

# AQUARIUS



Game Design Document  
Version 1.0  
Team Thalassic

# Contents

<b>1</b>	<b>Document Conventions</b>	<b>5</b>
1.1	Highlighted Text and other Caveats . . . . .	5
1.2	Square Braces . . . . .	5
1.3	Tooltips . . . . .	5
<b>2</b>	<b>God Mode</b>	<b>6</b>
2.1	Point of View (POV) . . . . .	6
2.2	Switching Between Movement Mode and GUI Mode . . . . .	6
2.3	Movement Mode . . . . .	6
2.4	GUI Mode . . . . .	7
2.5	Pull-Down MenuBar . . . . .	8
2.6	Edit Terrain Falloff Value . . . . .	12
2.7	Fish Sidebar . . . . .	13
2.8	Fish Sidebar . . . . .	14
2.9	Object Sidebar . . . . .	16
<b>3</b>	<b>Objects</b>	<b>19</b>
<b>4</b>	<b>Fish Mode</b>	<b>21</b>
4.1	Point of View (POV) . . . . .	21
4.2	Fish Statistics . . . . .	22
4.3	Movement Mode . . . . .	22
4.4	GUI Mode . . . . .	23
<b>5</b>	<b>Fish Statistics</b>	<b>25</b>
5.1	Fish Actions . . . . .	25
5.1.1	Mouth . . . . .	25
5.1.2	Mating . . . . .	27
5.1.3	Body . . . . .	28
<b>6</b>	<b>Mating and Evolution of Species</b>	<b>29</b>
<b>7</b>	<b>Fish Editing Mode</b>	<b>30</b>
7.1	Point of View (POV) . . . . .	30
7.2	Pull-Down MenuBar . . . . .	31
7.3	Fish Statistics . . . . .	32
7.4	AI Architecture Text . . . . .	32
7.5	AI Unit Grid . . . . .	32
7.6	Action Decision Node SideBar . . . . .	33
7.7	Fitness Function SideBar . . . . .	35
7.8	Data Filter SideBar . . . . .	35
7.9	Connection Tool . . . . .	36
7.10	3D Rotating Fish . . . . .	37

<b>8</b>	<b>Fish Attributes</b>	<b>38</b>
8.1	Base Attributes . . . . .	38
8.1.1	Metabolism (Meta) . . . . .	39
8.1.2	Size . . . . .	39
8.1.3	Eyes . . . . .	39
8.1.4	Teeth/Baleen (TB) . . . . .	40
8.1.5	Armor . . . . .	40
8.2	Derived Attributes . . . . .	40
8.2.1	Base Energy Consumption Universal Multiplier (BECUM) . . . . .	40
8.2.2	Attack Strength . . . . .	41
8.2.3	Reaction Time . . . . .	41
8.2.4	Max Speed . . . . .	41
8.2.5	Suckability . . . . .	41
8.2.6	Life Span . . . . .	41
8.3	Reactive Attributes . . . . .	41
8.3.1	Stomach . . . . .	42
8.3.2	Health . . . . .	42
8.3.3	Energy Level . . . . .	42
8.3.4	Mating Inclination . . . . .	43
8.4	Other Fish Formulae . . . . .	43
<b>9</b>	<b>Offline Mode</b>	<b>45</b>
<b>10</b>	<b>Game Setup</b>	<b>46</b>
10.1	Pull-Down MenuBar . . . . .	46
<b>11</b>	<b>Title Screen</b>	<b>51</b>
<b>12</b>	<b>Networking</b>	<b>52</b>
<b>13</b>	<b>Graphics</b>	<b>53</b>
<b>14</b>	<b>Sound</b>	<b>54</b>
<b>15</b>	<b>3D Physics</b>	<b>55</b>
<b>16</b>	<b>Comparative Analysis</b>	<b>56</b>
<b>17</b>	<b>Marketing Analysis</b>	<b>56</b>
<b>A</b>	<b>GUI Primitives</b>	<b>58</b>
<b>B</b>	<b>Game Actions</b>	<b>60</b>
<b>C</b>	<b>General Appendix</b>	<b>61</b>
<b>D</b>	<b>Team Sign Off Sheet</b>	<b>63</b>

## High Concept

You coexist with a diverse aquatic population as the omniscient ruler of an Aquarium, and can wield your power over their evolution as you see fit.

## Project Overview

Aquarius is an open-ended dynamically-learning AI game where the player can observe and experiment with the aquatic dwellers of the Aquarium. While the player has a significant amount of control over the Aquarium's environment, and can actually take control of the Aquarium's inhabitants and do everything that they can do, the inhabitants themselves will learn to behave differently over the course of their lives, as well as interbreed amongst themselves, thereby giving rise to their genetic evolution.

## Target Demographic

Our game does not intentionally exclude, or cater to any one demographic group. It lacks graphic violence which some find objectionable and should garner appeal in response to that. It also has generic appeal for being as close to unique as is possible in this modern world. However, there are certain areas that will find special interest in this game, as well as those that will not enjoy our game.

## Tinkering

Population segments that will find the greatest appeal in this game will be tinkerers. At heart everybody has a little tinkerer itching to see what happens when an AI connection is changed or a statistic gets decremented. The best way to please a tinkerer is create a flexible design and flexibility was paramount during the design of this game. Almost every fish portion is tweakable and extendable which will draw even the most picky tinker back to the computer for more.

## AI

Another segment of the population that will have special interest in this game are those fascinated with Artificial Intelligence. Possibly, this segment would have greater education then the average gamer or even degrees in Computer Science. However, AI fascination is not limited to people with an Advanced Computer Science degree, everybody in today's modern world finds allure at the chance of a computer that can learn. Our game design holds the prospect of creating a system that can both learn and evolve, creating a massive genetic algorithm unfolding before the player's eyes.

## **Interactivity**

The Ex-DigiPen Professor and Great Atari designer Chris Crawford has said that a game designer's job is to allow utmost flexibility in gameplay. In his mind the Video Game Industry's only great tool against other forms of media was interactivity. His famous battle cry was, "What can the player do!?!?" To answer Crawford's question players can do anything in our game. If the design is taken to its zenith every item in the tank will be dynamic, touchable, and changeable. This dynamism creates a superb example of digital interaction empowered to bring pleasure to almost any player.

## **Non-appeal**

There are certain segments of the populace that will have little interest in this title. Players under the age 14 will likely find little interest in a game lacking Mario or Pokémon. Also, players looking for a quick fix of guns, graphic violence and sex will find disappointment in our title. Our game contains no single element that would disenchant women from playing it, but it is not a Holy Grail that will draw all women into the gamer's world. However, with all of those negative statements made, our game will still intrigue a large enough demographic to make it well worth the resources of development.

## **Target Platform**

### **Minimum System Requirements**

<b>CPU</b>	Pentium III 800 MHz
<b>RAM</b>	64 MB
<b>HD Space</b>	100 MB
<b>Video Card</b>	TNT 2 or equivalent

### **Recomended System Requirements**

<b>CPU</b>	Pentium IV 1.4 GHz
<b>RAM</b>	128 MB
<b>HD Space</b>	200 MB
<b>Video Card</b>	GeForce2 or equivalent

# 1 Document Conventions

## 1.1 Highlighted Text and other Caveats

**Highlighted Text** refers to a Higher Goal. When un-Highlighted text is followed by contradictory **Highlighted Text**, the functionality described by the **Highlighted Text** will eventually supersede the functionality described by the un-Highlighted text. While the Technical Design Document and its corresponding Timeline plan for implementing everything described in this document, all **Highlighted Text** can be cut from the design at any time for any reason.

Note that the Project Overview, Sample Game Turns and Introductory Paragraphs are meant to convey the general idea of each section, and are not to be construed as “hard features” that are definitely to be implemented in the game. Refer to the color coded text succeeding the Introductory Paragraphs for definitive information on which game features are absolutely required and which are considered Higher Goals.

## 1.2 Square Braces

Many in-game actions are enacted by the Player pressing a key. All of these actions can be mapped to any key (where the Left, Middle, and Right Mouse Buttons are considered to be keys). To reflect this, any references to a key that performs an action, such as Switching Between Movement Mode and GUI Mode, does so in the following manner:

In General:     [*Player Action*]

For Example:    [GUI Visibility]

## 1.3 Tooltips

Aquarius follows standard constructs in many games and attaches tooltips to the mouse cursor whenever it is appropriate. However, Aquarius adds dimension to these tooltips by using Short Tooltips and Long Tooltips. A Short Tooltip appears very quickly and briefly gives the name of the item that the mouse is current hovering over. Long Tooltips activate much slower and go into far greater detail about the item in question.

## 2 God Mode

God Mode provides the player with all the abilities an Aquarium God would expect. He is able to move about freely (unseen by the Fish of the Aquarium) and ethereally (passing right through the Fish), manipulate the environment, toss more Fish and Objects into the Aquarium or remove them, and even save the Aquarium, so He can return sometime later.

### 2.1 Point of View (POV)

- The Aquarium and the Fish, Terrain and Objects that populate it are rendered in realtime 3D from the player's current position and orientation.

### 2.2 Switching Between Movement Mode and GUI Mode

- There are two mutually exclusive methods:
  1. Toggle Switch  
Striking [GUI Visibility] toggles from Movement Mode to GUI Mode and back.
  2. Quick Command Switch  
Holding down [GUI Visibility] switches to GUI Mode. Releasing [GUI Visibility] returns to Movement Mode.

### 2.3 Movement Mode

*The player enters the Aquarium for the first time. Floating and weightless, he shifts his gaze around his watery environs. Cool, rolling sands extend as far as the eye can see, and a school of multifarious Fish swims smoothly by, each looking a little different from the other. Moving briskly, and without apparent mass, the player glides toward the vertebrates.*

- There are no graphics (Mouse Cursor or otherwise) overlaid on the POV.
- The Mouse's x-axis movement will be directly mapped to rotating about the POV's Y Axis (i.e. alters the POVs yaw).
- The Mouse's y-axis movement will be directly mapped to rotating about the POV's X Axis (i.e. alters the POVs pitch). Note that this movement can be inverted, at the player's option.
- [Move Forward] and [Move Backward] will cause the POV to move forward or backward, respectively, along the POV's Z axis at a brisk, constant velocity.
- [Left Strafe] and [Right Strafe] allow the POV to slide along the POV's X axis. As in [Move Forward] and [Move Backward] strafing velocity is always constant.

## 2.4 GUI Mode

*Coming to a sudden, gossamery halt, the player thoughtfully regards the congregation of Fish.*

- Upon switching from Movement Mode to GUI Mode, the POV becomes locked in the last position and orientation it had last assumed from Movement Mode.
- A Mouse Cursor (a crosshair) is overlaid on the POV, and tracks Mouse movement as in Windows.
- 2D Mouse Cursor Movement allows the Player to Highlight Fish and Objects currently visible in the 3D viewport in the following manner:
  - A ray is projected through the Mouse Cursor's (x,y) screen position (a pixel) into the Aquarium. The first Fish or Object the ray strikes is Orange Highlighted.
  - Pressing [Primary Action] causes the Fish or Object that is currently Orange Highlighted to become Red Highlighted, and the Fish or Object (if any) that was previously Red Highlighted is no longer Red Highlighted.
- Pressing [Secondary Action] while the cursor is hovering on a Fish causes the player to switch to Fish Mode  $\longrightarrow$  Movement Mode as that Fish. In other words, the player “possesses” the Fish he pressed [Secondary Action] on.
- 3D Mouse Cursor Movement allows the Player to specify a location and orientation in the 3D Aquarium in the following manner:
  - The Mouse Cursor's (x,y) screen position (a pixel) is projected onto the x-y plane of the height map that defines the Terrain. The Terrain Square that the projected location lands in is Red Highlighted.
  - If the Mouse Cursor's (z) position is less than the Red Highlighted Terrain Square's (z) position, then the Mouse Cursor's (z) becomes equal to the Red Highlighted Terrain Square's (z) position. Otherwise the Mouse Cursor's (z) position retains its previous value.
  - The Local-X-Axis Orientation and Local-Y-Axis Orientation both retain their previous values.
    - \* At the start of a game, the Mouse Cursor's (z) position is equal to 0.
    - \* At the start of a game, the Local-X-Axis Orientation and the Local-Y-Axis Orientation are both equal to 0.

