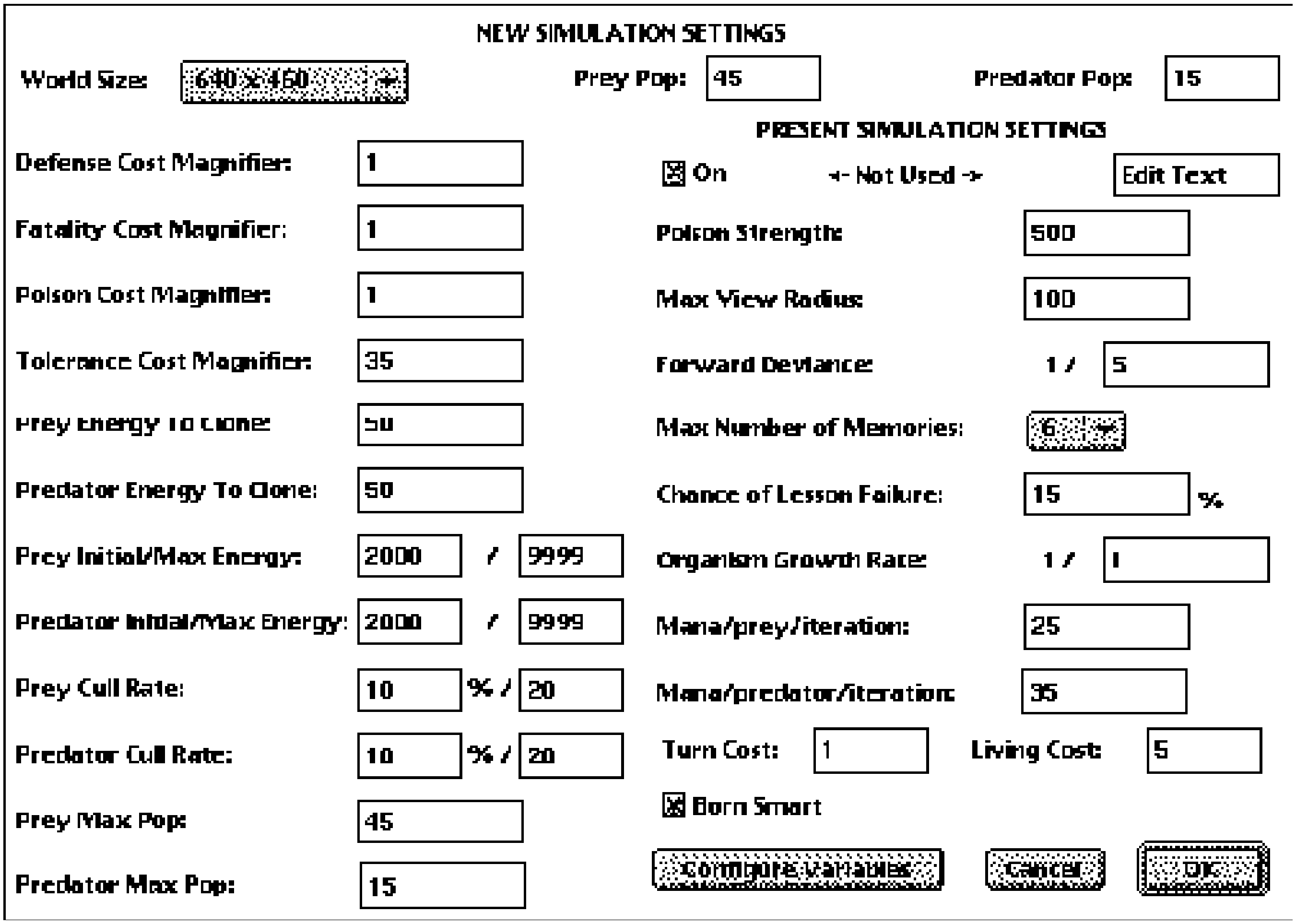
Make Predators: Makes the number of predators you set according to the criteria you choose in the Make Predators dialog box:
- Unrelated: Makes random predators using the same method for creating the initial predators .
- Exact Clones: Makes identical clones of the various predators .
- Mutated Clones: Makes mutated clones of the various predators using the mutation rate set in the Settings dialog box.
- Make a Specific Number: Make a flat number of predators .
- Make a Percentage: Increase the predator population by a percentage.

Tornado: Randomly distributes all creatures. The first three move all the creatures up to a certain distance in a random direction. The last option reassigns every creature's coordinates, thus assuring a totally heterogeneous distribution.

Radiation: Mutates the DNA of all creatures. The stronger options don't actually cause more drastic changes but do cause more changes.

Randomize Ages: Randomly reassigns the ages of all creatures to a value between one and one less than the maximum age for prey or predators (whichever each creature is) as per the Settings dialog box.

Settings: Displays the Settings dialog box:



The top three variables under New Simulation Settings won't take effect until a new simulation is begun. All the other variables in this dialog box will take effects immediately upon hitting the OK button, including any changes you may make inside the Configure Variables dialog box under the button of the same name. The above variables are as follows:

World Size: Pixel resolution of the map with zoom set to x8.

Prey Pop: Number of prey randomly generated at the beginning of the simulation. This cannot be greater than the value of Prey Max Pop.

Predator Pop: Number of predators randomly generated at the beginning of the simulation. This cannot be greater than the value of Predator Max Pop.

