Nathan Koch
Major Studio I Final Paper
December 4, 2017

Background and Purpose

The stories I love were written, designed, and coded by people, but many of my most memorable experiences emerged out of the fabric of city-life.

Stories and imaginary worlds, as we traditionally imagine them, are hand-crafted, populated painstakingly by authors or designers. What happens if, instead, we observe narrative agents in a persistent virtual space? Can happy accidents arrive through emergent systems?

I explored and studied these techniques and associated algorithms in order to create and populate "generative worlds." I believe that the combination of procedural content, algorithmically simulated behavior, and programmatic world-building can create something that is both like and entirely unlike a game; although it is built on game technologies and has the look and feel of a game, it isn't playable. There's a sense of place and users may infer their own story, but it will resist conventional narrative readings.

Context and Inspiration

Procedural content in games in particular can be defined as "algorithmic creation of game content with limited user input,"[1] and to date, it is an area of interest that has been driven primarily by the economics of large-scale game creation. For example, if game designers and artists mediate computer-generated work rather than create it by hand, they can produce more work in less time.

There's more to procedural content than efficiency. I'm not looking to speed up the traditional level design and AI behavior tasks that go into game creation; I want instead to create an *observable bubble of simulated reality* in which a set of pre-established parameters drive the motivations of virtual actors in unpredictable ways.

My approach has more in common with Brian Eno's rules-based works like Discreet Music or Music for Airports than commercial game development. These techniques have long been used in new

media art, but usually the system is fully exposed for the viewer – it *is* the piece. By comparison, experimental music is rules-based, but those rules are a tool through which a composer creates new work that can exist on its own merits. My intent with this piece is to effectively ambient storytelling instead of ambient music–in that a set of rules causes behavior which slowly emerges over time, rewarding both "up close" and "at a distance" styles of viewing.

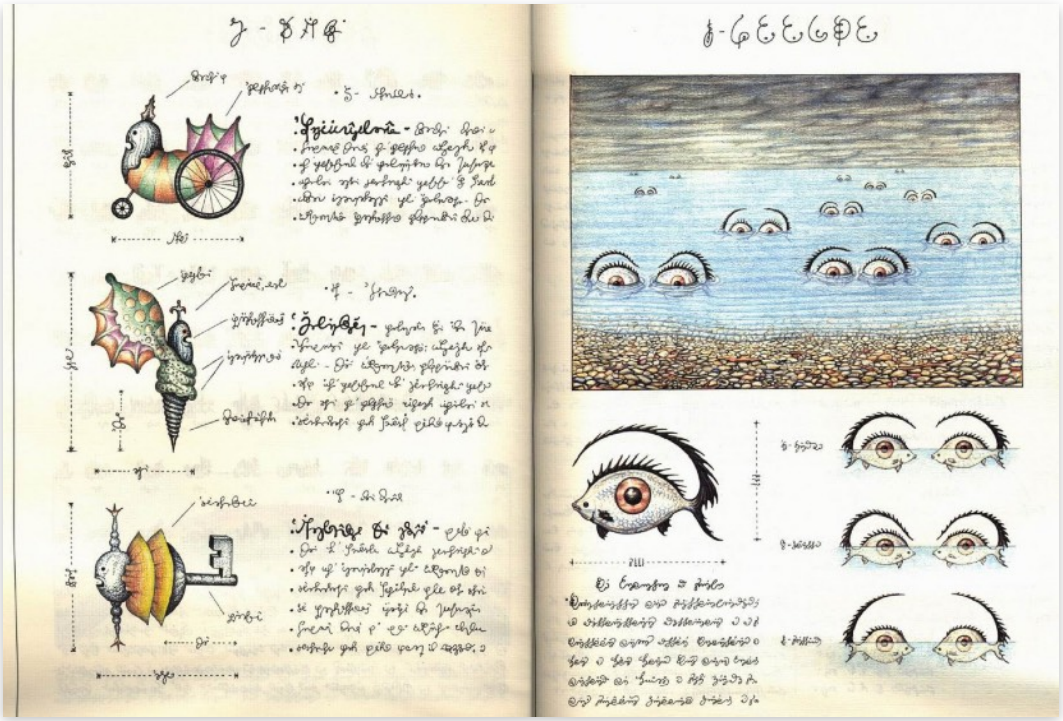
Emergent systems are larger than either music or the visual arts, however. Mitchel Resnick's book "Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds" shows real-world examples of emergent behavior. "Ant colonies, highway traffic, market economies, immune systems–in all of these systems, patterns are determined not by some centralized authority but by local interactions among decentralized components."[2]

Ian Cheng – *Emissaries*. Image courtesy MoMA.