- The Mouse Cursor’s (z) position can be altered by holding down [Alter Z Position]. While [Alter Z Position] is depressed, the Mouse Cursor’s (z) position increases and decreases with the y-axis movement of the Mouse. While [Alter Z Position] is depressed, the on screen Mouse Cursor (a crosshair) does not move or reflect the Mouse’s movement in any way.
- The Local-X-Axis Orientation and Local-Y-Axis Orientation of the Mouse Cursor can be altered by holding down [Alter Orientation]. While [Alter Orientation] is depressed, the Mouse Cursor alters the value of Local-Y-Axis Orientation with the the x axis movement of the Mouse, and the Mouse Cursor alters the value of Local X-Axis Orientation with the y-axis movement of the Mouse. While [Alter Orientation] is depressed, the on screen Mouse Cursor (a crosshair) does not move or reflect the Mouse’s movement in any way.
- Note that there is no explicit graphic to show what (z) position and orientation the Mouse Cursor has currently specified; however, when Objects or Fish are being placed in the Aquarium, that Object or Fish will be Red Highlighted and will update its position and orientation in accordance with the 3D Mouse Cursor Movement scheme until the player places the Object or Fish.

## 2.5 Pull-Down MenuBar

*On a whim, the player causes the nearby Aquarium bottom to rise up in a large cliff of coral, blocking the Fish’s forward movement. Sensing the sudden change in terrain, the leading Fish veer off their intended course, and circle around to swim back toward the player.*

*Thinking the new coral edifice a bit plain looking, the player begins to riddle the cliff with small pockmarks. This done, the player notices that a few of the smallest Fish have begun to swim into the pockmarks perhaps seeking to hide themselves from their larger brethren. Out of curiosity, the player causes the pockmarks to jump outward into small ridges, and watches as the little Fish are harmlessly bounced up from the terrain some acting skittish even as others seem unconcerned.*

*Pleased with his work, the player saves his Aquarium, in case things take a turn for the worse.*

- Located across the top of screen
- Some Menu Choices spawn Dialog Boxes, which all consist of Text, Text Fields, and PushButtons (“OK”, “Cancel”). Additional Dialog Box features are noted below on a case-by-case basis.

Game

Pause

- Long Tooltip (Paused State): *Unpauses the Aquarium.*

- Long Tooltip (Unpaused State): *Pauses the Aquarium.*
- A Toggle Menu Item
- Toggles between “paused” and “unpaused” states

#### Save Aquarium To File

- Long Tooltip: *Saves the Aquarium’s current state in a file.*
- Spawn “Save Aquarium to File Dialog Box,” which is a “Save File Forced Pathname Dialog Box”
- “Save Aquarium to File Dialog Box” is a “Standard Windows Construct”

#### Load Aquarium From File

- Long Tooltip: *Quits and discards the current Aquarium, and launches an Aquarium that has been previously saved to file.*
- Spawn “Sanity Prompt Dialog Box”
  - \* Spawn “Load Aquarium From File Dialog Box,” which is a “Load File Forced Pathname Dialog Box”.
  - \* “Load Aquarium From File Dialog Box” is a “Standard Windows Construct”
  - \* When a new Aquarium is loaded it is immediately launched with
    - The player in God Mode  $\longrightarrow$  Movement Mode
    - The game toggled to “Paused”

#### Go to Offline Mode

- Long Tooltip: *Switches to a statistic-only display. No interaction with the Aquarium will be possible until you return to God Mode. The activity in the Aquarium will be accelerated while you are in Offline Mode.*
- Spawn “Sanity Prompt Dialog Box”

#### Join Server Game

- Long Tooltip: *Quits and discards the current Aquarium, and join another player’s Aquarium.*
- Spawn “Sanity Prompt Dialog Box”
  - \* Spawn “Join Server Dialog Box,” which consists of:
    - Text: *Enter the IP Address of the machine which is running the Aquarium you wish to join.*
    - Text Field (for the Server IP Address) and PushButton “OK”

- Entering the IP Address of a valid Server and pressing [Primary Action] over “OK” causes the player to join that Game as a Client.

#### Return to Title Screen

- Long Tooltip: *Quits and discards the current Aquarium, and return to the splash screen.*
- Spawn “Sanity Prompt Dialog Box”

#### Quit Game

- Long Tooltip: *Quits and discards the current Aquarium, and exits the application.*
- Spawn “Sanity Prompt Dialog Box”

### Options

#### Game

Behaves as per “Game Options” – See Game Setup

#### System

Behaves as per “System Options” -- See Game Setup

### Aquarium

#### Select Area Tool

- Long Tooltip (Unchecked State): *Activate Grid-Based Terrain Editing. By pressing [Primary Action], you will be able to select a square on the Terrain, and then make that square higher or lower by holding down [Alter Z Position] and moving the Mouse away from you, or toward you, respectively. Alternately, you will be able hold down the [Primary Action] on a square of the Terrain, and then move the Mouse to select a rectangular region of many Terrain squares. When you’ve selected the region with the right dimensions, release [Primary Action]. Then hold down [Alter Z Position] and move the Mouse away from you, or toward you, to make all of the selected region of Terrain squares become higher or lower, respectively. Use the Edit Terrain Falloff value to make the region of Terrain squares have a steeper, or shallower slope.*
- Long Tooltip (Checked State): *Deactivate Grid-Based Terrain Editing. Pressing [Primary Action] will select Fish and Objects in the Aquarium, not Terrain squares.*

*Since Grid-Based Tile Editing is currently, activated, by pressing [Primary Action], you will be able to select a square on*

*the Terrain, and then make that square higher or lower by holding down [Alter Z Position] and moving the Mouse away from you, or toward you, respectively. Alternately, you will be able hold down [Primary Action] on a square of the Terrain, and then move the Mouse to select a rectangular region of many Terrain squares. When you've selected the region with the right dimensions, release [Primary Action]. Then hold down [Alter Z Position] and move the Mouse away from you, or toward you, to make all of the selected region of Terrain squares become higher or lower, respectively. Use the Edit Terrain Falloff value to make the region of Terrain squares have a steeper, or shallower slope.*

- A Check Mark denotes activation (on activation, this Menu Choice de-activates all other Aquarium Terrain Editing Tool Menu Choices)
- 3D Mouse Cursor Movement allows the Player to Red Highlight a single Terrain Square (with a single [Primary Action]), or Red Highlight a rectangular area of Terrain Squares (by specifying the Origin Terrain Square and the Extent Terrain Square with a [Primary Action] and Drag). Once one or more Terrain squares are selected, pressing [Alter Z Position] causes the raising and lowering of the middle Terrain Square (or Terrain Squares, if the either height or width of the Red Highlighted rectangle is an odd number) in exact accordance with the Mouse's y-axis movement, and outlying Red Highlighted Terrain Squares are raised or lowered according to the Edit Terrain Falloff value.

#### Select Scribble Terrain Tool

- Long Tooltip (Unchecked State): *Activate Scribble Based Terrain Editing. Holding down [secondary action] will cause the Terrain underneath the Mouse cursor to be, little by little, worn down. Holding down [Primary Action] will cause the Terrain underneath the Mouse cursor to be, little by little, raised upward.*
- Long Tooltip (Checked State): *Deactivate Scribble-Based Terrain Editing. Pressing [Primary Action] will select Fish and Objects in the Aquarium, and not modify the Terrain.*

*Since Grid-Based Tile Editing is currently, activated, holding down [Secondary Action] will cause the Terrain underneath the Mouse cursor to be, little by little, worn down. Holding down [Primary Action] will cause the Terrain underneath the Mouse cursor to be, little by little, raised upward...*

- A Check Mark denotes activation (on activation, this Menu Choice de-activates all other Aquarium Terrain Editing Tool

Menu Choices).

- 3D Mouse Cursor Movement is used in the following manner: while [Primary Action] is depressed, for every pixel (of the screen) the Mouse Cursor moves over, the Red Highlighted Terrain Square's (z) position is increased by a small amount. [Secondary Action] behaves identically, save that the Red Highlighted Terrain Square's (z) position is decreased by a small amount.

Change the Duration of the Day/Night Cycle

- Long Tooltip: *Change the length of time that a full Aquarium Day spans.*
- Spawn "Day/Night Cycle Length Dialog Box," which consists of
  - \* Text: *Input the length of time that a full Aquarium Day spans.*
  - \* A Text Field that defines the length, in minutes and seconds, each Day and Night lasts for.
  - \* PushButtons "OK" and "Cancel"

Visibility

- Long Tooltip(Unchecked): *Make [entity] visible.*
- Long Tooltip(Checked): *Make [entity] invisible.*
- The following Items are Toggle Menu Items; if Checked, that entity is rendered in the POV, where "entity" is one of the following:
  - Fish
  - Objects
  - Terrain

## 2.6 Edit Terrain Falloff Value

- Located at the bottom middle of the screen if the Select Area tool is activated (see Select Aquarium Terrain Editing Tool).
- Consists of
  - Text: *Edit Terrain Falloff Value*
  - Numerical Text Field that accepts floating point values in the range from 0 to 1.
  - Short Tooltip: 1 = plateau, 0 = very steep slope
  - Long Tooltip: Once you've selected a rectangular region of Terrain squares, and raised or lowered the region by holding down [Alter Z

Position], this number defines how steep the slope of the rectangular region will be. 1 means that the rectangular region will be raised or lowered all at once, like a plateau. The smaller the number, the steeper the slope from the middle of the rectangular region to its edges will be.

- Has the following functionality:
  - 0 indicates that while the [Alter Z Position] key is depressed, only the central Red Highlighted Terrain Square (or Terrain Squares, if the either height or width of the Red Highlighted rectangle is an odd number) is raised or lowered in strict accordance to the Mouse's y-axis movement; no other Red Highlighted Terrain Squares are moved.
  - 0 indicates that while the [Alter Z Position] key is depressed, only the central Red Highlighted Terrain Square (or Terrain Squares, if the either height or width of the Red Highlighted rectangle is an odd number) is raised or lowered in strict accordance to the Mouse's y-axis movement; no other Red Highlighted Terrain Squares are moved.
  - 1 means that while the Alter Z Position key is depressed, the entire Red Highlighted Terrain Square area is raised or lowered in strict accordance to the Mouse's y-axis movement.
  - Intermediate values define varying falloff rates as the Terrain Squares get farther away from the centermost Terrain Square(s) such that:
    - \* The (z) position for a given Red Highlighted Terrain Square = (Z Position of Central Red Highlighted Terrain Square(s))\*(Edit Terrain Falloff Value)\*(Number of Terrain Squares Away From Central Red Highlighted Terrain Square(s)).

## 2.7 Fish Sidebar

*Taking a particular liking to one of the small minnows the player had buffeted with his morphing terrain, the player carefully grabs hold of the silvery creature and deposits it into a conservatory of unmoving Fish. He then saves the plucked Fish so as to preserve it at the very moment of its plucking.*

*The player notices that the remaining minnows seem to be clumping together, missing their friend. Taking pity on the small Fish, the player resolves to find a proper replacement for the missing minnow, and brings into existence another Fish nearby. This Fish, however, is huge and violent, and immediately goes about ramming its large, predatory head against the minnows' hiding places, gobbling up any who stray far from the remaining cliff pockmarks.*

- Located across the Left of the Screen
- Consists of:
  - The RandomFish Icon

- The RandomFish Icon has the following functionality:
  - Short Tooltip: *Spawn RandomFish*
  - Long Tooltip: *Pressing [Primary Action] on this icon causes a Fish with randomly generated Attributes and AI to be placed in a random location in the Aquarium.*
  - Pressing [Primary Action] on the button causes a RandomFish to appear in a random position in the Aquarium.