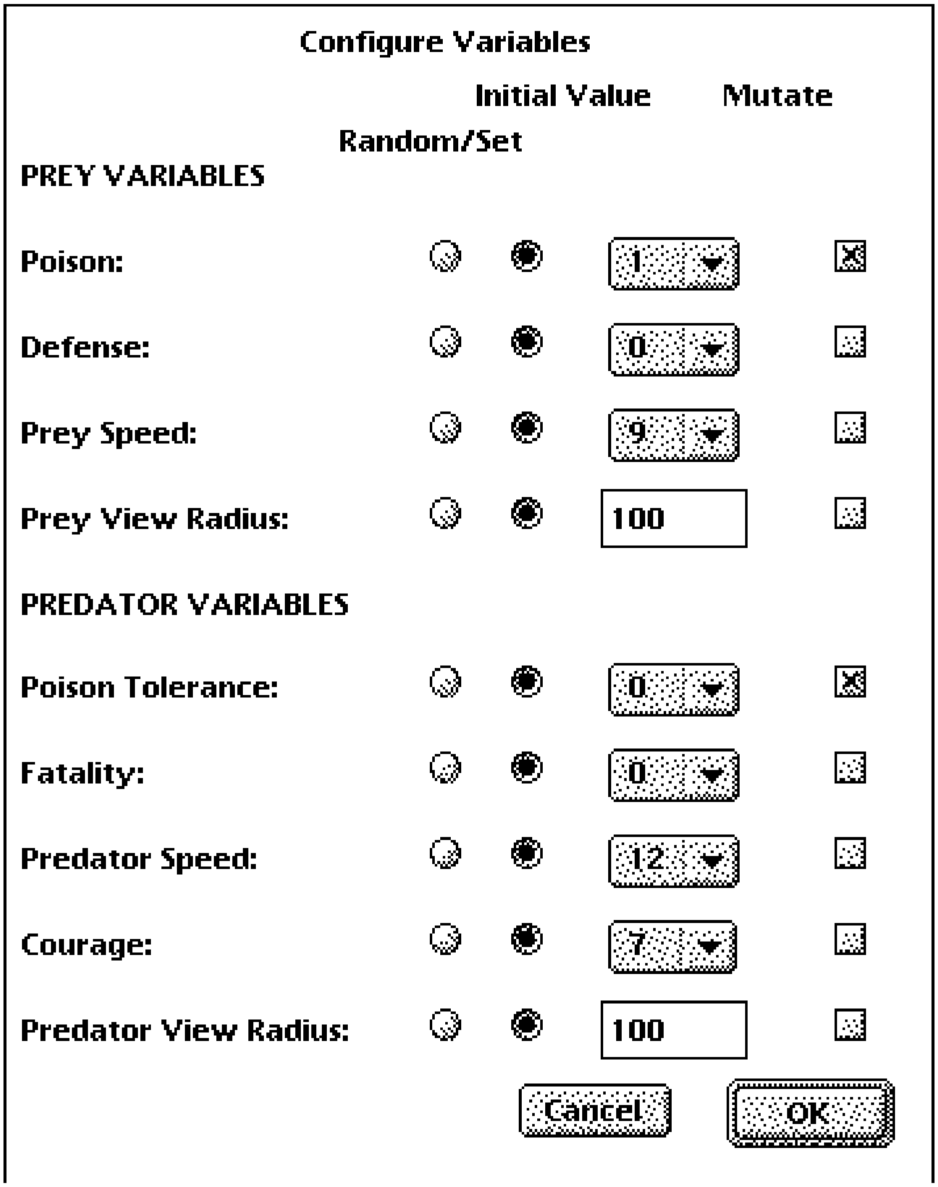
Defense Cost Magnifier: Each prey loses this value times their defense gene value amount of energy each timestep.

Fatality Cost Magnifier: Each predator loses this value times their fatality gene value amount of energy each timestep.

Poison Cost Magnifier: Each prey loses this value times their Poison gene value amount of energy each timstep.

Tolerance Cost Magnifier: Each prey loses this value times their Poison Tolerance gene value amount of energy each timestep.
Prey Energy to Clone: Minimum amount of energy that a prey must have in order for it to be cloned.
Predator Energy to Clone: Minimum amount of energy that a predator must have in order for it to be cloned.
Prey Initial/Max Energy: Amount of energy that the prey made at the beginning of the simulation have, and the maximum amount of energy a prey can have. Initial energy cannot be greater than maximum energy.
Predator Initial/Max Energy: Amount of energy that the predators made the beginning of the simulation have, and the maximum amount of energy that a predator can have. Initial energy cannot be greater than maximum energy.
Prey Cull Rate: Percentage of the prey population that is killed every X number of timesteps.
Predator Cull Rate: Percentage of the predator population that is killed every X number of timesteps.
Prey Max Pop: Maximum prey population.
Predator Max Pop: Maximum predator population.
<- Not Used->: Two boxes that I left in that are of no use presently.
Poison Strength: When a predator eats a prey that is more poisonous than the predator's tolerance, the predator loses this value time the prey's poison gene value amount of energy.
Max View Radius: Maximum distance (in pixels as x8 zoom) that prey and predators can evolve to see at.
Forward Deviance: One time out of this value number of times, a prey or predator will turn instead of continuing forward when all its inputs receive values of zero.
Max Number of Memories: Maximum number of memories that predators can obtain.
Chance of Lesson Failure: Percentage chance that a predator will not acquire a particular prey color memory from its clone parent.
Organism Growth Rate: It take a prey or predator this number of timesteps to grow by on pixel in radius. Thus, it takes ten times this number of timesteps to reach full-grown size.
Mana/Prey/Iteration: Amount of energy administered to each prey each timestep.
Mana/Predator/Iteration: Amount of energy administered to each predator each timestep.
Turn Cost: Amount of energy it cost a prey or predator to turn.
Living Cost: Amount of energy prey and predators lose every turn just to live.
Born Smart: Brains are pretailored according to specifications listed further on in the documentation. Otherwise, the initial prey and predators have random brains and clones have mutated brains from their clone parents.

Configure Variables: Brings up the Configure Variables dialog box:



The Random/Set buttons determine if those variables will be randomly assigned or set to preset values for the initial organisms. If they are set, the value of the setting will be whatever the popup menu or text box says. The Mutate checkbox determines if that variable is allowed to mutate when clones are produced.

View:
The first group of four set the view zoom of the window. Some of the wider zooms may be unavailable depending on the size of the world you set in the Settings dialog box.

The second group of four maneuver your viewing window around on the world. These commands move the window in half-window wide increments, depending on the zoom size you have set. You cannot scroll around the edges of the world. Sorry.

Draw: Brings up the Draw submenu:

Update Screen: Sets how often the screen is redrawn:

Now: Redraw screen immediately, just once. Use this to take a glance at the screen when you have screen updating turned off or whenever the screen becomes messy. Sometimes the plants become sloppy (especially when Draw Views is on). Use Update Now to fix this.

In Real Time: Each creature is redrawn every timestep.

Every # times: Creatures are not redrawn every timestep but an Update Now command is executed every # timesteps.

Never: Completely turn off screen updating. This also turns off cursor updating which can help speed up the simulation a little bit more.

Draw Prey Vision: Draws a 180 degree semicircle around every prey showing both the viewing distance and the viewing inputs of every prey. The three sections of the viewing semicircle will be one of three colors denoting the overall input the prey is receiving in that section of its view:

Black: No inputs
Red: Overall bad inputs

Draw Predator Vision: Draws a 180 degree semicircle around every predator showing both the viewing distance and the viewing inputs of every predator. The three sections of the viewing semicircle will be one of three colors denoting the overall input the predator is receiving in that section of its view:

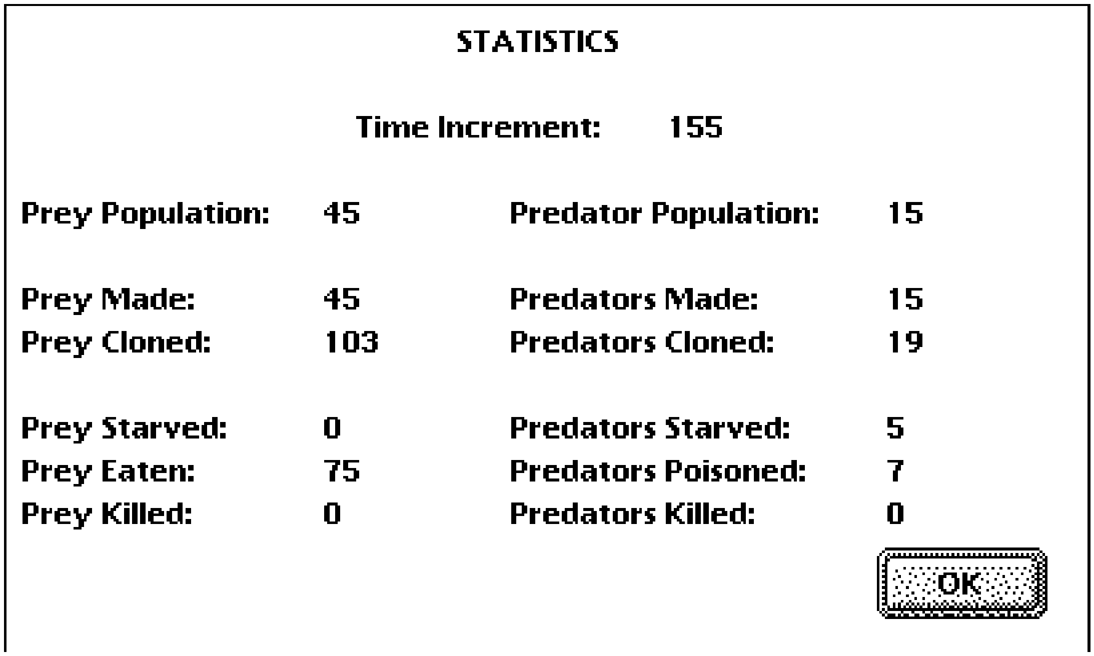
Black: No inputs
Yellow: Overall good inputs

Draw Prey Data: It is possible to print some of the numerical information about the creatures on top of them as they meander around their world. Prey data is written in black. The options you can turn on and off are energy, defense, poison, and speed.

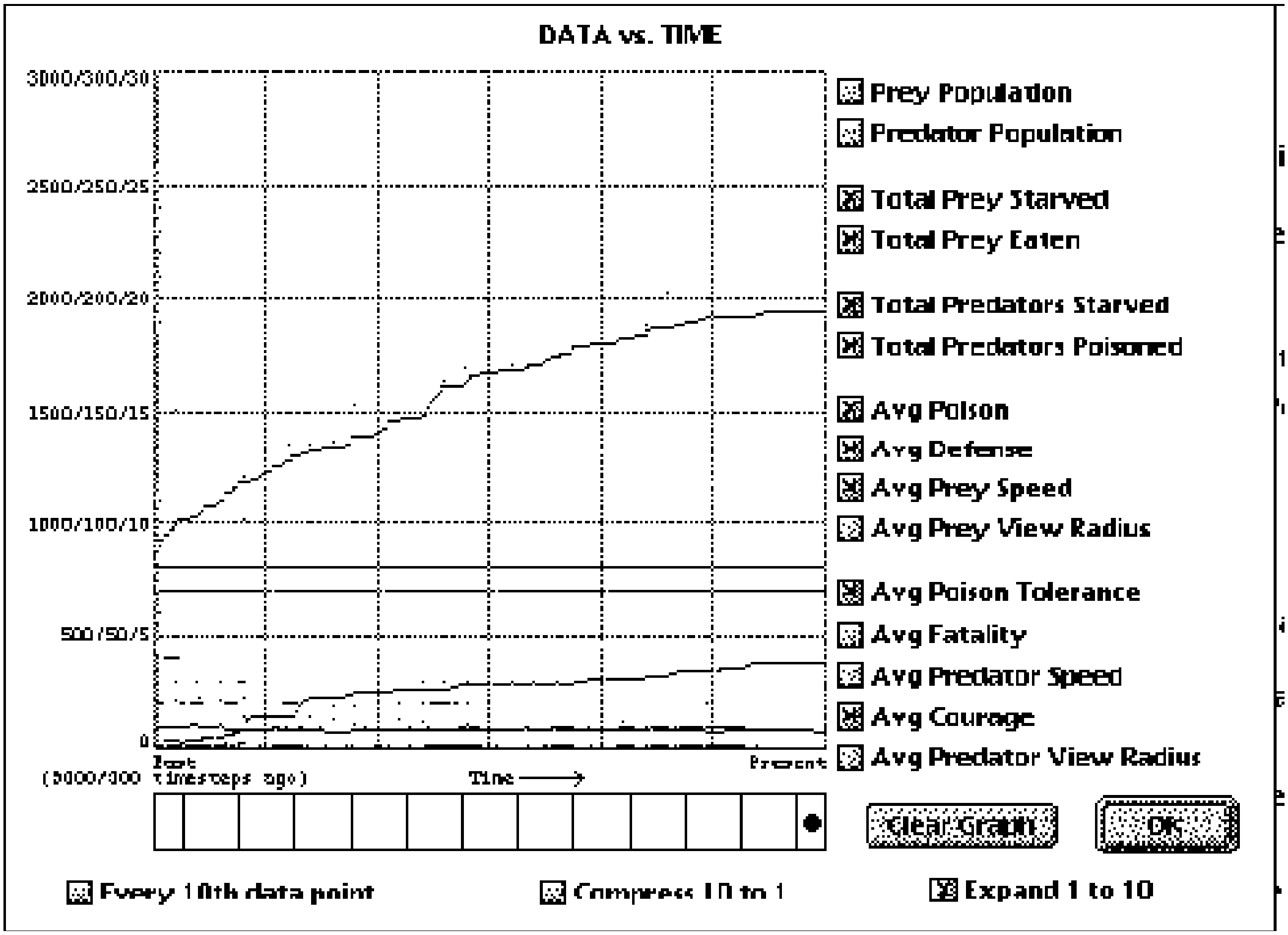
Draw Predator Data: It is possible to print some of the numerical information about the creatures on top of them as they meander around their world. Predator data is written in black. The options you can turn on and off are energy, fatality, poison tolerance, courage, and speed.

Info:

Show Statistics: Displays the Statistics dialog box in which all relevant information pertaining to the present simulation can be seen:



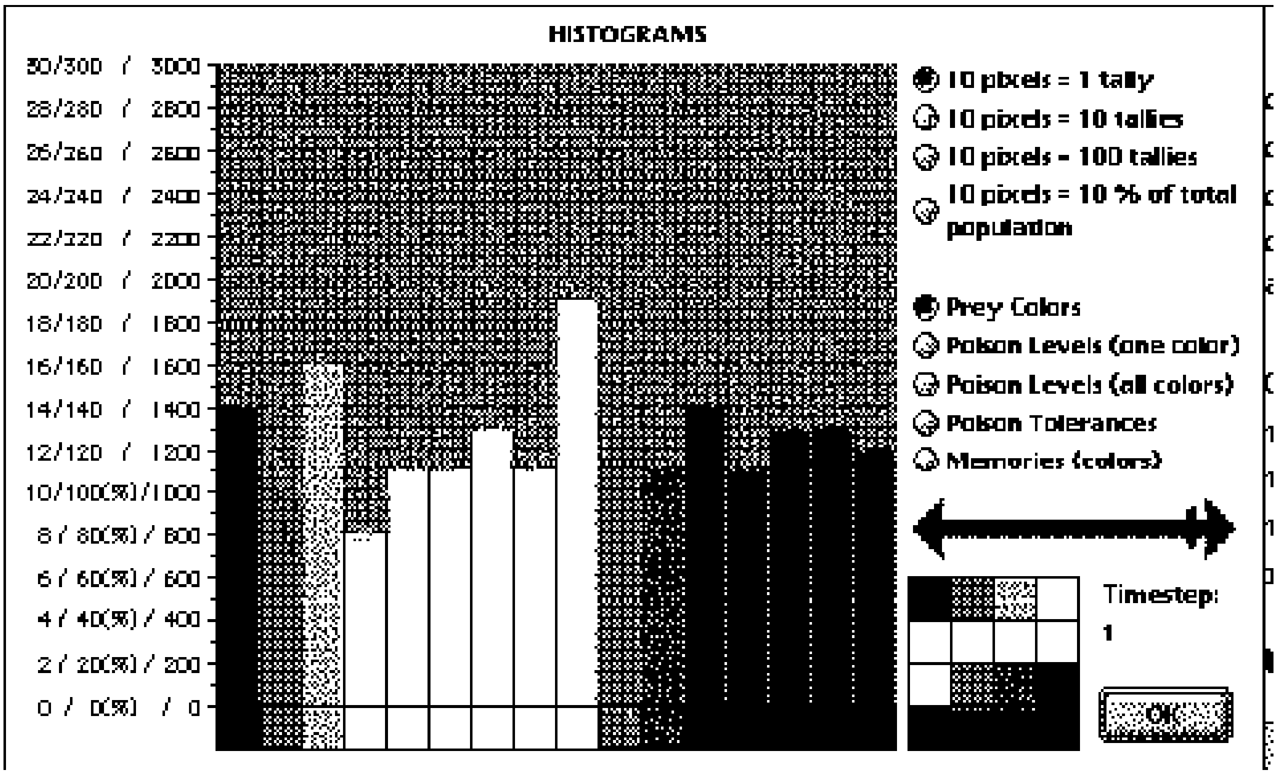
<u>Time Increment</u>: Present timestep
<u>Prey Population</u>: Present prey population
<u>Prey Made</u>: Initial number of prey plus prey made with the menu command.
<u>Prey Cloned</u>: Number of prey cloned including the menu and cursor commands.
<u>Prey Starved</u>: Number of prey starved to death.
<u>Prey Eaten</u>: Number of prey eaten by predators.
<u>Prey Killed</u>: Number of prey killed with the menu and cursor commands.
<u>Predator Population</u>: Present predator population
<u>Predators Made</u>: Initial number of predators plus predators made with the menu command.
<u>Predators Cloned</u>: Number of predators cloned including the menu and cursor commands.
<u>Predators Starved</u>: Number of predators starved to death.
<u>Predators Poisoned</u>: Number of predators poisoned to death by a single prey (as opposed to starving simply by running out of energy)
<u>Predators Killed</u>: Number of predators killed with the menu and cursor commands.
<u>Show Graphs</u>: Displays the Graphs dialog box:



Prey information is plotted in blue and predator information in red. The three columns of numbers on the left side pertain to data that is plotted at a 10 to 1, 1 to 1, and 1 to 10 ratio. The only possible point of confusion is the bar across the bottom of the graph. The data arrays store 3000 points of data for each variable. The duration of time this spans is dependent on the Data vs. Time collection rate. The window you look at in the graph is 300 pixels wide. With Every 10th Data Point on, you then see the entire data collection, but only every tenth data point. If you want to zoom in on part of the graph you can look at 300 consecutive data points at several marks along the entire array. Clicking in the boxes along the bottom chooses which 300 you will see. Clicking the furthest left box shows the oldest data, clicking the furthest right box shows the most recent data (right up to the present moment) and clicking in between chooses various points in time.

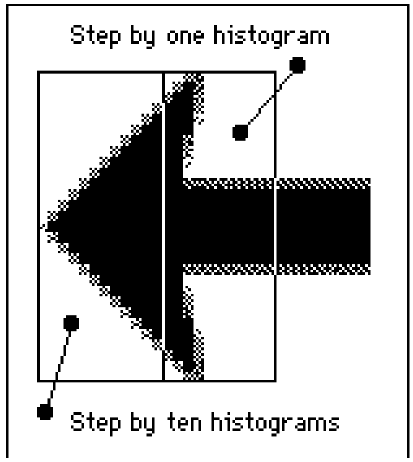
Also note that you can combine various data compression and data expansion on the same graph as appears in the example above. You can plot one data array at ten to one data compression, click on the one to ten data expansion, and plot another data array expanded, and click on the data expansion again (bring the plot back to a one to one ratio) and plot a third graph that way. It's up to

you to remember which plots were with which data compressions however. A point of advice: if you plot a graph and see now graph, one of two things might have happened. If the values are zero straight through the array, a line of data will be plotted along the bottom of the graph dictating zero values. This can be difficult to see sometimes. If you're sure no zero-line graph is plotted, then the reverse may be true. Your graph might be off the scale. Try setting Compress 10 to 1 and try it again. If you still get no result, then it's too great a value for you to see anymore. Sorry.
Show Histograms: Displays the Histograms dialog box:



When you first open this dialog box you won't see any histograms. In order to bring a histogram up, you must choose both a plotting ratio and a set of data. Then and only then will you see a histogram on the screen in front of you. The three columns of data and the percentage signs on the left side pertain to the first four buttons, which dictate the ratio of the data being plotted to the pixelation of the plot. If the percentage option is chosen, then the percentage markers on the left side tell you what percentage of the whole population are represented in each sixteenth section, so that 100% would be a bar reaching up to the 100% marker.
Prey Colors: color breakdown of the population
Poison Levels (one color): poison dispersal within a single color of prey
Poison Levels (all colors): poison dispersal across the entire prey population
Poison Tolerances: poison tolerance dispersal across the predator population

Memories (colors): number of memories of each particular color that the predator population has remembered.
The next thing on the right is a scroll bar. With this you can flip back and forth through histogram records. By clicking on different parts of the arrows at the ends you can jump by one histogram at a time or ten histograms at a time:



The frequency with which histogram data is gathered is set by the Histogram Collection Rate. Below the scroll bar is the number of the timestep to which the histogram you are presently viewing pertains.
The last thing you need to know how to use is the block of colors at the lower right. When you choose the Poison Levels (one color) option, this is how you dictate which color's poison dispersal you are viewing. Just click on a color and that color becomes the color of choice. Histograms of this type are plotted in the color of the prey that they represent.
Data vs. Time Collection Rate: Brings up a submenu from which you can choose how many timesteps must pass between the collection of data shown in the Data vs. Time dialog box. By setting this to higher values you can store information further back in time, but the further back you wish for it to record, the bigger the gaps between data collection will be. The first option, Every Timestep, gathers data every timestep, and so on. With this set to Every Timestep, the data recorded will stretch back 3000 timesteps.
Histogram Collection Rate: Brings up a submenu from which you can choose how many timsteps must pass between the collection of histograms shown in the Histogram dialog box. This is similar to the Data vs. Time Collection Rate.