One particular precedent that steered me in the direction of simulation-as-storytelling-tools was Ian Cheng's Emissaries exhibition at MoMA PS1, which he described as "video games that play themselves."[3] I found Emissaries striking and endlessly watchable, as a cast of AI characters played out their programmatic scripts in unpredictable ways.

Catching Cheng's exhibition before it closed in September inspired me to think about systems and simulations. What he and his team were able to accomplish was infinite, storytelling through winding something up and letting it go.

His work was steeped in video game tropes while living completely outside the context of games as we know them – narratives emerged secondarily out of a collection of colliding systems and algorithms. "Unlike conventional animations, whose story arcs are predetermined, Cheng's digital worlds are driven by a technology in which narrative agents behave within coded parameters, but on their own terms. This means that the action is unpredictable and unfolds at a lifelike pace, which is to say it's both absorbing and tedious."[4]



Luigi Serafini – *Codex Seraphinianus*. Image courtesy Dimka Daily.

In an entirely different direction, a very different precedent for this project is the Codex Seraphinianus.[5] The Codex is a profoundly strange book dating back to the late 1970's, an encyclopedia of an imaginary world documented extensively in an artificial language. It's filled with grotesque organic imagery, utterly unlike our own world but similar enough to evoke emotions ranging from disgust to wonder. Serafini effectively created a virtual world using entirely personal,

analog methods, but we can see parallels with both Cheng's work and modern video games like *Skyrim*.

Alan Warburton – *Primitives*. Image courtesy Alan Warburton.

Alan Warburton's Primitives series explores the possibilities of programmatic, AI-generated behaviors applied to 3D forms and animation.[6] He uses crowd-simulation techniques applied to modeled human figures to achieve results that are both grotesque and affecting.

Much of Warburton's work is critical of how the medium of 3D animation is driven almost entirely by commerce, reflecting on how Hollywood continues to push the boundaries of photorealism in the service of producing massive spectacles for crowd-pleasing movies. He uses the same tools as big-budget game and movie studios but with an avant-garde, subversive bent.