## 2.8 Fish Sidebar

- Located across the Left of the Screen
- Consists of:
  - An array of squares, inside each of which will be one of the following:
    - \* The RandomFish Icon (always the first square)
    - \* An Empty Icon (always the last square)
      - Short Tooltip: *You can put a Fish here.*
      - Long Tooltip: *You can drag a Fish from the Aquarium and put it into this Fish SideBar by positioning the Mouse Cursor over a Fish in the Aquarium, holding down [Primary Action], moving the Mouse Cursor into this Empty Icon, and finally releasing [Primary Action].*
      - Fish in the Aquarium can be Dragged into the Empty Icon (whereupon a new Fish Icon, representing the Dragged Fish, is spawned directly above the Empty Icon on the SideBar). The Fish is initially Grabbed and Dragged via 2D Mouse Cursor Movement, and is Dropped by releasing [Primary Action] while the Mouse Cursor hovers over the Empty Icon.
    - \* A Fish Icon
      - Short Tooltip: *Drag and drop into Aquarium or open to Edit*
      - Long Tooltip: *You can drag this Fish into the Aquarium by holding down [Primary Action] while the Mouse Cursor hovers over this Fish Icon, and then moving the Mouse Cursor into the Aquarium. Alter the Fish's position by moving the Mouse Cursor normally, and hold down [Alter Z Position] if you want to move the Fish higher or lower. Alter the Fish's orientation by holding down [Alter Orientation] and moving the Mouse Cursor. When the Fish is positioned and oriented the way you want, release [Primary Action] to set it free in the Aquarium.*

- [Secondary Action] on this Fish Icon to open this Fish in the Fish Editor. Remember that the Aquarium keeps going even while you're Editing!
  - Fish Icons will have several small bar graphs that graphically represent the value of each Base Attribute of the Fish.
  - Pressing [Secondary Action] on a Fish Icon opens that Fish in the Fish Editor.
  - Pressing [Primary Action] on an Fish Icon Selects it, and un-Selects the previously Selected Fish Icon
  - The first Fish Icon is, by default, Selected.
- A Vertical scrollbar, which spans the right border of the SideBar. The scrollbar cannot be scrolled past the Empty Icon (at lowest), or the RandomFish (at highest)
- Four Pushbuttons, located in a grid at the top of the SideBar, allow the Player to Save a Fish To File, Load a Fish From File, Save All Fishes To File, and Load All Fishes to File
  - \* Each Pushbutton spawns a “Save/Load (All) Fish(es) In(to) Fish SideBar Dialog Box,” all of which are “Save/Load File Forced Pathname Dialog Boxes” (Higher Goal: The Dialog Boxes will be “Standard Windows Constructs”)
    - When the player Saves a single Fish Icon, the Selected Fish Icon is the Fish Icon that is saved
    - A (single) Loaded Fish is added to the Fish SideBar in the same manner as if it had been Dragged from the Aquarium and Dropped in the Empty Icon.
    - Load All Fish spawns a “Sanity Prompt Dialog Box”
    - If chosen, all current Fish are replaced by the Loaded Fish
- One Delete Fish PushButton, located just underneath the Save/Load PushButtons, that, when [Primary Action]ed
  - \* Spawns a Sanity Prompt Dialog Box
    - Deletes the Selected Fish
    - The Fish Icon directly above the Deleted Fish becomes Selected

Fish and RandomFish can be Dragged and Dropped into the Aquarium. The Grab half of the operation is accomplished by depressing [Primary Action] on the Fish Icon or RandomFish Icon. The Drag-and-Drop half of the operation is accomplished through positioning and orienting the Fish with 3D Mouse Cursor Movement and then releasing [Primary Action]. A two-step procedure is applied to the Dropped Fish to ensure no Fish or Objects reside inside of one another:

1. If the Fish is Dropped inside of the Terrain, the Fish's (z) position is redefined to be the lowest possible value while still placing the entire Fish model above the Terrain. The Fish then checks to see if the Fish is now inside of other Fish/Objects.

2. If the Fish has been moved inside of other Fish/Objects, then those Fish/Objects' positions are updated by a vector emanating radially from the center of the moved Fish such that the moved Fish is no longer inside of the other Fish/Objects. Step 1) is now performed on each Fish/Object that was radially moved.

## 2.9 Object Sidebar

*Aghast at the violence he wreaked on the unsuspecting minnows, the player quickly creates an invisible object that generates a strong, forceful current. This current sweeps the large, predatory Fish away from the cliff and past the player, leaving the surviving minnows to cower in their cliff pockmarks.*

- Located across the Right of the Screen
- Consists of:
  - An array of squares, inside each of which will be one of the following:
    - \* An Object Icon
      - Short Tooltip: *Drag and drop into Aquarium or open to Edit.*
      - Long Tooltip: *You can drag this [Object] into the Aquarium by holding down [Primary Action] while the Mouse Cursor hovers over this [Object] Icon, and then moving the Mouse Cursor into the Aquarium. Alter the [Object]'s position by moving the Mouse Cursor normally (hold down [Alter Z Position] to move the [Object] higher or lower). Alter the [Object]'s orientation by holding down [Alter Orientation] and moving the Mouse Cursor. When the [Object] is positioned and oriented the way you want, release [Primary Action] to release it in the Aquarium.*
  - [Secondary Action] on this [Object] Icon to Edit its Attributes*
  - Pressing [Primary Action] on an Object Icon Selects it, and un-Selects the previously Selected Object Icon
  - The first Object Icon is, by default, Selected
  - Object Icons can be Dragged and Dropped into the Aquarium. The Grab half of the operation is accomplished by depressing [Primary Action] on the Object Icon. The Drag-and-Drop half of the operation is accomplished through positioning and orienting the Object with 3D Mouse Cursor Movement and then releasing [Primary Action].
  - Pressing [Secondary Action] on an Object Icon calls up an "Edit Object Dialog Box" (with Pushbuttons, Sliders, and Text Fields the exact interface will be uniquely defined for

- each Object), which allows the Player to edit the properties of that Object.
- \* Create New Object Template Icon (always the second to last square)
    - Short Tooltip: *Create New Object Template*
    - Long Tooltip: *[Primary Action] on this Icon to create a new Object. You will be able to specify the Attributes this Object will have. When you are done choosing which Object to create and what its Attributes will be, the Object will appear at the bottom of the Object SideBar.*
    - The Create New Object Template Icon brings up that Object's "Edit Object Dialog Box," which consists of:
      - Pairs of Text and Text Fields, which enumerate the Attributes of the Object - they begin set to default values.
      - PushButtons "OK" and "Cancel"
      - If "OK" is selected, then the new Object spawns in a new Object Icon square, which becomes the second-to-last square of the SideBar.
  - \* An Empty Icon (always the last square)
    - Short Tooltip: *You can put an Object here.*
    - Long Tooltip: *You can drag an Object from the Aquarium and put it into this Object SideBar by positioning the Mouse Cursor over an Object in the Aquarium, holding down [Primary Action], moving the Mouse Cursor into this Empty Icon, and finally releasing [Primary Action].*
  - \* Objects in the Aquarium can be Dragged into the Empty Icon (whereupon a new Object Icon, representing the Dragged Object, is spawned directly above the Create New Object Template Icon on the SideBar). The Object is initially Grabbed and Dragged via 2D Mouse Cursor Movement, and is Dropped by releasing [Primary Action] while the Mouse Cursor hovers over the Empty Icon.
  - A Vertical scrollbar spanning the left border of the SideBar. The scrollbar cannot be scrolled below the Create New Object Template Icon square, or above the first Object Icon square.
  - Four Pushbuttons, located in a grid at the top of the SideBar, allowing the Player to Save an Object To File, Load an Object From File, Save All Objects To File, and Load All Objects to File.
    - \* Each Pushbutton spawns a "Save/Load (All) Object(s) In(to) SideBar Dialog Box," all of which are "Save/Load File Forced Pathname Dialog Boxes" (Higher Goal: The Dialog Boxes will be "Standard Windows Constructs")
      - When the player Saves a single Object Icon, the Selected Object Icon is the Object Icon that is saved.

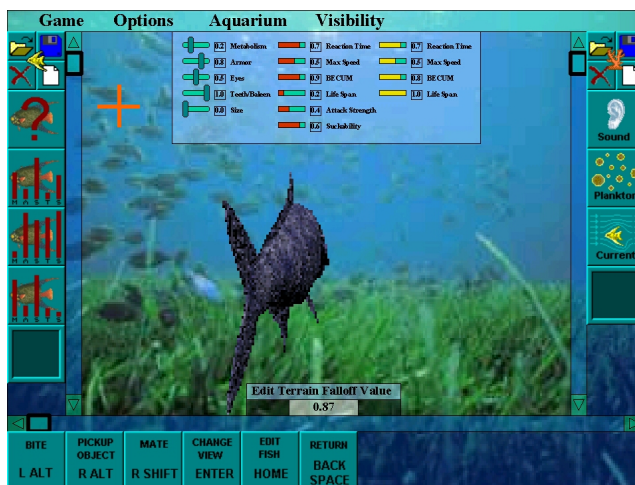
- A (single) Loaded Object spawns in a new square, which becomes the second-to-last square of the SideBar.
  - Load All Objects spawns a “Sanity Prompt Dialog Box.”
  - If chosen, all current Objects are replaced by the Loaded Objects
- One Delete Object PushButton, located just underneath the Save/Load PushButtons, that, when [Primary Action]ed
  - \* Spawns a Sanity Prompt Dialog Box
    - Deletes the Selected Object
    - The Object Icon directly above the Deleted Object becomes Selected

### 3 Objects

Objects are the interesting environmental features of the Aquarium. Whether it's Plankton Nodes generating vital sustenance for the less predatory Fish, Current Nodes creating slipstreams Fish must strive to navigate, or Sound Nodes manifesting an aural ambience, Objects are sure to keep the denizens of the Aquarium interested.

- Plankton Nodes generate plankton in the area emanating from each Plankton Node. They have the following property:
  - Rate of Plankton Generation
    - \* A floating point value from 0 to 1
    - \* Manipulated by the Player via a numerical Text Field
    - \* Long Tooltip: *The larger the number, the faster this Node generates Plankton.*
- Current Nodes push Fish and Objects about by generating current in a single direction, and have the following properties:
  - Magnitude of Force At Node
    - \* A floating point value from 0 to 1
    - \* Manipulated by the Player via a numerical Text Field
    - \* Long Tooltip: *The larger the number, the stronger the Node's current will be nearby this Node.*
  - Force Falloff
    - \* A floating point value from 0 to 1
    - \* Manipulated by the Player via a numerical Text Field
    - \* Long Tooltip: *The larger this number is, the farther this Node's current will reach.*
- Sound Nodes generates ambient noise, and have the following properties:
  - Amplitude
    - \* A floating point value from 0 to 1
    - \* Manipulated by the Player via a numerical Text Field
    - \* Long Tooltip: *The larger this number is, the louder the sound this Sound Node will generate.*
  - Sound Effect
    - \* The filename of the sound effect to be continuously emitted

- \* Graphically depicted as a button, with the text of the filename centered on it
- \* Manipulated by the Player via a "Standard Windows Construct" for loading files
- \* Long Tooltip: *This Node emits the Sound in this file.*



## 4 Fish Mode

A player enters Fish Mode when he takes Possession of a Fish. He closely watches the Fish and moves with that Fish's body. He is capable of commanding the Fish to do all the things it naturally does: Bite and kill other Fish (thereby creating nutritious Chum that can be eaten), feed on Plankton, play with Objects in the Aquarium, Mate with other Fish, explore its surroundings, and even view the inner workings of the Fish's biological body. What's more, the player still retains all of his old abilities from God Mode while playing in Fish Mode. But now, it's personal.

With the exception of the POV, Fish Mode is a superset of God Mode. That is, it adds user interface elements on top of the user interface elements already present in God Mode.

