CART353
Samuelle Bourgault
January 31th, 2017

Research evidence & UML diagram

1.The Concept

This project is a simulation of an abstract computational species populating the environment of a screen. In contrast to biological studies, this experiment doesn't try to simulate existing population dynamics, it only refers to itself, it is a pure simulation. The species evolves then according to rules that are computer-specific. These rules emphasize the basic concepts of programming such as: inheritance, polymorphism, encapsulation, parent/child relationship, etc. Instead of thinking in terms of mating, fighting, eating, the digital species interacts by connecting, disconnecting, absorbing and rejecting. The interaction between phenotype and genotype as understood in biology is maintained: each organism owns a specific genotype but their phenotype (appearance and behavior) is influenced by its experience through the screen, adding a bit of randomness into the simulation.

The three principles of Darwin's evolution are used to develop the project. In first comes the **heredity**; the digital organism needs a way to populate the screen with its own genotype. Then a notion of **variation** in the traits of the organism is necessary for computational selection to occur. And finally, the **selection** happens based on environmental settings. Here the environment depends on the activities of the computer. A vector field will be mapped on different indexes such as the energetic requirement, the network activity, the memory and the disk usage. Digital potential wells will appear on the screen and influence the displacement of the individuals.

The Goal of the Organism
Like Daniel Shiffmann, in Chapter 6 of *Nature of Code*, defines his goal as: "breath[ing] life into the things that move around our Processing windows", the purpose of the autonomous agents living in the computer screen is to create a superorganism that strives to create connectivity between the individuals, a web of knowledge that contains a genetic diversity. The rules are set to optimize uniqueness among the individuals. They don't survive either proximity or loneliness, so connections will rather be flexible than dense.

2.The Resources for Code

Daniel Shiffmann' s *Nature of Code* will be of great help in the conceptual structure of this experiment as well as in the code used for a digital ecosystem.

- Autonomous agents: Computational paradigm to create the organism http://natureofcode.com/book/chapter-6-autonomous-agents/
- Evolution of Code: http://natureofcode.com/book/chapter-9-the-evolution-of-code/

Michel Resnict

- Conceptual ideas on evolution in code: http://www.uvm.edu/pdodds/files/papers/others/1994/resnick1994b.pdf
- Exploring the emergence: http://www.playfulinvention.com/emergence/contents.html

3.The Inspirations in Biology

Even though this experiment takes some distance with biological lives, the inspirations remain in the living-being sphere. The digital species can be seen as a eukaryotic organism proliferating in agar.

- Bacteria evolution in contact with high dose of antibiotics: https://www.youtube.com/watch?v=plVk4NVIUh8
- New Growth/Exploration of morphogenesis: http://www.creativeapplications.net/c/hybrid-forms-new-growth-exploration-of-morphogenesis-by-andy-lomas/
- Genotype: https://en.wikipedia.org/wiki/Genotype
- Phenotype: https://en.wikipedia.org/wiki/Phenotype

4.Similar works by artists

Karl Sims, for the use of evolution in code

- Evolved Virtual Creatures: https://www.youtube.com/watch?v=JBgG_VSP7f8
- Evolved Virtual Creatures: http://www.karlsims.com/evolved-virtual-creatures.html
- Galapagos: http://www.karlsims.com/galapagos/

John Conway, *the Game of Life,* for the use of cellular automata, another paradigm in modelling platform that could help me understand the interaction between autonomous agents and their environment

- https://www.youtube.com/watch?v=C2vgICfQawE

Michael Noll, for the aesthetic of connectivity:

- http://dada.compart-bremen.de/item/agent/16

Georg Nees, for the aesthetic of connectivity:

- http://dada.compart-bremen.de/item/agent/15

Piratefsh, for particles' behavior and connectivity:

- https://github.com/piratefsh/generative-art

Onformative studio, for general aesthetic:

- http://onformative.com/work/immaterials

5.The Scope

The *Nature of Code,* Daniel Shiffmann introduces multiple exercises for the readers but one of them goes on through the whole book. He calls it “The Ecosystem Project” and add elements to it as he teaches a variety of physical concepts to the readers. This experiment will follow partially the structure Shiffmann gives to this exercise keeping the elements that inspire me to create my own digital ecosystem.

The Organism

a) The first organism: I will start by creating one type of creature that will have a really basic appearance to focus mostly on the computational work. As minimalist as I can, the creature will acquire a shape using the library Box2D to give a realist aspect of the whole simulation.

b) The creatures will all have a genotype (DNA) and a phenotype. The latter will behave according to the genotype, a bit of randomness and its interaction with the environment. The organism will reproduce in an asexual way like eukaryotes or bacterias, using DNA replication. This is an aesthetic choice as well as a way to reduce the code. Every organism will have only one parent.

c) Each instance of organisms will be a “cellular automaton” as theorize by Daniel Shiffmann in the Chapter sharing the same name. This cellular automaton will be influenced by the environment rather than by the addition of predators in the screen. I will not worry about adding a prey/predator relationship.

d) The main goal for the organisms will be to stay connected. Therefore the “alive” state will depend in the capacity of each individual to connect to each other.

e) If I have some time left, I will create other species to add diversity and complexity to the interactions.