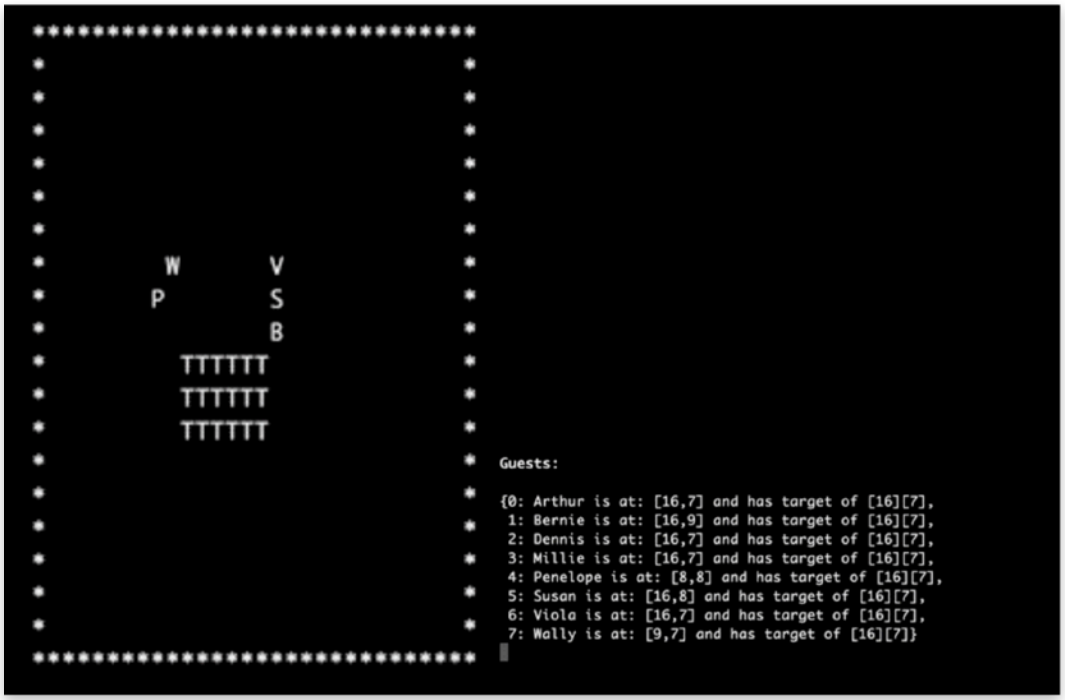
Intention and Creation

I have created a persistent, ambient game world (or, as described above, an observable bubble of simulated reality) using behavioral simulation and rules-based AI techniques to create an explorable virtual space where an observer can watch these algorithms unfold in real-time.

With this project, I felt it was important to see if I could effectively *back into* storytelling, by setting up a lightweight framework of space and character and letting the story emerge organically out of multiple systems interacting in unexpected ways. My intent was to observe the possibilities of story and setting derived from emergent systems, and expose my audience to this "ambient storytelling" as well.

The technology of games, much like the technology of 3D animation, has long been driven more by the commercial realities of large scale "AAA" titles than individual artists. I wanted to reclaim these tools for my own use, as an individual creator working apart from the goals of either large scale studios or solo indie game developers, to create a game-like object, or piece of toy software, completely separate from the usual sales-driven ecosystems like the PSN Store.

Over the course of this project I've read about and recreated a number of algorithms that may find a place in my project, found in sources ranging from Daniel Shiffman's "The Nature of Code"[7] to A.K. Dewdney's "The Magic Machine."[8]

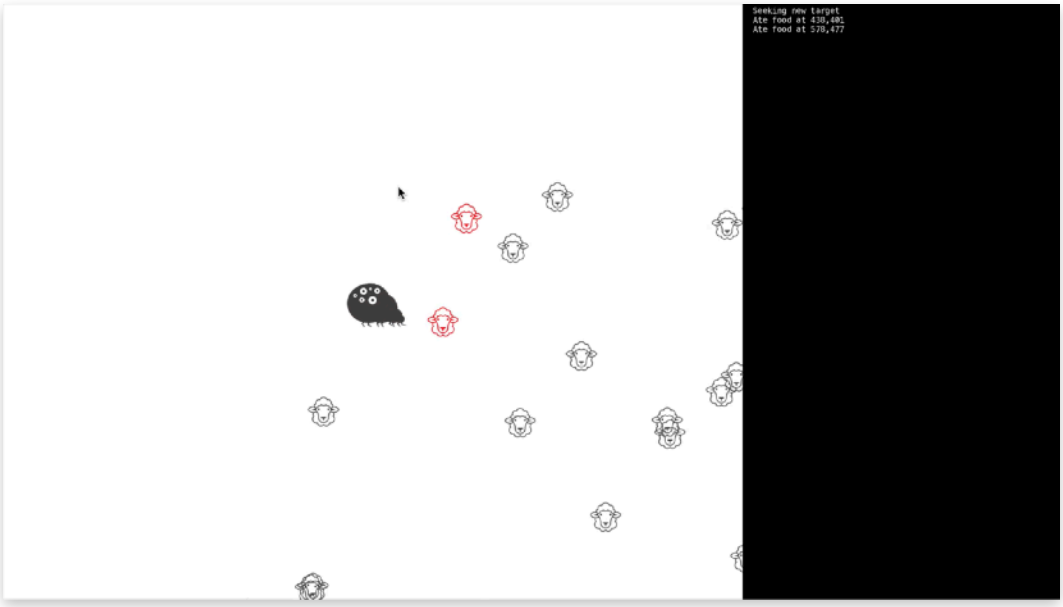


Nathan Koch – *Dinner Party Study*. Image courtesy Author.

I started with an initial study which recreated A.K. Dewdney's "Programmed Parties" (an algorithm from his column in Scientific American) in Python. "Programmed Parties"[9] simulates the guests at a dinner party, with each one being motivated to get closer or farther away from either each other or the food table. Although a fairly simple AI algorithm by today's standards, it created a seemingly ever-changing dynamic system.