### 4.1 Point of View (POV)

The Aquarium, and the Fish, Terrain and Objects that populate it, are rendered in realtime 3D from one of the following POVs' position and orientation:

1. The Glass Fish View places the POV such that it looks directly along the Fish's Local Z-Axis (the axis that goes through the middle of the Fish model). The Fish model is transparent to a Player-configurable degree. Although a Mouse Cursor is overlaid on the POV and 2D Mouse Cursor Movement allows the player to [Primary Action] on Fish and Objects in the POV, Mouse movement also causes the POV to rotate (though a stationary Fish will remain stationary and won't inherit the POV's rotation), as in God Mode  $\rightarrow$  Movement Mode.
2. The Traditional View places the POV such that it is looking at the Fish from above and behind the Fish (i.e. the Fish would be seen top-down

if the Player looked straight down, and would actually move out of the field of view if the Player looked straight up, as the Fish would be underneath the POV). Camera AI will move the POV in special circumstances (such as when Terrain obscures the player’s entire view of the Aquarium) so as to consistently provide the Player with a clear view of the Fish’s surroundings. Mouse movement is handled as per the Glass Fish View.

3. The Traditional View With Moving Cursor is identical to The Traditional View, save that the Mouse Cursor, via 2D Mouse Cursor Movement, defines the endpoint of an invisible line that originates from the center of the Fish. The endpoint of this invisible line is the Red Projected Cursor, and is displayed as a red circle projected onto the first entity (Fish, Object or Terrain) that the invisible line intersects. In other words, the Fish has a “lasersight” mounted on its nose. When the Red Projected Cursor is projected onto a Fish or Object, that Fish or Object becomes Orange Highlighted. [Primary Action]ing causes the Fish or Object that is currently Orange Highlighted to become Red Highlighted.
4. The Traditional View With Tunnel view is identical to The Traditional View, save that a Tunnel (composed of Orange Circle Sprites) is traced out from the center of the Fish. The first Fish or Object that intersects the Tunnel is Orange Highlighted. [Primary Action]ing causes the Fish or Object that is currently Orange Highlighted to become Red Highlighted.

## 4.2 Fish Statistics

- Are visible in both Fish Mode → Movement Mode, Fish Mode → GUI Mode, and Fish Editing Mode.

## 4.3 Movement Mode

*Becoming curious about the minnow’s view of the world, the player promptly Possesses the closest one. Vision and locomotion alike recalibrate, yet remain intrinsically familiar as the player takes command of the little Fish and swims, experimentally, out of the cliff pockmark.*

- [Move Forward] will cause the Fish to turn towards, and then move in, the direction specified by the POV’s Z Axis. [Move Backward] will cause the Fish to turn towards, but move away from, the direction specified by the POV’s Z Axis.
- The player can perform context-sensitive Fish Actions by [Primary Action]ing in the POV. For example, if the Fish is swimming by a piece of Chum, [Primary Action] would cause the Fish to Eat. However, if the Fish’s nose is up against another Fish, [Primary Action] would cause the player-controlled Fish to Bite the nearby Fish.

- The Mouse Cursor (be it a crosshair, a Red Projected Cursor, or a Tunnel) will change color to indicate what Fish Action will be taken when the player [Primary Action].
- Each Fish Action will also be mapped to a key. The player can explicitly perform a Fish Action by striking the key the Action is mapped to.

## 4.4 GUI Mode

*Feeling hungry, the player darts after another small minnow, and causes his Possessed Fish to nip at its schoolmate's tail. Startled, the other minnow speeds away. Noticing a small bit of Chum floating near the Possessed Fish's mouth, the player gobbles it down, and feels a little closer to satiation. Eying some floating Plankton, the player swims into the biological mass and automatically slurps it down until the Possessed minnow can eat no more.*

### Fish Action SideBar

- Located across the Bottom of the Screen
- Consists of:
  - An array of squares
    - \* is depicted with a Fish Action Icon
    - \* has a key associated with it, and the text of said key ['A', 'Alt', 'Enter', etc.] is displayed on the square, in front of the Icon.
    - \* is greyed out if that Action cannot be performed.
    - \* [Primary Action]ing on a square causes the Fish to perform the corresponding Fish Action, just as if that square's associated key had been struck
    - \* [Secondary Action]ing on a square spawns the "Fish Action Key Map Dialog Box" which consists of:
      - Text: *Strike the key you would like to map [Fish Action] to.*
      - The following behavior: the next key pressed becomes the key mapped to the given square's Fish Action. The Dialog Box vanishes.
    - \* See Fish Actionsfor Tooltips
  - The third to last square will always be Change View
    - \* Short Tooltip: *Change view*
    - \* Long Tooltip: *Shift to a new camera view of the Fish you are Possessing*
    - \* Upon a [Primary Action], depending on the number of POVs implemented, either
      - shifts to the other POV or
      - Spawns a "POV Selection Dialog Box" that consists of a PushButton for each possible POV.

- The second to last square will always be Edit Fish
  - \* Short Tooltip: *Edit Fish*
  - \* Long Tooltip: *Open the Fish you are Possessing in the Fish Editor. Remember that the Aquarium keeps going even while you're Editing!*
  - \* Opens this Fish in the Fish Editor
- The last square will always be Return to God Mode
  - \* Short Tooltip: *Unpossess Fish*
  - \* Long Tooltip: *Unpossess the Fish and return to God Mode. You'll revert to a disembodied style of movement.*
  - \* Unpossesses the Fish, returning the player to God Mode. The God Mode POV will have the same position and orientation as the last position and orientation assumed by the Fish POV.

## 5 Fish Statistics

*Noticing that his Possessed Fish is built to swim quickly, but is also unlikely to live for much longer, the player decides to make the most of his Possession by zipping away from the coral cliff face as quickly as he can. Getting caught in the current the player created beforehand, the player zooms away from the cliff face, toward a pair of truly enormous Fish, both far larger than the Possessed minnow.*

A Fish's Statistics are numerical descriptions of its Attributes its biological state. The Fish's healthiness, level of energy, hunger, and many other traits are shown, in detail, for the interested player.

- Located in the Middle Top of the screen
- All Fish Attributes are textually represented as Statistics Base Attributes in green, Derived Attributes in red, and Reactive Attributes in yellow.
- See Fish Attributes for the Long Tooltip that are associated with each Fish Attribute.
- Sliders
  - Graphically depict the value of each Fish Attribute next to its textual representation.
  - The Base Attributes' values can be changed by manipulating the Sliders or by entering in normalized numbers in the Base Attributes' Text Field. The Derived Attributes' values, the Reactive Attributes' maximum (and possibly current) values, and the 3D Fish Model immediately reflect these changes.

### 5.1 Fish Actions

Note that any two Fish Actions that both make use of the same part of the Fish's body are mutually exclusive one Action cannot be taken until the other is completed. For example, if a Fish is in the midst of Eating Chum, then it cannot Bite another Fish until its Mouth has closed, completing the Eating Action.

#### 5.1.1 Mouth

- Eat
  - Short Tooltip: *[Primary Action] to Eat. [Secondary Action] to remap key.*
  - Long Tooltip: *[Primary Action] or strike [mapped key] to Eat any Chum that is near the Fish's mouth. Note, Plankton consumption is automatically done unless the fish is already using its mouth or its*

*stomach is full.*

*[Secondary Action] to change the key that causes the Fish to Eat.*

- Requires Chum to be near the Fish’s mouth
- The Fish consumes nearby Chum.

- Bite

- Short Tooltip: *[Primary Action] to Bite. [Secondary Action] to remap key.*
- Long Tooltip: *[Primary Action] or strike [mapped key] to Bite any other Fish that is nearby the Fish’s mouth.*

*[Secondary Action] to change the key that causes the Fish to Bite.*

- Requires another Fish (alive or dead) to be near the Fish’s mouth.
- The Fish Bites a nearby Fish, causing some amount of damage to the Bitten Fish. The Bitten Fish also generates some amount of Chum.
  - \* For live Fish, the damage is deducted from that Fish’s current Health value and a proportional amount of Chum is generated.
  - \* For dead Fish, the damage is deducted from that Fish’s current Size value and a proportional amount of Chum is generated.

- Pick Up Object

- Short Tooltip: *[Primary Action] to Pick Up. [Secondary Action] to remap key.*
- Long Tooltip: *[Primary Action] or strike [mapped key] to Pick Up any Object that is near the Fish’s mouth. While the Fish is carrying an Object, it will not be able to pick up another Object.*

*[Secondary Action] to change the key that causes the Fish to Pick Up Objects.*

- Requires that the Fish is not carrying an Object and that a carry-able Object is near the Fish’s mouth.
- Fish picks up Object in its mouth and thenceforth carries the Object about.
- The Fish’s Size becomes the Fish’s Size plus the Size of the Object it is carrying, so long as that Fish carries that Object.
- If a Fish dies while carrying an Object, that Object is released as if it had been Dropped.

- Drop Object

- Short Tooltip: *[Primary Action] to Pick Up. [Secondary Action] to remap key.*
- Long Tooltip: *[Primary Action] or strike [mapped key] to Drop any Object that the Fish's is carrying.*
- [Secondary Action] to change the key that causes the Fish to Drop Objects.*
- Requires that the Fish is carrying an Object
- Fish lets go of the Object

### 5.1.2 Mating

- Mate
  - Short Tooltip: *[Primary Action] to Mate. [Secondary Action] to remap key.*
  - Long Tooltip: *[Primary Action] or strike [mapped key] to Mate this Fish with another Fish. In order to Mate, a Fish must have a high enough Mating Inclination and there must be a compatible Fish available to Mate with (but a Fish is can always Mate with itself!). The offspring Fish will have a combination of the Attributes of both parents, as well as some random mutations. The offspring Fish will appear next to the parent that has more Current Health.*
  - [Secondary Action] to change the key that causes the Fish to Mate.*
  - Requires Mating Inclination requirements to be met
  - The Fish either
    1. Selects another Mating-capable Fish, and produces a Newborn Fish
    2. Buds asexually, producing a Newborn Fish
  - When a Player performs this action, the "Mating Dialog Box" is spawned. It consists of:
    - \* Text: *Select a Fish to Mate with.*
    - \* A Scrollable column of Fish Icons which
      - Lists every Fish Icon that is available to Mate
      - Short Tooltip: *[Primary Action] to Select, [Secondary Action] to View*
      - Long Tooltip: *[Secondary Action]ing on a Fish Icon to check out its Attributes might help you decide whether or not this Fish is a good candidate to Mate with. [Primary Action]ing on a Fish Icon Highlights that Fish Icon*

- [Secondary Action]ing on a Fish Icon spawns the “Fish Statistics Dialog Box,” which consists of that Fish’s Statistics and PushButton “OK.”
- The first Fish Icon is always “Bud Asexually”. This line cannot be [Secondary Action]ed upon.
- \* PushButtons “OK” and “Cancel”
  - If “OK” is selected while a Fish Icon is Highlighted, then the Fish will Mate with the Highlighted Fish, or will Bud Asexually, depending on the Player’s choice.

### 5.1.3 Body

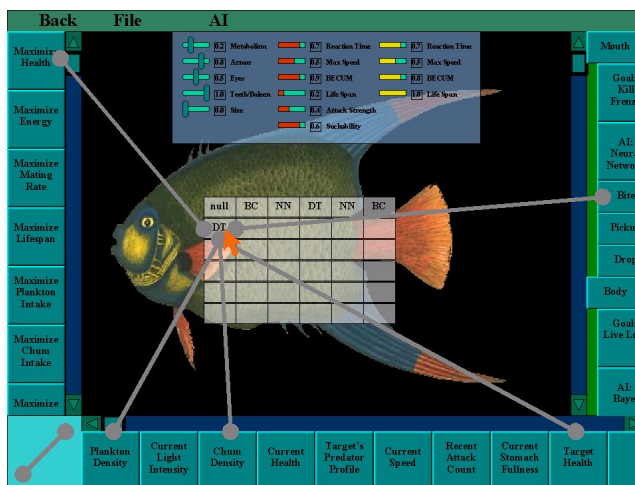
- Swim
  - While moving, the Fish consumes nearby Plankton proportional to both the Plankton density and the Fish’s speed.
  - Note that these Fish Actions do not appear on the Fish Action Side-Bar, and are not accessible by the player.
  - Swimming Actions that require a subject denote valid subjects in parentheses.
  - Several variations:
    1. Chasing (a Fish or Object)
    2. Following (a Fish or Object)
    3. Fleeing (from a Fish or Object)
    4. **Hiding (from a Fish or Object)**
    5. Searching for Food
    6. **Schooling**
    7. **Exploring**

## 6 Mating and Evolution of Species

*Feeling Inclined to Mate, the player takes a look at other Fish that his Possessed Minnow could Mate with. Noticing that one of the large Fish passing above the Possessed minnow is available to Mate, the player quickly performs the brief Mating procedure, and a new, misshapen Fish comes into existence next to the minnow. The minnow's offspring promptly eats its parent, and the player is hurtled back into God Mode.*

The Fish in the Aquarium evolve over time as natural selection weeds out the unfit Fish and rewards those Fish best suited for survival, who pass on their traits to a new generation of Fish not only by Mating, but also by Budding Asexually.