Clicking Inside the Window and the Different Cursors:

In addition to the menu commands you have four other commands at your disposal. These consist of clicking on a single creature inside the view window with various modifier keys held down. The four commands are:

No Modifier Keys: cursor: Syringe 
Clicking on a creature with no modifier keys held down brings up a dialog box with information on an individual creature. This dialog box is explained below. If you click in a location where both a prey and a predator are present, you will first see the information for the prey and when you end that dialog box you will then see the information for the predator. If you click on a location with more than one prey or predator, you will only see the information for the first one in the linked list. If you have never executed a Randomize Ages command you will be looking at the oldest prey or predator, otherwise there is no significance to the one you see.

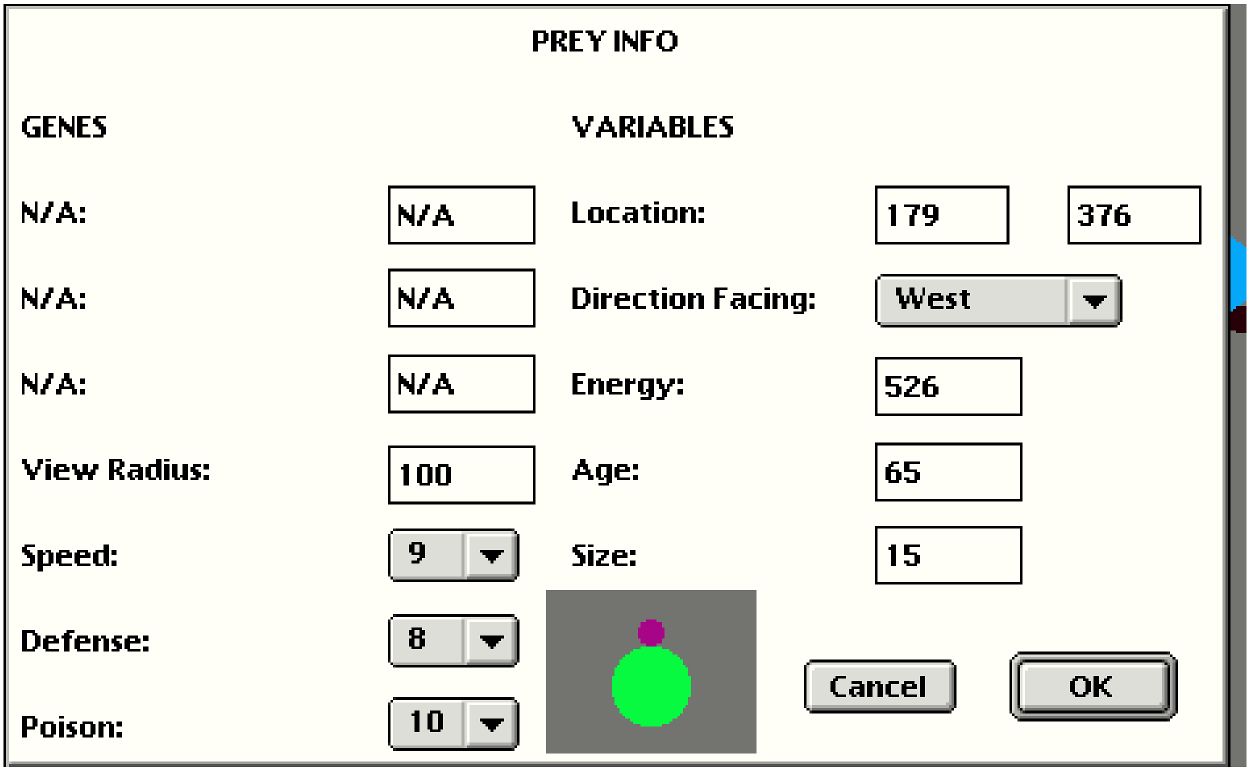
Option Key: cursor: Skull 
Kills individual creatures. This actually sets their energy to a value below one so that they will die on the next timestep, but the statistics won't record that they starved, only that they were killed.

Control Key: cursor: Unbroken test tube 
Creates an exact clone of the creature clicked on. The clone is different only in that its age is set to zero. The location of the clone is that of its parent.

Shift Key: cursor: Broken test tube 
Creates a mutated clone of the creature clicked on using the mutation rate set in the Settings dialog box. The clone's age is set to zero. The location of the clone is that of its parent.

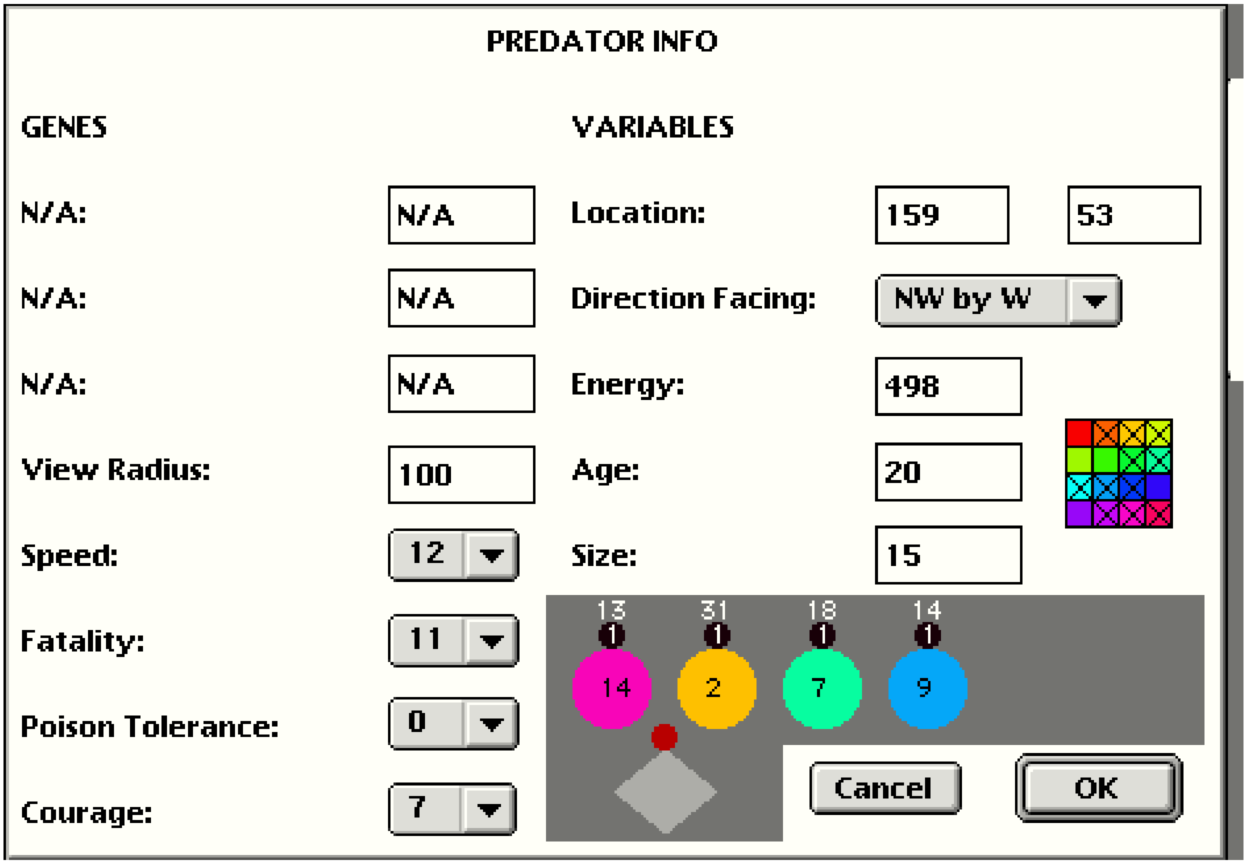
These last two commands are fairly obsolete because by the time you kill a an organism the program will automatically saturate the populations with clones of other organisms. If you really want to clone a particular organism, pause the program to do so.