Another study I created used Craig Reynolds' "flocking" algorithm[10] to drive a set of bird-like creatures around a screen according to a simple set of steering rules. When the birds intersect with sound-emitting clouds, they produce different notes, and, at times, harmonies.

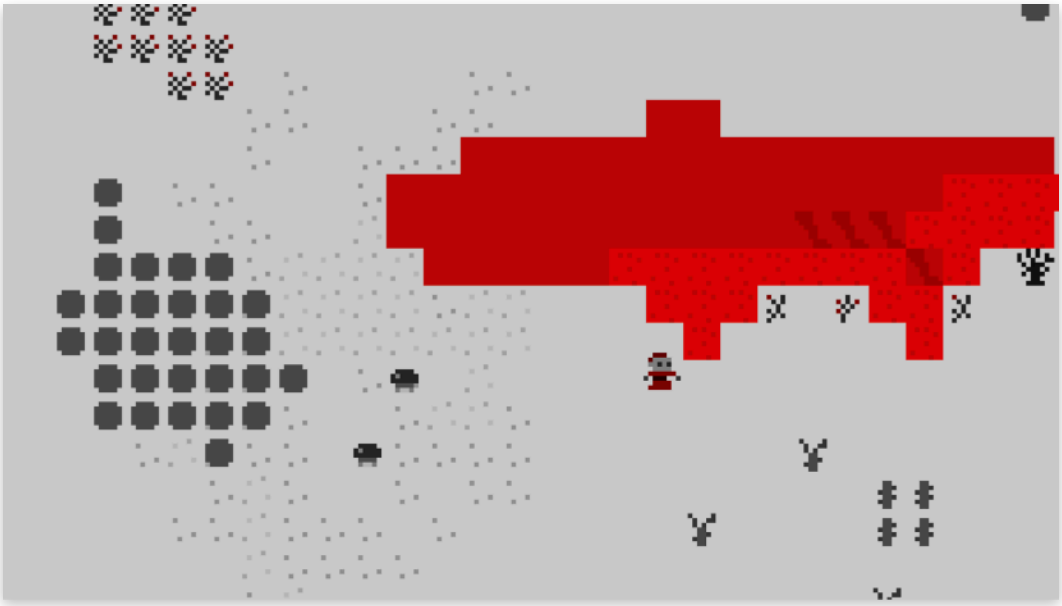
From there I attempted to add more complexity, and coded a prototype that used one of Reynolds' steering examples,[11] combined with a naive resource-gathering algorithm of my own creation. This prototype also contained a few different modes, including icon-based graphics, a visible text-based console exposing the motivations of the agents on-screen, and even simple text-to-speech audio.



Nathan Koch – *Resource Gathering Study*. Image courtesy Author.

Testing this prototype in-class, it became clear how powerful sound could be at grabbing observer attention. It became clear to me that giving agents in this virtual world unique voices was a window into their individual interactions.

When I began building the final product after iterating through a series of studies, I had to drastically reduce my intended visual fidelity to keep within the scope of the project. Although I believed a compelling scenario could emerge from interesting rules and algorithms, in the final work it was important to me to combine behavior and aesthetics in a unified whole.



Nathan Koch – *Ecosystem #1*. Image courtesy Author.

I chose a 2D, 8-bit execution of the project, heavily inspired by the Nintendo Entertainment System. Lo-fi pixel art that fit into 8 by 8 blocks was something I was capable of producing in the time available. Also, the aesthetics of mid-80's pixel art personally resonate with me, as I grew up alongside the ascendance of classic 2D video games.