- When a Fish's Mating Inclination reaches or exceeds a value of 0.8, it makes itself available to Mate.
- Periodically, each Fish that is available to Mate will be evaluated based on all defined Fitness Functions and/or Goals. If two Fish are found to be optimizing a (Player defined) minimum number of Fitness Functions and/or Goals, those two Fish will be selected by the simulation to Mate.
- When two Fish Mate, crossover and mutation techniques will be applied to their respective AI Architectures to generate the AI Architecture for their offspring. Likewise, the offspring will have Attributes that are a combination (plus possible mutation) of the Attributes of their parents.
- If no compatible partner becomes available within a set time period, the Fish will Bud Asexually; i.e. will produce offspring based solely on its own characteristics. Mutation techniques will be applied (to a more extreme degree than in regular Mating) to introduce variation in offspring produced in this manner.
- Offspring will be given small Attribute adjustments based on the Fitness Functions that their parent(s) were optimizing at the time of the offspring's birth. This will cause Fish that, say, are optimizing The Vicious Killing Of Other Fishes to gradually, over generations, gain sharper Teeth.
- Offspring's starting positions are near the parent with the greater Current Health value. If both parents have the same Current Health value, then the offspring appears next to one parent at random. A two-step procedure is applied to the offspring to ensure no Fish or Objects reside inside of one another.



## 7 Fish Editing Mode

*Wondering about the patricidal misshapen Fish's motives, the player focuses his attention on the inner workings of newborn Fish's brain. Viewing the Fish brain's components and their interconnections, it is clear to the player that this Fish is fixated on gorging itself on as many Fish as it can find. Feeling somewhat vengeful (after all, this misshapen Fish had the gall to eat one of its own parents), the player reconnects the Fish's brain so that it becomes irrationally afraid of itself.*

When a player opens a Fish in Fish Editing Mode, he is witnessing the private thought processes of the Fish firsthand. The Goals the Fish strives for, the Stimuli it examines, and the AI Units that collectively form the brain of the Fish are laid bare before the player, who can even take matters into his own hands and rewire the Fish's brain personally!

- While in Fish Editing Mode, the Fish being Edited vanishes from the gameworld. The rest of the game continues as usual while the Fish is being edited. In other words, the simulation continues "behind the scenes" during Fish Editing, and when the player stops Fish Editing and returns to the gameworld (in God Mode or Fish Mode) the simulation will likely have visibly changed since the player began Fish Editing.
- The following user interface completely replaces, and is independent of, the user interfaces defined in God Mode and Fish Mode.

### 7.1 Point of View (POV)

- There is no POV into the gameworld in Fish Editing mode. Instead, the following graphical elements are displayed:

- A Mouse Cursor (an arrow) is overlaid over all other graphical elements, and tracks Mouse movement as in Windows.

## 7.2 Pull-Down MenuBar

- Located across the top of screen

Back

- Long Tooltip: *Exit the Fish Editor and return to [mode to return to]. returns player from whence he came (either God Mode or Fish Mode)*
- Returns player from whence he came (either God Mode or Fish Mode)

File

Quit Game

Save Fish To File

- Long Tooltip: *Saves the Fish you are currently Editing to a file.*
- Spawn “Save Fish Dialog Box,” which is a “Save File Forced Pathname Dialog Box”
- “Save Fish Dialog Box” is a “Standard Windows Construct”

Load Fish From File

- Long Tooltip: *Places the Fish you are currently Editing back into the Aquarium at its last known position (or in the Fish SideBar if it was not swimming in the Aquarium), and loads up a Fish from file for Editing. The loaded Fish will appear in the Fish SideBar when you go Back to the Aquarium.*
- Spawn “Load Fish Dialog Box,” which is a “Load File Forced Pathname Dialog Box.”
- “Load Fish Dialog Box” is a “Standard Windows Construct”
  - The Fish that was previously being Edited is returned to the Aquarium at its last known position, or to the SideBar if it was not in the Aquarium.
  - A Loaded Fish:
    - \* is added to the Fish SideBar as if it had been Dragged from the Aquarium and Dropped into the Fish SideBar’s Empty Icon
    - \* becomes the current subject of Editing

AI

Clear AI Architecture

- Long Tooltip: *This will cause the Fish’s AI Architecture to be “clean-slatted” - completely erased! The Fish will be braindead unless you choose to build it a new AI Architecture.*

- Spawn “Sanity Prompt Dialog Box”
  - If chosen, this Fish’s AI Architecture becomes blank

Randomize AI Architecture

- Long Tooltip: *This will cause the Fish’s AI Architecture to be scrambled in a completely random manner! Its original AI Architecture will be lost.*
- Spawn “Sanity Prompt Dialog Box”
  - If chosen, this Fish’s AI Architecture and Attributes are scrambled randomly, – i.e. the Fish becomes a RandomFish

### 7.3 Fish Statistics

- Are visible in both Fish Mode → Movement Mode, Fish Mode → GUI Mode, and Fish Editing Mode

### 7.4 AI Architecture Text

- Located in the center of the Screen
- Describes the AI Architecture the “brain” of the Fish in sufficient detail to enable effective debugging of the AI Module. No manipulation of the text is allowed; only viewing.

### 7.5 AI Unit Grid

- Located in a Grid in the center of the screen (replaces AI Architecture Text)
- Consists of:
  - A two dimensional array of squares, inside of which can be one of the following:
    - \* The Null AI Unit (represented as an Empty Icon)
      - Short Tooltip: [Primary Action] to Add AI Unit.
      - Long Tooltip: This is the Null AI Unit - it doesn’t do anything at all. [Primary Action] to change this Unit into an AI Unit that will help the Fish think.
    - \* A functioning AI Unit (represented as an icon that corresponds to the AI Technique the AI Unit is using)
      - Short Tooltip: [Secondary Action] or Connect to alter.
      - Long Tooltip: *This AI Unit is Connected to one Action, one Fitness Function, and any number of Stimuli. The AI Unit will think about the Connected Stimuli (what’s happening to*

*the Fish) and, if taking (or refraining from) its Connected Action is considered by the Connected Fitness Function to be "good", then the AI Unit will "argue" with the other AI Units, trying to get the Fish to take (or refrain from) its Connected Action. You can add, remove, or alter these Connections using the Connection Tool.*

*[Secondary Action] to choose a different AI Technique to do the thinking for this AI Unit.*

- [Primary Action]ing on an AI Unit
  - Red Highlights the AI Unit
  - Un-Highlights the previously Red-Highlighted Square
  - Draws the Connection from this AI Unit to a Fitness Function, the Connection from this AI Unit to a Fish Action, and any and all Connections to Data Filters. No other Connections are drawn.
- [Secondary Action]ing on an AI Unit spawns the "AI Unit Dialog Box," which consists of:
  - Four PushButtons, three of which allow the Player to select an AI Technique, where the fourth allows the Player to define the Empty Icon (the Null AI Unit)
- Note that when the player chooses to endow a Null AI Unit with an AI Technique, the newly-functioning AI Unit is automatically Connected to the first Fish Action on the Action Decision Node SideBar, the first Fitness Function on the Fitness Function SideBar, and the first Data Filter on the Data Filter SideBar.

## 7.6 Action Decision Node SideBar

- Located Across the Right of the Screen
- Consists of:
  - An array of squares, inside of each of which is an Action Decision Node, which consists of:
    - \* An AI Unit square with the following exceptions:
      - Has the following Tooltips:
        - Short Tooltip: *[Secondary Action] to alter.*
        - Long Tooltip: *This Action Decision Node AI Unit will think about each Action that Connected AI Units are suggesting, and then try to achieve its Goal by either doing*

*nothing, or taking a suggested Action. [Secondary Action] to choose a different AI Technique to do the thinking for this Action Decision Node AI Unit.*

- There are never Connections to this AI Unit.
- This AI Unit's "AI Unit Dialog Box" does not allow the player to choose the Null AI Unit (it simply omits that Push-Button)
- \* A Goal Square, which consists of a Goal Icon and has the following functionality:
  - Short Tooltip: *[Secondary Action] to alter.*
  - Long Tooltip: *The Goal is what the Action Decision Node constantly tries to achieve. All of the Actions under this Action Decision Node will only be taken if the Action Decision Node thinks that taking an Action will get the Fish closer to this Goal. [Secondary Action] to choose a different Goal for this Action Decision Node to try to achieve.*

*[Secondary Action]ing spawns a "Goal Dialog Box", which consists of:*

- Text: *Select which Goal the Action Decision Node*
- A Scrollable Text Field, where each Text Line
  - is a Goal (for example "Kill Everything," or "Stay Away From All Other Fish")
  - has a Long Tooltip that gives a more verbose description of the Goal it defines (for example, "Fish will be very aggressive, and try to maximize its kill count by ending the lives of other Fish whenever possible.")
- PushButtons "OK" and "Cancel"
- \* A Fish Action square...
  - ...one for each Fish Action. Each Fish Action is represented as a unique icon that resides in the square.
  - Short Tooltip: *Connect to alter [Fish Action].*
  - Long Tooltip: *Any AI Units that are Connected to this Action are able to "argue" with the other Connected AI Units, trying to persuade the Action Decision Node to either [Fish Action] or to refrain from attempting to [Fish Action]. Any number of AI Units can be Connected to a Fish Action. You can add, remove, or alter these Connections using the Connection Tool.*
  - [Primary Action]ing on a Fish Action
    1. Red Highlights the AI Unit
    2. Un-Highlights the previously Red-Highlighted Square
    3. Draws the Connection from this Fish Action to any and all other AI Units. No other Connections are drawn.

## 7.7 Fitness Function SideBar

- Located Across the Left of the Screen
  - Consists of:
    - An array of squares, inside each of which will be:
      - \* A Fitness Function Icon
        - Short Tooltip: *Connect to alter.*
        - Long Tooltip: *[Fitness Function information describing what type of situations it considers "good" (for example: Maximizes Health: "Fish will always be concerned with having a full Current Health statistic.")]*
- Fitness Functions tell any Connected AI Units what kinds of Stimuli it considers "good." A Fitness Function can be Connected to any number of AI Units. AI Units use Fitness Functions to decide what kinds of situations are desirable and what kinds of situations are to be avoided. You can add, remove, or alter these Connections using the Connection Tool.
- [Primary Action]ing on a Fitness Function
    1. Red Highlights the Fitness Function
    2. Un-Highlights the previously Red-Highlighted Square
    3. Causes all Connections from this Fitness Function to AI Units to be drawn. No other Connections are drawn.
  - A Vertical scrollbar, which spans the right border of the SideBar. The scrollbar is not allowed to be scrolled past the bottommost Fitness Function (at the lowest), or the topmost Fitness Function (at the highest).

## 7.8 Data Filter SideBar

- Located Across the Bottom of the Screen
- Consists of:
  - An array of squares, inside each of which is a unique icon, which represents the stimulus the Data Filter expresses to an AI Unit. They have the following functionality:
    - \* Short Tooltip: *Connect to AI Units*

- \* Long Tooltip: *[Data Filter information describing what type of Stimulus it encapsulates (for example: “Nearby Plankton describes how much, and how close Plankton is to the Fish.”)]*

*Stimuli are used to let the Fish perceive its surroundings, so it can make decisions based on its observations. A Stimulus can be Connected to any number of AI Units. AI Units think about Connected Stimuli while trying to satisfy their Connected Fitness Function. If an AI Unit isn’t connected to a Stimulus, then it won’t even consider that Stimulus when it is deciding whether or not to “argue” for taking its Connected Action. You can add, remove, or alter these Connections using the Connection Tool.*

- \* [Primary Action]ing on a Data Filter
  1. Red Highlights the Data Filter
  2. Un-Highlights the previously Red-Highlighted Square
  3. Causes all Connections from this Data Filter to AI Units to be drawn. No other Connections are drawn.
- A Horizontal scrollbar, which spans the top border of the SideBar. The scrollbar cannot to be scrolled past the leftmost Data Filter or the right-most Data Filter.