The Prey Information Dialog Box:



Several genes from version one are no longer used, as marked by N/A. The rest if fairly obvious. The picture will show both the present size and color, including head color of the prey. From this dialog box you can edit the prey to any liking you desire.

The Predator Information Dialog Box:



Several genes from version one are no longer used, as marked by N/A. The rest is fairly obvious. The picture of the predator show size, head color due to fatality, and body color due to number of memories. The pictures of the prey represent memories of prey that have been eaten. Because these are memories and not actual prey, the heads will never show the blue aspect of a prey's head due to defense. Predators don't remember that. The memories' heads will show red due to poison however. The number inside the head corresponds to the exact poison value of that particular memory. The number inside the memory shows precisely which color it is. Some colors, especially shades of green, are a little difficult to distinguish. The colors are numbered left to right, top to bottom according the square on the right, or the same identical square in the histogram window. Basically, pure red is color number zero. The square on the right side has Xs over some colors and not other colors. Based on the colors of all the predator's memories, and the courage of the predator (which determines how many colors on either side of a memory are acceptable), this X chart marks out the colors of prey that this predator will refuse to follow or eat. In the example above, the predator will eat color numbers 0, 4, 5, 11, and 12.

Prey:

Prey have several genotypic and phenotypic traits. They have several genes stored simply as numerical values, which mostly affect physical traits and energy requirements. The also have genes that determine the weights in their neural network brains.. This is explained below.

Prey lose and gain energy constantly. The lose energy for moving as well as the some of their phenotypic traits. The genetic traits that cost a prey energy are poison and defense. Energy is evenly distributed to all prey continuously.

The cloning of prey is pretty straightforward. An exact clone is created and then various genes are mutated slightly.

The Numerical Genes:

View Radius: The distance (in pixels at the x8 zoom setting) out to which the prey can see. Any plants or prey within the 180 degree semicircle defined by 90 degrees to the left and 90 degrees to the right of the prey's facing direction within the distance set by this gene are subject to visibility.

Speed: The number of pixels (in the x8 zoom setting) that the prey moves in any timestep that it decides to go forward. The value of this gene is subtracted from the prey's energy every time it goes forward .

Defense: The defensive capability of the prey. Prey/predator confrontation is detailed below. The value of this gene is subtracted from the prey's energy every timestep.

Poison: The strength of the poison that the prey possesses. Poison effects on predators are detailed below. The value of this gene is subtracted from the prey's energy every timestep.

Predators:

Predators are the most mentally complex life forms in Copy Cat. They have several genes stored simply as numerical values, which mostly affect physical traits and energy requirements. They also have genes that determine the weights in their neural network brains. The brains of both prey and predators are described together below because of their similarity. Predators have the capability of remembering prey they have eaten. They will only remember prey that were more poisonous than their poison tolerance gene allows and they only remember information pertinent to such a memory: what the prey looked liked and how poisonous it was. They also keep a tally of how many times they have encountered prey matching a particular memory, but they don't actually make use of this information. It is saved solely for your informational benefit.

Predator feeding is described in Prey/Predator Confrontation below.

Predator mating is exactly like prey mating.

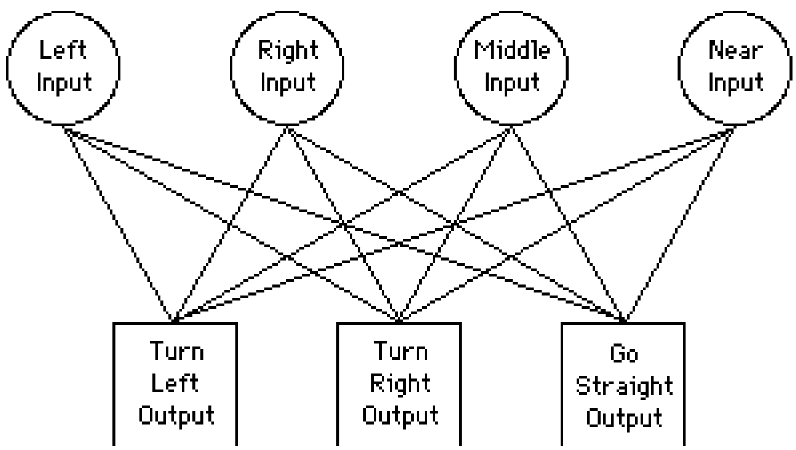
The Numerical Genes:

View Radius: The distance (in pixels at the x8 zoom setting) out to which the predator can see. Any plants or predator within the 180 degree semicircle defined by 90 degrees to the left and 90 degrees to the right of the predator's facing direction within the distance set by this gene are subject to visibility.

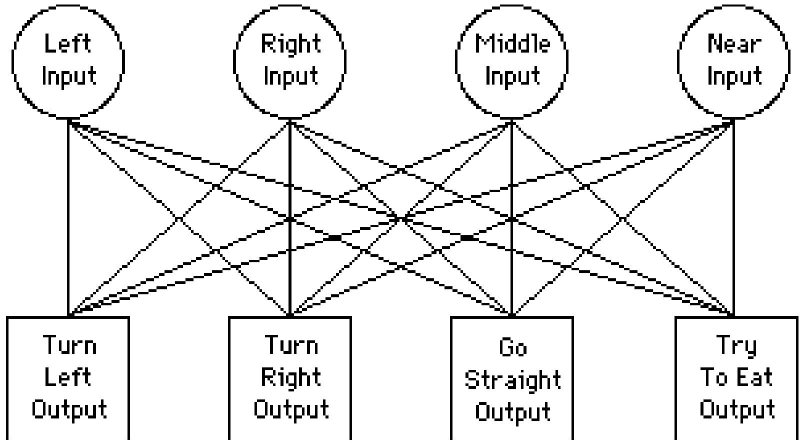
Speed: The number of pixels (in the x8 zoom setting) that the predator moves in any timestep that it decides to go forward. The value of this gene is subtracted from the predator's energy every time it goes forward.
Fatality: The offensive capability of the predator. Predator/predator confrontation is detailed below. The value of this gene is subtracted from the predator's energy every timestep.
Poison Tolerance: The strength of poison that the predator is willing to eat. Eating prey that are poisonous but not as poisonous as this gene dictates will not be remembered as bad to eat.
Courage: The degree of similarity a prey must have to a memory for a predator to think it is too risky to eat. The higher this value, the more closely a prey must match a memory in order to be avoided.

