Dan Miller

Game Dev Enthusiast + Amateur Cook + Lifelong Learner

"Build **smarter** to go **faster**."

linkedin.com/in/the-daniel-miller/
fraculation.com

SKILLS

Backend/Systems

- Unity's Jobs + DOTS
- Multithreaded conversion
- C#, NodeJs, Sql
- Component and systems-oriented design

Tooling

- Unity editor extensions
- ShaderGraph, VFX Graph
- Designing custom editor authoring workflows
- Git and Perforce
- 3D modeling in Blender

Specific applications

- Procedural mesh generation
- Behavior tree NPC AI
- Scriptableobject config
- Real time simulations
- GPU acceleration

EDUCATION

BS Software Engineering
Milwaukee School of Engineering | 2017

EXPERIENCE

Seeb Defender Unity3D fraculation.com/blog/seeb-defender-project/
12/26/2020 – Current

- Farming simulation, tower defense, roguelite built around dynamic plant simulation
- Learned how to manage a large unity project and reach target 60FPS
- Used scriptable objects to accelerate game design and configuration
- Designed realism-inspired genome simulation (github.com/dsmiller95/Genetics)
- Skills: C#, Unity DOTS, 3D modeling, Performance Optimization

L-System language extension Unity3D fraculation.com/blog/lindenmayer-implementation
12/26/2020 – Current

- Highly optimized procedural mesh generation language
- Extended the Unity editor to compile custom language files into mesh generators.
- Used Unity's multithreading and mesh editing tools to increase performance.
- Skills: C#, Unity Jobs, Unity library management, Unit and Performance testing

AWS developer @ Milwaukee tool
12/14/2020 – 06/18/2021

- Near real-time data processing architecture
- Skills: C#, Python, Kinesis, Lambda, ECS, SQL, dynamodb, terraform

Spideyboi made with Unity2D playable at fraculation.com/Spideyboi/
07/31/2020 – 08/03/2020

- Weekend game-jam
- Player uses a step-based visual programming scheme to build instructions telling a Spider how to build a force-directed-graph "web"

Dimensional Rifter made with Unity2D: github.com/dsmiller95/Dimensional-rifter
08/08/2020 – 12/25/2020

- 2D colony simulation game in which the player designates build orders, and the colonists gather resources and build structures.
- Leverage Unity's DOTS system to maintain high performance when updating many agents at the same time.
- Create a custom Behavior Tree system as the AI of colony members.
- Skills: C#, DOTS, Unity's Jobs, 2D sprite design.