## 7.9 Connection Tool

- A Square Icon located at the Bottom Right of the Screen
  - Short Tooltip: *Drag Connections*
  - Long Tooltip: *Hold [Primary Action] down and move the Mouse Cursor to draw Connections between AI Squares. If the Connection is invalid it will not be drawn. To erase Connections, hold [Secondary Action] down and draw a red line between the two Connected Squares that you wish to Disconnect. Click this Connection Tool again when you want to stop drawing and erasing Connections.*
  - [Primary Action]ing on the Connection Tool
    - \* When it is not Highlighted
      1. Red Highlights, and activates, the Connection Tool
      2. Un-Highlights the previously Red-Highlighted Square
    - \* When it is Highlighted
      1. Un-Highlights (and deactivates) the Connection Tool
- While the Connection Tool is active,
  - The player can draw Connections (depicted as lines) in the following way:

- \* Holding down [Primary Action] interactively defines a line, where the starting point is the pixel where [Primary Action] was initially depressed, and the ending point is the current position of the Mouse.
  - Note that any functionality triggered by [Primary Action]ing on other Squares (except for Connection-drawing functionality) is suppressed.
- \* Releasing [Primary Action] makes the Connection, if it is valid according to the following rules:
  1. Each AI Unit can be Connected to one Fish Action
  2. Each AI Unit can be Connected to one Fitness Function
  3. Each AI Unit can be Connected to any number of Data Filters
  4. Each Fitness Function can be Connected to any number of AI Units
  5. Each Fish Action can be Connected to any number of AI Units
  6. Each Data Filter can be Connected to any number of AI Units
  7. No “doubling” or “tripling” of Connections is possible - a Connection between two squares either exists or it does not.
- \* If a proposed Connection is invalid (say, because the player tried to Connect an AI Unit to a second Fish Action), then the blocking Connection (in this case the Connection from the AI Unit to the first Fish Action) blinks red for 2 seconds.
- The player can erase Connections in the same way they were drawn, except that [secondary action] is used instead of the [Primary Action], and the interactively defined line is colored red. Connections vanish when they are erased in this manner.

## 7.10 3D Rotating Fish

- Located in the background
- The model of the Fish is updated in realtime - it appears just as it would look in-game with its current set of Attributes

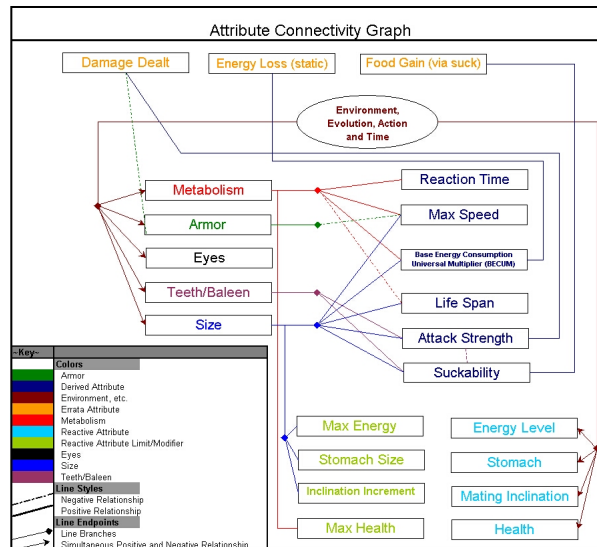
## 8 Fish Attributes

Having bestowed a serious psychosis to the patricidal, misshapen Fish, the player examines its Attributes. He notices that, in general, the Attributes seem to be fairly similar to the demised minnow's Attributes with some notable differences, especially the Size difference. Still harboring a grudge against the now psychologically impaired, misshapen Fish, the player reduces its Metabolism to the lowest possible setting, and returns to God Mode.

Note, formula and table generation routines in this section will use the letter K to specify a generic answer.  $K_i$  would imply the current solution and  $K_{i-1}$  would imply the previous answer in a series. Also, N and P are meant to represent generic constants.

A Fish's Attributes describes its physical makeup. A complex system of interweaved factors, Fish Attributes allow for a wide variety of Fish to be created from small predators whose short lifespan fuels furious activity, to gargantuan Plankton-eaters who ponderously go about their existence, and everything in between.

Below is a graph that visually depicts all of the Attributes and how they connect with each other. It would be useful for readers to look back at this graphic as they read through this section.



### 8.1 Base Attributes

Base Attributes are the primary Attributes for a Fish.

### 8.1.1 Metabolism (Meta)

- Long Tooltip: *High Metabolism makes a Fish more nimble and able to Mate more often, but uses up Energy faster, lives for a shorter time, and digests food more quickly.*
- Positively affects a Fish's Reaction Time and Mating Inclination.
- Negatively affects a Fish's Base Energy Consumption Universal Multiplier (BECUM) and Life Span.
- Increases the rate at which Food in the Fish's Stomach is converted into Energy.
- Metabolism is balanced by itself and Size.
- Metabolism values range from 0.0 to 1.0. A suggested "mutt" value for Metabolism is 0.2.

### 8.1.2 Size

- Long Tooltip: *A large Size makes a Fish live longer, have a bigger Stomach, more total Energy and more total Health, a higher top swimming speed, stronger Attacks, but will use up Energy more quickly.*
- Positively affects a Fish's Life Span, Max Stomach, Attack Strength, Suckability, Max Speed, Max Energy, and Max Health.
- Negatively affects a Fish's BECUM.
- Size is balanced by Metabolism and itself.
- Current formula to generate size from pounds is  $K_i = K_{i-1} * 1.07$
- Valid values for Size are 0.0 to 1.0 and a good "mutt" value is 0.2

### 8.1.3 Eyes

- Long Tooltip: *A high number for Eyes means that the Fish can see well in daylight and a low number means that the Fish can see well in the darkness of night. A number near 0.5 means the Fish is able to see fairly well all of the time.*
- Has no effect on a Fish's other Attributes.
- Are purely self-balanced as a result of being a ratio between Day and Night. A Fish with a 1.0 has perfect Day vision but is completely blind at Night (and vice versa). If a Fish has a Eyes value of 0.5, then it has equally poor vision at all times.

- Eyes are defined to be “Night-oriented” when less than 0.5 and “Day-oriented” when greater than 0.5.
- Valid values for Eyes are 0.0 to 1.0 with a “mutt” value of 0.5.

#### 8.1.4 Teeth/Baleen (TB)

- Long Tooltip: *A high number for Teeth/Baleen means that the Fish can Attack other Fish very well, but is poor at digesting Plankton. A low number means that the Fish is good at digesting Plankton, but is not very strong when Attacking.*
- Has no effect on a Fish’s other Attributes.
- Are purely self-balancing as a result of being a ratio from perfect Suckability (with no Attack Strength) to perfect Attack Strength (with no Suckability).
- TB are thought of as “Baleen” when less than 0.5 and “Teeth” when greater than 0.5.
- Valid values for Teeth/Baleen are 0.0 to 1.0 with a “mutt” value of 0.5.

#### 8.1.5 Armor

- Long Tooltip: *A high Armor means that the Fish is tougher and more resistant to being Attacked, but also has a slower top swimming speed, and isn’t as strong of an Attacker itself.*
- Reduces the damage taken from an attack by another Fish.
- Negatively affects the Max Speed and Damage Dealt of a Fish.
- Is balanced by itself.
- Valid values for Armor are 0.0 to 1.0 with a “mutt” value of 0.1.

### 8.2 Derived Attributes

Derived Attributes are generated from the Base Attributes and cannot be directly altered.

#### 8.2.1 Base Energy Consumption Universal Multiplier (BECUM)

- The BECUM is the rate of Energy that a Fish constantly loses simply by being alive (i.e. the Fish’s “standing energy consumption rate”). Taking Fish Actions add to the Energy loss of a Fish.
- $BECUM = (Size + Meta) / 2$ .

### 8.2.2 Attack Strength

- Attack Strength is the highest possible damage a Fish can deal.
- $\text{Attack Strength} = (\text{TB} * \text{Size}) * \text{P}$

### 8.2.3 Reaction Time

- Reaction Time determines how often a Fish gets the opportunity to update its AI.
- The formula for Reaction time is:  $K_i = K_{(i-1)} * 1.20$ , where the input is  $(\text{Meta} * \text{P} - \text{Size} * \text{N}) + K$

### 8.2.4 Max Speed

- Max Speed is the maximum velocity a Fish can achieve.
- $\text{Max Speed} = ((\text{Size} * \text{P} - \text{Armor} * \text{N}) * \text{MaxGameSpeed}) + K$

### 8.2.5 Suckability

- Suckability is a measure of how well a Fish can consume the Plankton that is floating in the Aquarium.
- $\text{Suckability} = (1 - \text{Teeth}/\text{Baleen}) * \text{N}$   
(Note: Size makes it influence known during the actual act of Sucking)

### 8.2.6 Life Span

- Life Span is a rating of how long a Fish could live for, assuming it doesn't get eaten or die of hunger.
- The formula for Life Span  $K_{i-1} * 1.03^2$ , where the input is  $(\text{Size} * \text{P} - \text{Meta} * \text{N}) + K$

## 8.3 Reactive Attributes

Reactive Attributes' maximum possible values are generated directly from Base Attributes. The current value of a Fish's Reactive Attributes fluctuates in real-time as various events occur to a Fish during gameplay. Neither the maximum, nor the current value of Reactive Attributes can be directly altered.

### 8.3.1 Stomach

- Long Tooltip: *The Stomach stores food while the Fish digests it into Energy. If the Stomach is empty, the Fish's Energy level will definitely decrease! If the Stomach is full, then the Fish can not eat until some Food is digested.*
- The Stomach is an entity that stores Food for a Fish before that Food is converted into Energy.
- The current value of Stomach will increase whenever the Fish eats. When Stomach's current value is equal to its maximum value, the Fish may no longer eat.
- The current value of Stomach will decrease constantly as Food is converted into Energy. The rate of conversion is determined by the Metabolism of the Fish.
- Newborn Fish will have a current Stomach value of 0.
- Maximum Stomach Capacity = Poundage / 2

### 8.3.2 Health

- Long Tooltip: *If a Fish's Health decreases to 0, the Fish dies. If a Fish has no Energy, then the Fish's Health will decrease. A wounded Fish will use some Energy to slowly increase its Health.*
- Health is a measure of a Fish's well-being. If it reaches 0, the Fish dies, and is only capable of floating in the water and being a source of Chum for other Fish.
- The current value of Health will gradually increase if the Fish's current Energy Level is above 5% of its Maximum Energy Level.
- The current value of Health is decreased if the Fish is attacked
- Newborn Fish's current value of Health will be equal to that Fish's Maximum Health.
- Maximum Health = Poundage / 2.

### 8.3.3 Energy Level

- Long Tooltip: *A Fish uses Energy to take any Action. A Fish also uses up Energy just by staying alive. Energy also causes Fish to slowly increase its Health if it's wounded. Energy increases as food in the Stomach is digested.*

- Energy is a measure of a Fish's vitality. If the current value of Energy reaches 0, further Energy expenditures will be deducted from that Fish's current Health value. If the current value of Energy is above 80% of Maximum Energy, then the Fish will gain Mating Inclination faster than usual.
- Energy will increase as Food in the Stomach is metabolized (i.e. as the current value of Stomach decreases, the current value of Energy will increase).
- The current value of Energy will constantly decrease, according to the Fish's BECUM, and will be further reduced by taking any Fish Action.
- Newborn Fish will have a current value of Energy equal to that Fish's Maximum Energy.
- $\text{Maximum Energy} = \text{Poundage} / 2$ .