Prey and Predator Brains:

Prey and predators have very similar brains. Predators are only attracted to things, and prey are only repelled by things, because predators have nothing to avoid and prey have nothing to appeal to. However, the premise of the brains is the same. The higher the value for a certain direction, the stronger that attractor or detractor for the predator or prey. The brains consist of numbers that represent weights of various synapses in the brain. All weights range from zero to fifteen. So, a prey's brain looks like this:



And predator's brain looks like this:



The following algorithm determines if an object is to the left, the right, straightforward, or nearby:

if (object is within 15 pixels)
        set near input
else if (object is greater than 22.5 degrees to left)
        set left input
else if (object is greater than 22.5 degrees to right)
        set right input
else set directly forward input

The amount sent to an input for a given object depends on its distance. If an object is nearby, the nearby input receives an addition of the value of the view radius of the creature. Otherwise, if an object is in the left, forward, or right visual areas, then the appropriate input receives a value of the creature's view radius minus the distance of the object, so that closer objects send strong inputs.

The brain itself is an extremely simple neural network with no hidden layers. Every input is connected to every output, but the synapses can have a weight of zero essentially negating those connections. Each input is multiplied by each of its synapses and added to an accumulator at each output, and the output with the greatest sum is executed that timestep.

In the Settings dialog box there is an option called Born Smart. This engineers the brains of all creatures such that they are identical. The brain DNA is not actually engineered, but the brain itself is. The weights in an engineered brain are:

Prey:

Input Neuron	Output Neuron	Prey Weight
[Bad To Left]	[TurnLeft]	0
[Bad To Right]	[TurnLeft]	3

[Bad To Middle]	[TurnLeft]	3
[Bad To Near]	[TurnLeft]	0
[Bad To Left]	[TurnRight]	3
[Bad To Right]	[TurnRight]	0
[Bad To Middle]	[TurnRight]	3
[Bad To Near]	[TurnRight]	0
[Bad To Left]	[Forward]	0
[Bad To Right]	[Forward]	0
[Bad To Middle]	[Forward]	0
[Bad To Near]	[Forward]	15

Predator:

Input Neuron	Output Neuron	Prey Weight
[Good To Left]	[TurnLeft]	1
[Good To Right]	[TurnLeft]	0
[Good To Middle]	[TurnLeft]	0
[Good To Near]	[TurnLeft]	0
[Good To Left]	[TurnRight]	0
[Good To Right]	[TurnRight]	1
[Good To Middle]	[TurnRight]	0
[Good To Near]	[TurnRight]	0
[Good To Left]	[Forward]	0
[Good To Right]	[Forward]	0
[Good To Middle]	[Forward]	2
[Good To Near]	[Forward]	0
[Good To Left]	[Eat]	0
[Good To Right]	[Eat]	0
[Good To Middle]	[Eat]	0
[Good To Near]	[Eat]	14

Predator Memory:

The effects of a predator's memory take place at the input level. If a predator decides that a prey it is looking at is too similar to a poisonous prey it remembers once eating, then the predator becomes essentially blind to that prey. When a predator spots a prey, it looks through every one of its memories comparing the prey it sees to the memories. If the prey is a close enough match to any of the memories the predator avoids it. Since predators only remember prey that are more poisonous than their poison tolerance, it is thus avoiding prey that it thinks might be too poisonous for its poison tolerance. To assess the similarity of a prey and a memory a comparison is made of the distance between the colors on the spectrum. If this difference is less than the courage of the predator, then the similarity is too close for the predator to risk and the predator promptly ignores the prey in question by not adding its location to any inputs. If the difference is greater than or equal to the predator's courage, then it assumed that the prey and the memory are different enough to not bother making an issue of and the predator will go ahead and add the prey's location to its inputs.

Prey/Predator Confrontation:

Prey and predators must deal with one another on a regular basis. In the event that a predator is close enough to attack a prey (meaning the predator got a Near input from a prey which means a prey is within fifteen pixels of the predator) and the predator chooses to attack that prey, then the fight must be evaluated. If the predator loses the fight, the prey escapes completely unharmed, otherwise, the prey is killed and eaten. There are no wounding injuries in Copy Cat. First, and only if the prey went forward that timestep (they are running away), the speed of the two creatures is compared (did the predator even manage to catch the prey) using the following table (if the prey did not go forward, then the predator won this test (caught the prey) and goes on to the next test):

Prey Speed Compared To Predator Speed	Probability of Catching Prey
+2 or greater	0 (impossible for predator)
+1	1 in 3
0 (they have equal speed)	1 in 2
-1	2 in 3
-2 or less	1 (certain for predator)

Second, and only if the predator won the first test (caught the prey), an identical type of comparison is made for the prey's defense and the predator's fatality (The prey might be able to fend off the predator):

Prey Defense Compared To Predator Fatality	Probability of Killing Prey
+2 or greater	0 (impossible for predator)
+1	1 in 3
0 (they have equal speed)	1 in 2
-1	2 in 3
-2 or less	1 (certain for predator)

If the attack passes the first test and fails the second test, then the prey has escaped, not due to running away but do to an armored body or defensive weapons. If both tests pass the predator wins, kills, and eats the prey. From the prey the predator will gain the full energy that the prey possessed and lose the value of the prey's poison gene times the Poison Strength as set in the Settings dialog box. If the predator's energy after the attack is less than one then the predator is said to have died of poisoning. If the predator survives its meal, but the prey was poisonous, but the poison value of the prey was less or equal to the poison tolerance gene of the predator, the predator doesn't notice the poison. However, in the rare event that a predator eats a poisonous prey and lives to tell about it, but the poison value of the prey was greater than the poison tolerance of the predator, then the predator will remember what the prey looked like and try to avoid its kind in the future. New memories usually replace the least used memory when the predator's memory slots are all full. The recency of a memory works a little in its favor too however. For example, if Maximum Number of Memories is set to two, then a predator can remember two prey. The third one it remembers takes the place of the memory with the least number of uses avoiding other prey, which is essentially forgotten. The details of predator memory are detailed below.

Energy:

Energy is fundamental to Copy Cat, but up until now I have not described in detail how mana works into Copy Cat. In the Settings dialog box you may set the mana for both prey and predators. This amount of energy is administered to every single prey and predator every single timestep. This is the only form of energy input for prey, and other than eating prey, it is the only form of energy input for predators. Don't set this too low, or everything will die out. You can determine the threshold by turning on the display of energy on prey and predators' backs in the Draw Prey Data and Draw Predator Data options of the View menu.

Culling and Population Saturation:

The prey and predator populations are continually cropped and saturated. According to the settings you give the program in the Settings dialog box, occasionally a couple of the weakest prey and predators (according to the energy variable) will be randomly chosen and killed. At the end of every single timestep, the program compares the populations against the maximum you set in the Settings dialog box. If the population is not maxed out, either due to culling, being eaten, starvation, poison, or you're own killing by option clicking individuals, the strongest individuals are cloned, their energy being divided between the parent and the clone, continuously until the population is maxed out, before the program is allowed to continue running.