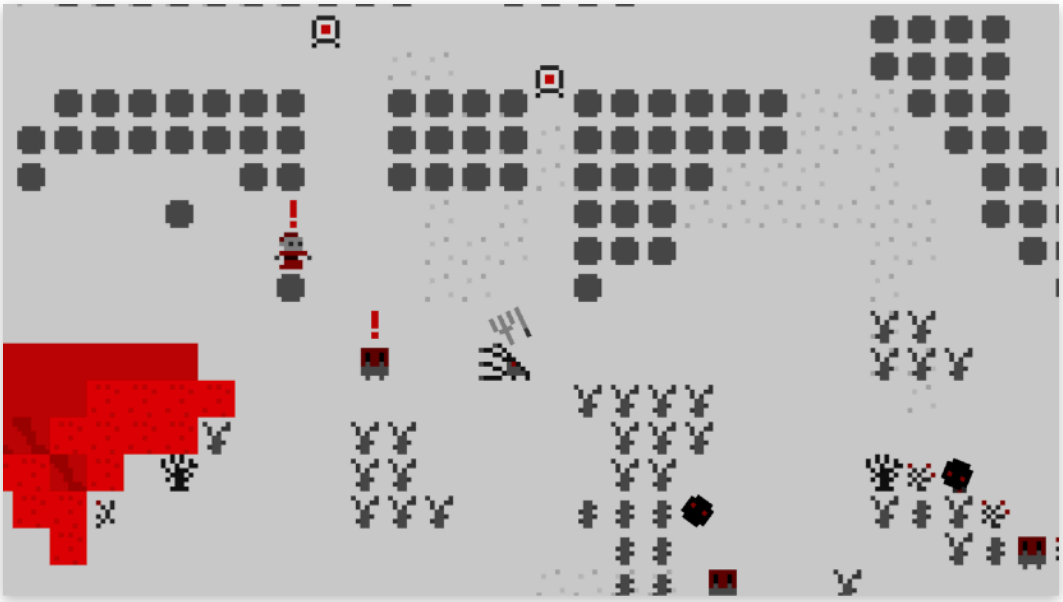
This visual approach also differentiates this piece significantly from the Cheng and Warburton precedents I outlined above. Those pieces were as much about their visual beauty as the underlying behaviors behind the world – in much of Cheng's work the behavioral simulation was masked by models detritus of the modern world. This project's pared down visual language allowed for a "close to the metal" approach that gives the viewer a more clear sense of the underlying behaviors behind the world.

There were a vast range of off-the-shelf game AI algorithms to choose from for the purposes of this project, but I focused on a small subset that supported my goals. I used Craig Reynold's steering

algorithms for some characters, as well Hart, Nilsson and Raphael's A* pathfinding for a number of others. I managed character motivation and movement via a finite state machine.

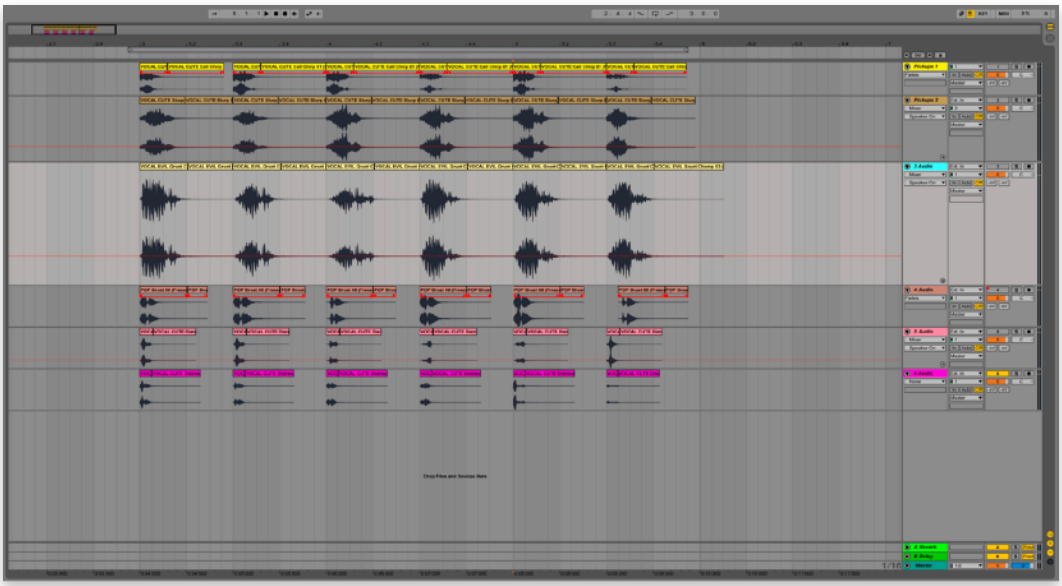
Like many game AI programmers, I also wrote a great deal of custom code to create specific behaviors I wanted to achieve. Some particularly interesting features were a naive resource gathering algorithm, where each character will seek out and eat any food within a supplied radius, and a predator/prey algorithm with flee behavior, influenced in part by Dewdney's dinner party project.