#### 8.3.4 Mating Inclination

- Long Tooltip: *A Fish cannot Mate until its Mating Inclination has reached a suitably high level. The Mating Inclination of a Fish will increase faster if its Energy level is high.*
- Mating Inclination is a measure of how strongly a Fish is inclined to Mate.
- The current value of Mating Inclination will constantly increase. The rate of this increase is affected by the Metabolism of a Fish and its current Energy Level.
- Mating Inclination will be measured as floating point value from 0 to 1.
- The Fish will be available to Mate once Mating Inclination reaches the required level.
- Newborn Fish will begin with a current Mating Inclination of 0.

#### 8.4 Other Fish Formulae

- $\text{Damage Dealt to Fish} = (\text{AttackStrength} * P - \text{Armor} * N) + K$  (Where K ensures that damage is never zero).
- $\text{Plankton Eaten Via Sucking} = \text{Poundage} * (\text{PlanktonDensity}) * \text{Suckability} * P$
- Below, the Energy Consumption Table shows the rates and formulas that define the Energy cost of a given Fish living and taking Fish Actions.

<b>Event</b>	<b>Loss Formula</b>	<b>Rate</b>
Static Loss	BECUM	Every AI Update
Bite	Size * $(5 \cdot 10^{-4})$	Each Instance
Eating	1	Each Instance
Sucking	1	Each Instance
Defending	1	Each Instance
Swimming	CurrSpeed* $(5 \cdot 10^{-4})$	Once per Second

## 9 Offline Mode

*Having returned to the Aquarium, and witnessing the sluggish, unhinged behaviour of the misshapen Fish, the player decides that he'd like to see how the genetic landscape of the Fish changes over a large period of time. Entering Offline Mode, he abdicates his presence in the Aquarium, and instead views only abstract statistics describing the trends in the Aquarium.*

*He notices that the Aquarium is evolving at a feverish pace. Fish are being born, living and dying rapidly. Feeling as though he might be missing something important, the player returns to the Aquarium in God Mode.*

When a player decides to go Offline, he has chosen to let the Aquarium evolve on its own, without his intervention, at a greatly accelerated pace. Though he will watch the Aquarium only through a simple veil of statistics, generations of Fish will rise and fall, drastic mutations will be introduced into the gene pool, and new species of Fish, with unique behavior, will arise. When the player chooses to return to the Aquarium, it is likely to be a very different place indeed.

- Displays some textually represented statistics, including
  - Number of Fish in the Aquarium
  - Amount of Plankton in the Aquarium
  - Real time passed since the genesis of this Aquarium
  - Virtual time (in Fish Years) passed since the genesis of this Aquarium
- Gameplay is accelerated; that is, all CPU cycles that had been devoted to graphics and sound are instead devoted to the running of the Fish's AI and Mating processes.
- Striking any button returns the player from whence he came (either God Mode or Fish Mode).

## 10 Game Setup

*Having returned to the Aquarium, and beheld the drastically evolved species of Fish that have come about during his time in Offline Mode, the player resolves to start a new Aquarium from scratch (perhaps importing his favorite minnow he had saved). The player delves into the deeper aspects of the Aquarium, setting it up just the way he wants it.*

Designed to be the ultimate in flexibility, Game Setup allows the player to configure many aspects of gameplay from in-game controls, to the style of viewing the world, to even limiting the abilities of those players who are connecting to your networked Aquarium!

- The Game Setup program is run outside of the game itself it consists of a window with a Pull-Down MenuBar.
- All of the following functionality will be integrated into the game engine as described in the previous sections that reference "Game Setup".

### 10.1 Pull-Down MenuBar

#### Game Options

##### Key Mapping

*Long Tooltip: Make any key perform any game action.*

- Spawns "Key Mapping Dialog Box", which consists of:
  - Text: *[Primary Action] on each Game Action, and then push the key you'd like to execute that Game Action.*
  - An array of PushButtons, one for each possible User Interface Action, which allow the Player to map all User Interface Actions to any key by [Primary Action]ing the PushButton for a given User Interface Action, and then striking the key the Player wants to map to that User Interface Action
  - PushButtons "OK" and "Cancel"

##### Method of Switching Between Movement Mode and GUI Mode

*Long Tooltip[Toggle Switch]: You must strike [GUI Visibility] to switch between Movement Mode (where Mouse movements make you look in different directions) and GUI Mode (where Mouse movements let you choose options on the screen). [Primary Action] to change method.*

*Long Tooltip[QuickCommand Switch]: You must hold down [GUI Visibility] to switch between Movement Mode (where Mouse movements make you look in different directions) and GUI Mode (where Mouse movements let you choose options on the screen). Releasing [GUI Visibility] returns you to Movement Mode. [Primary Action] to change method.*

- Is a Toggle Menu Item
- State 1 is “Toggle Switch”; State 2 is “QuickCommand Switch”

#### Y-Axis Mouse Movement Inversion

Long Tooltip[“Y-Axis Inverted”]: *Moving the Mouse away from you will make you to look downward, and moving the Mouse toward you will make you look upward. [Primary Action] to switch.*

Long Tooltip[“Y-Axis Not Inverted”]: *Moving the Mouse away from you will make you to look upward, and moving the Mouse toward you will make you look downward. [Primary Action] to switch.*

- Is a Toggle Menu Item
- State 1 is “Y-Axis Inverted”; State 2 is “Y-Axis Not Inverted”

#### Glass Fish Opacity

Long Tooltip: *A high Opacity means that a Possessed Fish will appear more solid. A low Opacity means that a Possessed Fish will appear more transparent and ghostly.*

- Spawns “Set Glass Fish Opacity Dialog Box”, which consists of:
  - A Numerical Text Field with a valid input range of 0.2 to 0.9
  - PushButtons “OK” and “Cancel”

#### Default Fish Mode POV

Long Tooltip: *When you first Possess a Fish, the camera view of the Fish will default to [Current Default Fish Mode POV]*

- Spawns “Set Default Fish Mode POV Dialog Box”, which consists of:
  - A series of PushButtons that enumerate all implemented Fish Mode POVs
  - PushButtons “OK” and “Cancel”

#### Client Permissions

Long Tooltip: *Anyone who joins your game remotely (either over a LAN or over the Internet) can only perform the game actions that you allow them to do. You can define different permissions for specific IP Addresses. The Guest Account enforces permissions over anyone whose IP doesn't match the IPs whose permissions you defined.*

- Spawns “Set Client Permissions Dialog Box”, which consists of:
  - Text: *Select an IP Address. Any Game Action that is checked is allowed for the selected IP Address. The Guest Account covers anyone who connects to your game whose IP does not match the IPs that you defined.*

- A Scrollable Text Field, where each Text Line contains either:
  - \* The text “Guest Account”, which is always the first Text Line, and is Highlighted by default
  - \* An IP Address
  - \* A Blank Text Line, which is always the last Text Line
    - [Primary Action]ing on the blank line spawns the “Create New IP Address” Dialog Box, which consists of:
      - Text: *Input the IP Address you wish to create an account for.*
      - A Text Field
      - PushButtons “OK” and “Cancel”. [Primary Action]ing “OK” checks IP for validity. If the IP is invalid, tell the player so in a Dialog Box with only PushButton “OK”. Otherwise create a new IP Address just above the Blank Text Line
  - \* An array of Check Boxes that are organized into vertical columns that represent their respective categories. There is also a “Category Check Box” above each Check Box column
  - \* Each Text Line of the Scrollable Text Field has the following functionality:
    - [Primary Action]ing on a “Guest Account” or an IP Address highlights that Text Line, and then all Check Boxes are updated to reflect the Permissions set for the Guest Account or the Client whose IP is currently Highlighted
  - \* Each Check Box has the following functionality:
    - [Primary Action]ing on one of the larger “Category Check Boxes” will cause all the Check Boxes beneath that Category to be set to its new state
  - \* Each Category Check Box has the following functionality:
    - If a Check Box is Checked, the User Interface Action it corresponds to will be activated for the Client with the Highlighted IP Address, or all Players who log on as Guests, depending on which text line is Highlighted. If a Check Box is not Checked, then the corresponding action will be disabled
- PushButtons “OK” and “Cancel”

#### Aquarium Dimensions

*Long Tooltip: Changes the height, length and width of the Aquarium.*

- Spawns “Change Aquarium Dimensions Dialog Box”, which consists of:
  - Three Numerical Text Fields, one for each dimension of the Aquarium, each must be in a valid range
  - PushButtons “OK” and “Cancel”

#### Fish Breeding Choosiness

Long Tooltip: *Defines how many other Fish a given Fish can potentially Mate with. A high number means that Fish will only be able to Mate with other Fish that are very similar to itself, whereas a low number means a Fish can Mate with many other Fishes. Remember that if it's impossible to Mate with another Fish, a Fish can still Mate with itself.*

- Spawns “Set Fish Breeding Choosiness Dialog Box”, which consists of:
  - Text: *A high number means that Fish will only be able to Mate with other Fish that are very similar to itself, whereas a low number means a Fish can Mate with many other Fishes.*
  - Numerical Text Field, which represents the minimum number of Fitness Functions that two Fish must be optimizing to allow them to chosen to Mate
  - PushButtons “OK” and “Cancel”

#### Fish Minimum/Maximum

Long Tooltip: *Defines the minimum and maximum number of fish to allow in the simulation.*

- Spawns “Set Minimum/Maximum Fish Dialog Box”, which consists of:
  - Text: *A higher maximum number of fish may produce more interesting evolution, but at a lesser frame rate. A lower minimum fish count may make it more difficult for predatory fish to survive.*
  - Numerical Text Field, which represents the maximum number of fish in the simulation.
  - Numerical Text Field, which represents the minimum number of fish in the simulation.
  - PushButtons “OK” and “Cancel”

#### System Options

##### Volume Setup

Long Tooltip: *Alter the master volume, as well as the relative volumes of sound effects and background music.*

- Spawns “Volume Dialog Box”, which consists of:

- Three Sliders: one for the Relative Volume of SFX, one for the Background Music, and one for the Master Volume
- PushButtons “OK” and “Cancel”

### MP3 Mapping

- Spawns “MP3 Mapping Dialog Box”, which consists of:
  - Long Tooltip: *Allows you to decide which mp3 files are played as background music.*
  - Text: *Each layer of the interactive soundtrack should evoke a level of emotional intensity Layer 1 should sound very relaxing, up to Layer X, which should sound nailbitingly aggressive, with other Layers being somewhere in between. [Primary Action] on a Layer to define an .mp3 file to play for this layer.*
  - X PushButtons, each corresponding to Layer 1 through X
    - \* When [Primary Action]ed, each PushButton spawns a “Standard Windows Construct” which the player is then able to use to specify an mp3 file for each layer
  - PushButtons “OK” and “Cancel”

### Set Title Screen Aquarium

Long Tooltip: *Lets you decide which Aquarium is used as the background for the Title Screen.*

- Spawns “Set Title Screen Aquarium Dialog Box” which functions identically to the “Load Aquarium From File Dialog Box”
- If a new Aquarium is chosen, the player [Primary Action]s the PushButton “OK”, then the selected Aquarium runs in the background as described under “Title Screen”

## 11 Title Screen

*Launching his brand new Aquarium, carefully tuned to his exacting specifications, the player sets off on a bold new aquatic adventure.*

Every game needs to start somewhere, and the Title Screen provides the player with a dynamic, striking glimpse at the possibilities that lie beyond it.

- Game logo, team logo, Digipen logo and scrolling credits will be superimposed over a (Player defined) Title Screen Aquarium. The Aquarium will be running as usual, with the exception that the only in-game user input that is accepted is the user input concerned with God Mode  $\longrightarrow$  GUI Mode  $\longrightarrow$  Pull-Down Menubar.

## 12 Networking

*After spending hour after engrossing hour husbanding his Aquarium's artificial life forms to fruition, the player decides to invite some friends into his watery domain. Setting his Networking options such that remote players can move about and Possess Fish, but not take any other (possibly destructive) action, he launches his Aquarium, looking forward to showing his friends around.*

At any time, remote players are able to dive into an already existing game. The host player is able to clearly state how much control each remote player has over his game, providing limitless possibilities for cooperation, competition or just taking a look around someone else's Aquarium.