Evidence for the occurs of mimicry in Copy Cat

Copy Cat is designed and written to evolve mimicry, but how does one know if mimicry is actually occurring? Well, I have designed the Histograms dialog box to provide information that would be necessary in order to suggest evidence for or against the existence of mimicry in any biological system, such as Copy Cat.

First, before you begin experimenting with the Settings dialog box, the Configure Variables dialog box, or any other variables such as data and histogram collection rates, I suggest you run a simulation with the default settings. These settings are values that I have run again and again and can confidently say will evolve mimicry nearly every time the program is run. If you start up Copy Cat and choose a new simulation right off, then you should be in business.

Leave the program running for at least one thousand timesteps and open the Histograms dialog box. As explained above, you will not see a histogram of any data at first. To begin, click in the first compression button (ten to one) and then click on prey colors. You will see the distribution of colors, which in all likelihood will be an approximate bell-shaped curve centered around the most prominent color. Now, to see if predator memories are shaping the prey color distribution, click on the Memories (colors) button and you will see the distribution of predator memories of particular colors. Hopefully, the two histograms you have so far looked at will both have bell-shaped curves centered at nearly the same color. The memories histogram will almost certainly be a sloppier curve, but the data should nonetheless suggest the a correlation of some sort. Now you have something to work with. From here several things can be noticed.

First, try clicking on the Poison Levels (all colors) button. You will see that a vast majority of the prey have a poison value of zero and the curve slants down sharply, with only an occasional prey of poison value four or five, possibly six. What this suggests is that most of the population is getting away with something. Most of the prey aren't actually poisonous. If you click on the Poison Tolerance button for a moment and see the Poison Tolerance distribution of the predators, you will discover that almost all the predators can indeed tolerate a poison value of zero. Zero is the lowest possible value but that's beside the point. The interesting fact, is that predators could eat those prey safely (and in so doing, gain energy over fellow predators increasing their cloning capacity and spreading their genes further) but their memories associating poison with that color prevent them from doing so. The occasional predator will be born that does not know a particular color of prey should be avoided, and that predator may very well eat several nonpoisonous prey before it encounters one of the rare poisonous ones. But then it will instantly learn and remember that those prey have become dangerous and must be avoided. If you click on the Poison Levels (all colors) button once again, you'll see once more that a very small amount of the prey population needs to be poisonous in order for the entire prey population to benefit from that poison. With most of the prey not making any poison, they can expend energy on other things. In Copy Cat, you only get cloned if you have more energy than any other prey on the entire map (the same is true for predators incidentally). To preserve energy and basically take advantage of your poisonous neighbors by not making poison gives you enough energy to get cloned once in a while.

Now here's something really interesting. Click on the Prey Colors button again and figure out where the center of the bell-shaped curve is. One or maybe two colors should dominate the population. Remember what color is the most prevalent and click on the last remaining button, the Prey Poison (one color) button. Then click on the Percentage compression button, the last of the four compression buttons. If you starting clicking in the various colored boxes picking different color populations to study, you notice the most peculiar thing. The color of prey that is the most widely represented in the population has a higher percentage of its prey with less poison. As you click on colors getting towards the periphery of the bell-shaped curve, the distribution of poison within a single color is more fairly spread out. What does this mean? I have come up with a theory that I like quite a bit. Basically I believe it comes down to this. If a prey "chooses", genetically speaking, to appear different from the majority of the population, then, statistically, fewer predators will have memories that prevent that prey from being eaten. Thus, more prey of that color will have to be poisonous in order to maintain the population. Remember when I said the predator that is occasionally born without the appropriate memories will statistically probably encounter several nonpoisonous prey before it eats a poisonous one and makes a memory to avoid that color? Well, if more predators are munching their way through a prey's color population, then statistically, not as many nonpoisonous prey will suffer if there's a higher chance that the predator will indeed encounter a poisonous prey and start avoiding that color as quickly as possible. Only after a prey color population is confidently secure from the maws of predators can some prey starting being less poisonous without getting eaten. The predators must avoid a highly poisonous population first, and then the prey actually have a blanket of safety and can start losing their poison to a great degree. Just a couple of poisonous prey is enough to reprogram the occasional odd predator that wanders onto the map not knowing that prey of a certain color should be avoided. There is probably a very mathematical statistical correlation between the standard deviation of a color from the most dominant color on the map and the distribution of poison within that color, and similarly, more interestingly, there is probably a very mathematical statistical correlation between a color population's poison distribution, and that color's representation in the predators' memories. That is to say, not only does deviation from the norm colorwise dictate more poison, but deviation from the norm memorywise is probably what is driving the greater amount of poison in that less represented color.

Speed Tips:

There are lots of things you can do to speed up Copy Cat:

2. Turn off Draw Prey/Predator Views.
3. Turn off Draw Prey/Predator Data.
4. Set Zoom to high powers (x8 is best of course). This way to program is updating as few individuals as possible.
5. Use the update every X timesteps feature.
6. Turn off Screen Updating (this is a biggy. Do this every time you leave the program running for a long time and you're not watching it.).
7. Put Copy Cat in the background. I don't why this works, but it literally doubles the speed, almost precisely as a matter of fact. I think it has something to do with the program looking for events; so bring the Finder to the front when you leave Copy Cat running for a long time.

In short, what I do whenever I leave the program running over night is turn off Screen Updating and then bring the Finder to the front. So long as Screen Updating is turned off, none of the other speed tips have any effect because they relate to drawing the screen.

The Future of Copy Cat:

Where is Copy Cat headed? Well I don't really know. In my attempts to reproduce mimicry with Copy Cat version 1.0 I failed, for a number of reasons that I am aware of and probably for several reasons that I am not familiar with yet. Version 2.0 confidently evolves mimicry however, much to my vast relief. If I wanted to elaborate on the program now that it works, there are plenty of things that could be done:

1. Open/Save/Close feature for a simulation so you can stop at any point and reload your simulation later on.
3. A way to see and edit the brain of prey and predators.
4. A much more complex way-too-awesome neural network brain.
5. Spicier environment with more varying features (impassable water areas for geographic isolation would seriously shape the evolution).
6. More creature senses. Possibly olfactory sensation for mate attraction (if I put sexual reproduction back into the program) (and mate mimicry by predators). Possibly taste and then I could let prey react to poisonous plants and bad tasting plants the same way predators react to prey at the moment. The sky is the limit.
7. More creature actions. Face it, right now it's pretty bland. You either turn left, turn right, go forward, or eat. Even running away or chasing is basically just moving around. It would be interesting to create melee type combat

between prey and predators or possibly between potential mates battling for a mate with really good genes.

8. Sex. Both sexual reproduction (which took out of version 1.0), and different sexes, which might possibly have different roles.
9. Floating windows so the program continues to run behind the various dialog boxes.
10. Ability to design your own brains that can then be used for the Born Smart option.
11. And the list goes on...

Keith Wiley
June 11th, 1996

Email: keithw@wam.umd.edu
World Wide Web: http://www.wam.umd.edu/~keithw

Snail Mail:
Keith Wiley
102 Keith Rd.
Carrboro, NC
27510
USA