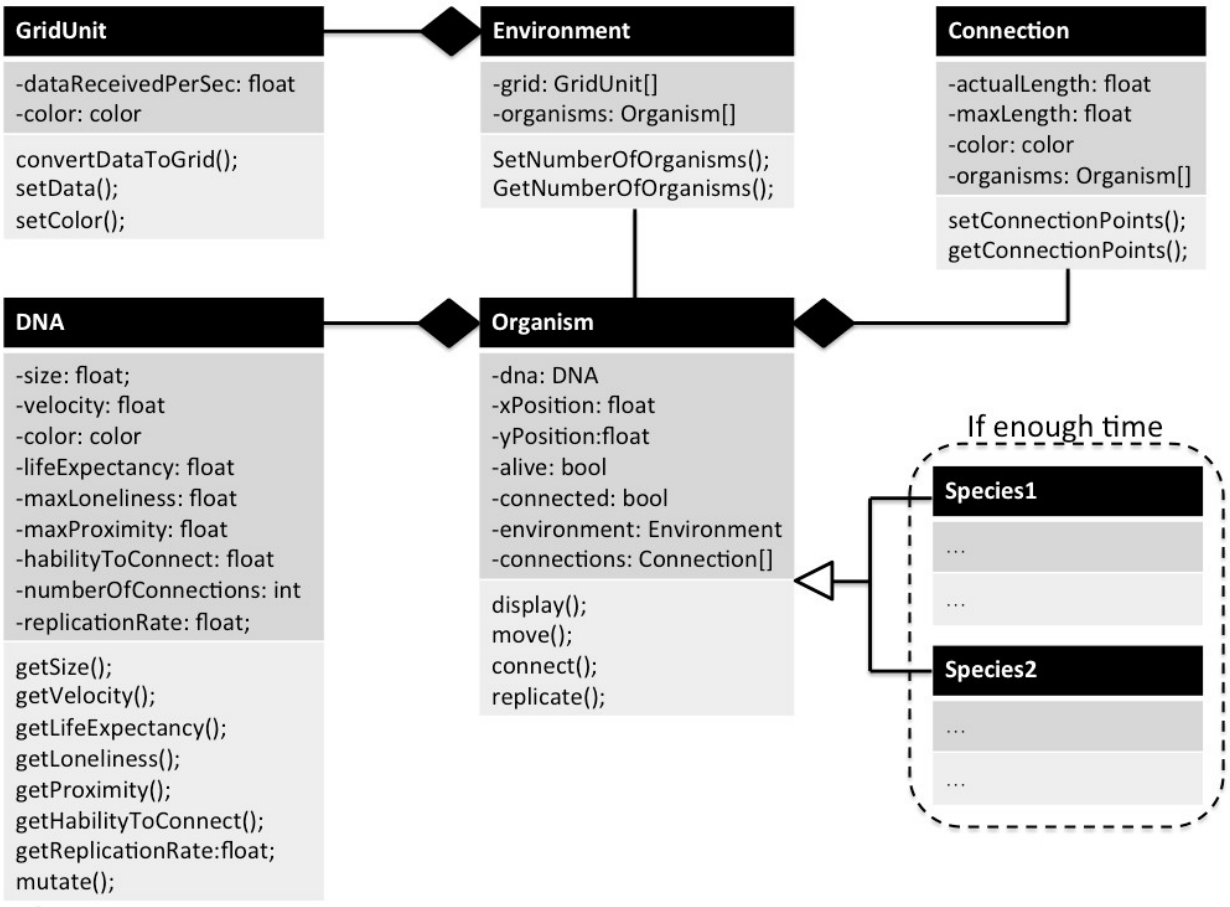
The Environment

a) I will start mapping only the network activity of the computer while running the program. If I have more time, I will increase complexity in adding dependencies of the environment on the CPU activity, the memory and the disk usage.

b) I will try to access the network activity in real time but if I don’t have time for it, I will record data and create the environment from there.

c) The environment will be mapped as a vector field with minima and maxima. The organisms will travel naturally towards the wells following the lower energy state rule.

6.UML


GridUnit
-dataReceivedPerSec: float
-color: color
convertDataToGrid();
setData();
setColor();
Environment
-grid: GridUnit[]
-organisms: Organism[]
SetNumberOfOrganisms();
GetNumberOfOrganisms();
Connection
-actualLength: float
-maxLength: float
-color: color
-organisms: Organism[]
setConnectionPoints();
getConnectionPoints();
DNA
-size: float;
-velocity: float
-color: color
-lifeExpectancy: float
-maxLoneliness: float
-maxProximity: float
-habilityToConnect: float
-numberOfConnections: int
-replicationRate: float;
getSize();
getVelocity();
getLifeExpectancy();
getLoneliness();
getProximity();
getHabilityToConnect();
getReplicationRate:float;
mutate();
Organism
-dna: DNA
-xPosition: float
-yPosition:float
-alive: bool
-connected: bool
-environment: Environment
-connections: Connection[]
display();
move();
connect();
replicate();
If enough time
Species1
…
…
Species2
…
…