- Networked games will use a client-server model based on TCP/IP.
- When a Client connects to a Server, that Client become a player in the Servers Aquarium.
- When the Server connects to their own Aquarium, they connect as the Administrator with all User Interface Actions enabled.
- A Client whose IP is not explicitly accounted for in the Client Permissions connects as a Guest.

## 13 Graphics

Enthralling, aquatic, lush, and visceral, Aquarius' graphics will communicate not only the player's position in space but also his place in the teeming multitude of underwater life.

- All 2D and 3D graphics in Aquarius will be rendered using OpenGL.
- The graphics engine will sport the following features:
  - 3D polygon rendering (for the Fish, Objects and Terrain inside the Aquarium)
  - 2D sprite rendering (for the GUI)
  - Texture mapping
  - Phong Shading
  - Transparency
  - Highlighting of models in arbitrary colors
  - Distance Fogging
  - Lightsourcing
  - Reflecting Fish Attributes graphically
    - \* Fish with a high Metabolism value will have faster moving fins
    - \* Fish with a high Armor value will have bumpy, tough-looking skin.
    - \* Fish with a low Teeth/Baleen value will have a sieve-like, baleen-oriented mouth, whereas Fish with a high Teeth/Baleen value will have sharp, predatory teeth
    - \* Fish will be uniformly scaled in accord with their Size
    - \* Fish with low Eyes (night-oriented vision) will have glowing eyes
    - \* Attributes will be associated with colors, such that a Fish's texture maps' color schemes will be algorithmically altered in accord with that Fish's Attributes

## 14 Sound

Evocative, context-sensitive sounds set the stage while an interactive soundtrack that changes emotional tenor to match the on-screen action completes the emotional experience of playing.

- 3D Spatially Oriented Sound Effects
  - “Sound Nodes” are randomly placed in the Aquarium
    - \* Bubbly noise (5 varieties)
    - \* Wave noise (5 varieties)
  - Fish Landing Attacks (5 varieties)
  - Eating Noises (5 varieties of Baleen Noises and 5 varieties of Teeth Noises)
  - “Affirmative Click” (GUI)
  - “Bad Buzzer” (GUI)
- Music
  - If the player has Toggled Background Music On, a 7 minute track is looped through.
  - If the player has Toggled Background Music On, an Interactive Soundtrack is constantly played. The Soundtrack changes emotional intensity based on the in-game action as perceived by the Player.
  - The Player can map any stream of music to an arbitrary mp3 file

## 15 3D Physics

The world of the Aquarium is governed by the untenable laws of physics – only the player (in God Mode) is exempt from the laws of gravity, mass and fluid buoyancy.

- Fish Buoyancy (live and dead)
- Fish Movement
- Fish/Objects Colliding with Other Fish/Objects (as a result of attack or simply bumping into one another)
- Fish/Objects Colliding with the Terrain

## 16 Comparative Analysis

### Creatures

The Creatures series has been one of the greatest examples of the successful use of machine learning for entertainment purposes. In each game, the player is allowed to supervise an intelligent troop of furry creatures called Norns. Each Norn is capable of learning, exploring, exhibiting basic emotions, breeding, and dying. The main feature that helped keep Creatures from just becoming another Tamagotchi is the ability for the physical statistics of a bloodline of Norns to change over time, thus allowing them to adapt to their environment. One of the main drawbacks of Creatures design was that some players became too attached to their Norns, thus becoming dissatisfied once their favorite Norns died.

Aquarius will be able to take advantage of the positive features of Creatures, namely machine learning, and the ability for the player to alter the physical characteristics of the fish they interact with. Problems associated with personal attachment to particular entities should never occur in Aquarius however, primarily due to the choice to use forms that are difficult for people to personify. This lack of personification does not necessarily mean that players are not capable of becoming attached to particular species or the environment as a whole, both of which are long lived.

### Aquanauts Holiday

Aquanauts Holiday allowed a player to enter an underwater world teeming with life. The player could build a reef, explore the surrounding ruins, and interact with numerous aquatic species by means of different sonar pings. A key feature was that the game had no specific goal, which in many reviews was heralded as the main reason that the reviewer enjoyed the game. Much of the fun of the game was that the player has the perspective of a fish, and thus could enjoy such activities as swimming with a school of fish, a feature that is also supported in Aquarius.

Nearly every review of Aquanauts Holiday mentioned that the game became boring once the reviewer had triggered nearly every creature's scripted reactions and explored the game's only map. Aquarius will avoid this pitfall by allowing the player to still interact with a goal-free environment, but one that will evolve and learn, thus overcoming Aquanauts Holiday's greatest limitation: limited content.

## 17 Marketing Analysis

The following table shows the total amount it will cost to create the game "Aquarius." This does not include marketing, packaging, or selling the game.

Item	Cost	Months/Amount	Total
<i>Team Members</i>			\$192,800
Producer	\$5,500	8	\$44,000
Designer	\$4,100	8	\$32,800
Testing Manager	\$3,500	8	\$28,000
Technical Director	\$5,500	8	\$44,000
Product Manager	\$5,500	8	\$44,000
Outside Development			\$5,000
Composed Music	\$5,000	1	\$5,000
<i>Software</i>			\$6,292
Borland C++ Builder 6.0 Prof	\$999	1	\$999
Visual C++ 6.0 Prof	\$499	5	\$2,495
Visual Source Safe 6.0	\$549	1	\$549
FMOD Sound Engine	\$1,250	1	\$1,250
Microsoft Project Prof.	\$999	1	\$999
Total			\$217,092

If the game is to be marketed well, an additional \$100,000 needs to be spent. This brings to total up front costs to \$317,092. Since the game is expected to be sold through retail outlets, the expected profit per game sold is going to be approximately \$10.00. This means we will have to sell over 30,000 titles to make a profit.

## A GUI Primitives

Listed here are all the GUI Primitives that will be needed for implementation of Aquarius.

- Any Game Action can be mapped to the Left Mouse Button, the Right Mouse Button, the Mousewheel Click, or any Key on the Keyboard Mouse
- Mouse
  - Visual Mouse Cursor
  - [Primary Action]ing, [Secondary Action]ing
  - Holding down the Left, Right or Mousewheel Button
- Text
  - Arbitrarily colored
  - Able to be Selected by line
- Text Field
  - Can have a numeric range of valid inputs inputting a number outside that range results in the input being reset to the nearest valid value
- Horizontal Sliders
- PushButtons
  - With an arbitrary bitmap inside
  - Can be Connected with Drawn Lines of Arbitrary Color
- Bar Graphs
  - Vertical and horizontal
  - Arbitrary color
- Doubly Layered Tooltips on any PushButton, Menu Item, Text or Text Field
  - Short Tooltip: If the Mouse Cursor hovers over said Tooltipped GUI Primitive for more than a fraction of a second, Short Tooltip's italicized text appears just above the Tooltipped GUI Primitive.
  - Long Tooltip: If the Mouse Cursor hovers over the Tooltipped GUI Primitive for more than a few seconds, Long Tooltip's italicized text replaces Short Tooltip's italicized text.
- Pull-Down MenuBar

- Checked Menu Items (i.e. the Menu Item is either “On” or “Off”).
- Toggle Menu Items

- Hierarchical CheckBoxes

- A series of nested categories of Check Boxes. All Check Boxes operate as per usual, with the following addition: checking or unchecking a Category Check Box causes every Check Box nested in that Category to also become checked or unchecked. For example, if “Humans” is a Category Check Box, and “John Corpening” and “Nathan Frost” are nested Check Boxes, checking the “Humans” Check Box also causes the “John Corpening” and “Nathan Frost” Check Boxes to become checked.

- Asynchronous Windowing

- Vertically scrolling Text
- Vertically and horizontally scrolling PushButtons
- All the above UI Primitives (excepting another Window) can appear in a Window (as in Dialog Boxes)

## B Game Actions

Each of these Client Actions can be permitted by the player who sets up the Server game. They will be structured in the following hierarchial fashion.  
Note that Load Aquarium can NEVER be performed by a Client.

### All Client Actions

- Edit Fish
- Alter Aquarium
  - Edit Terrain
  - Change Duration of Day/Night
  - Drop Fish In
  - Drag Fish Out
  - Drop Object In
  - Drag Object Out
  - Change Aquarium Dimensions
  - Set Fish Breeding Choosiness
- Gameflow
  - Pause and Unpause Aquarium
  - Go to Offline Mode
- Save Aquarium
  - Save Fish
- Possess Fish
  - Eat
  - Bite
  - Pick Up
  - Drop
  - Mate
  - AutoSwim

### Default Key Mappings

[GUI Visibility]	Right Control
[Move Forward]	Directional Arrow Up
[Move Backward]	Directional Arrow Down
[Alter Z Position]	Middle Mouse Button
[Alter Orientation]	Right Mouse Button
[Primary Action]	Left Mouse Button
[Secondary Action]	Right Mouse Button
[Eat]	Directional Arrow Right
[Bite]	Directional Arrow Left
[Pick Up]	Numpad 0
[Drop]	Numpad 1
[Mate]	Right Shift

## C General Appendix

Listed here, for convenience, is a variety of terms (and their definitions) used commonly in this document.

### Fish Attributes

The internal traits that define a Fish’s physical makeup.

### Fish Statistics

The textual representation of Fish Attributes.

### Gameplay Modes

The Modes of gameplay are referenced in the following manner:

In General: (Mode A)  $\longrightarrow$  (SubMode of A)

For Example: God Mode  $\longrightarrow$  Movement Mode

This differentiates, for example, Movement Mode, the submode of God Mode (God Mode  $\longrightarrow$  Movement Mode), from Movement Mode, the submode of Fish Mode (Fish Mode  $\longrightarrow$  Movement Mode).

### Load File Force Pathname

Load File Forced Pathname Dialog Box consists of:

- A Text Field listing every valid file from the preset directory that is available to be loaded
- Each file can be Highlighted with a [Primary Action]
- PushButtons “OK” and “Cancel”
  - [Primary Action]ing ”OK” while a file is Highlighted:
    - \* Spawns a “Sanity Prompt Dialog Box”
    - \* If “OK” is selected, the Highlighted file is loaded

### Networking

Administrator: A Player whose computer is acting as the Server for an Aquarium. Client: A Player whose computer is not acting as the Server for an Aquarium.

### P, N, K

P, N and K are generic constants used for calculating Fish Attributes.

### RandomFish

A Fish whose Attributes and AI Architecture has been randomly generated.

### **Red Highlighting**

Player actions that require a subject (“Follow Fish,” for example) use the current Red Highlighted Fish or Object as their subject.

### **Sanity Prompt Dialog Box**

A Dialog Box asking the player if he really wants to go through with the action he chose, and informing him of any possibility of information loss (for example, “Would you like to save your Aquarium?”). PushButtons “OK” and “Cancel” are available.

### **Save File Forced Pathname**

Save File Forced Pathname Dialog Box consists of:

- A Text Field displaying the name of the preset Save File directory
- A Text Field that defines the filename of the file to be saved
- PushButtons “OK” and “Cancel”
  - [Primary Action]ing “OK” causes the file to be saved under its defined filename

### **Standard Windows Construct**

A fully functional Windows “Save/Load File” style Dialog Box, which enables the player to specify an arbitrary file path and filename. It consists of:

- A Window with a Horizontal scrollbar that is capable of graphically displaying, and, when [Primary Action]ed upon, Highlighting, Files
- Text Field informing the player of the Window’s function (Save or Load a given entity)
- A Text Field that defines the filename of the file to be saved/loaded PushButtons “OK” and “Cancel”

### **Suckability**

A Fish’s efficiency at consuming Plankton.

### **User Interface Action**

Any “verb” the player is able to express to the game.

## D Team Sign Off Sheet

---

<b>John Corpening</b>	January 13, 2003
-----------------------	------------------

---

<b>Nate Cleveland</b>	January 13, 2003
-----------------------	------------------

---

<b>Jared Finder</b>	January 13, 2003
---------------------	------------------

---

<b>Nathan Frost</b>	January 13, 2003
---------------------	------------------

---

<b>Austin Spafford</b>	January 13, 2003
------------------------	------------------