By adding together all these disparate algorithms, 8-bit artwork, and the Unity 2D game engine, I ended up with a project which achieves what I might term "human-assisted emergence." Although I strove to give the game world some space to grow organically, it became clear that the activity was heavily informed by the underlying design framework I had created. By tweaking a few numbers (speed, alertness, hunger) the experience would be radically different.



Nathan Koch – *Ecosystem #1*. Image courtesy Author.

Some behaviors were truly unexpected – and might become the kernel of a new story. Large predators couldn't fit through small spaces, driving fleeing creatures into tightly crowded sections of the map. Predators also had a limited lifespan, which was sometimes exploited by fleeing/hiding creatures waiting them out, and staying just out of range until a predator died of starvation.



Nathan Koch – *Ecosystem #1* sound effects. Image courtesy Author.

It's worth also calling out the sound as supporting the larger experience of the project. Sound effects for nothing more than fleeing and eating gave the game world a striking amount of life. In order to add sonic interest, I created six iterations of each sound effect in Ableton Live. I also supported the piece with a dark, dissonant score of four chords in an endlessly random progression that brought up the level of underlying tension.

By creating this simulation within the boundaries and context of game art and technologies, I've framed it in a long familiar way, while also expanding the boundaries of what observers see a "game" or "not-game." It also gave me an opportunity to learn about the algorithms and techniques that lie behind the curtain of AI programming in modern game development.

Values and Goals

I felt it was important to include some specific design values beyond the larger domains of systems and simulations:

No human interaction

This project should be completely unmediated by human hands. What's possible at the bounds of pure procedural design, rather than merely parametric?

Affecting ambience

The world creates a sense of space that encourages restful observation.

Aesthetic poetry

The look and feel of this simulation is aesthetically coherent and finished. It feels less like a technical demo or prototype than a finished work.

Game-like objects

This world is compelling to a observer who doesn't understand the underlying concept. The world's mechanics evoke watching someone else play video games.

I believe that AI-assisted world-building and generative stories may be able to transform the roles of game designers, advertisers, filmmakers, and even authors. My hope is that this project demonstrates what's possible in algorithmic storytelling, and inspire additional work in the area of simulated virtual worlds.

Notes

1. Shaker, Noor, Julian Togelius, and Mark J. Nelson, *Procedural Content Generation in Games* (Cham, Switzerland: Springer, 2016), 3.
2. Resnick, Mitchel, *Turtles, Termites, and Traffic jams: Explorations in Massively Parallel Microworlds* (Cambridge, MA: Bradford Book, 1997), 1
3. "Emissaries," MoMa, accessed November 20, 2017, https://www.moma.org/calendar/exhibitions/3656
4. Scott, Andrea K, "Watch the Absorbing and Tedious Simulations of Ian Cheng." The New Yorker, May 16, 2017, https://www.newyorker.com/culture/culture-desk/watch-the-absorbing-and-tedious-simulations-of-ian-cheng
5. Serafini, Luigi, *Codex Seraphinianus* (New York, NY: Rizzoli, 2013)
6. Warburton, Alan. Primitives, Laboral, Gijon, Spain, 27 May 2016.
7. Shiffman, Daniel, *The Nature of Code* (New York, NY: Self-published, 2012)
8. Dewdney, A. K, *The Magic Machine: More Computer Recreations Columns from Scientific American* (New York, NY: W.H. Freeman, 1990), 151–160
9. Dewdney, A. K, *The Magic Machine: More Computer Recreations Columns from Scientific American* (New York, NY: W.H. Freeman, 1990), 151–160
10. Shiffman, Daniel, *The Nature of Code* (New York, NY: Self-published, 2012), 308–314
11. Shiffman, Daniel, *The Nature of Code* (New York, NY: Self-published, 2012), 262–269

Project Credits

Pathfinding courtesy of A Pathfinding Library for Unity. https://arongranberg.com*

Final paper editing courtesy of Terry Koch.