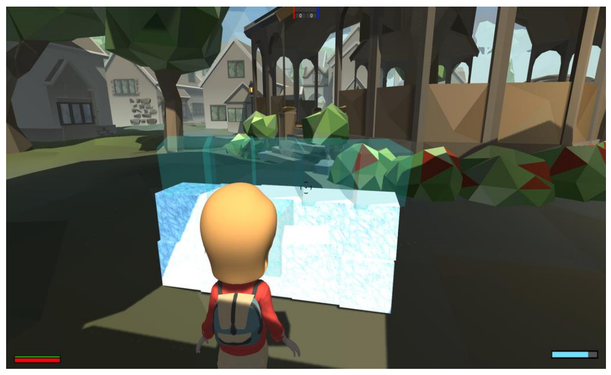
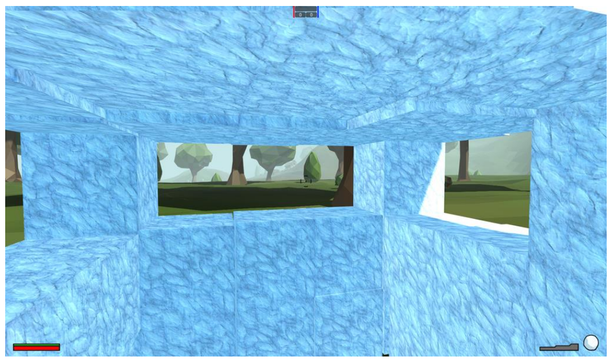
Snow Day

Paul Breuler, Jacob Edelen, RJ Kucia, Mitchell Loveall, Kevin Smearsoll, & Alan Thornburg





Building

- Snow Day features a building system that allows players to build walls and simple structures.
- When the build tool is active, players see a translucent block(s) in front of them allowing visualization of alignment. When the block is moved near surfaces it will attempt to “snap” to the surface; raycast is used to determine if snapping is allowed.
- Clicking and dragging the mouse causes a visual blueprint to be created. The player can right click to cancel or release the left mouse button to begin building.
- A priority queue is used to determine the order of blocks placed by the player. Lower blocks are given higher priority thus building the structure from the ground up.
- Each block has physics to prevent players from building outside of the map (i.e. unrealistic building).

Artificial Intelligence

- Mode based AI which is tailored to each game type including, Capture the flag, Domination and Team Deathmatch.
- AI players utilize “brain” classes to control movement and actions. There is one abstract “brain” class containing movements and patterns common to all game modes. For specific game modes this class is extended by child “brain” classes.

Controlling Actions:

- Each brain contains a number of goal states specific to a given game mode such as attacking, seeking cover, and pursuing objects.
- Movement and action scripting was completed with reusability in mind; as such portions of scripts were used for multiple states and modes reducing redundancy.
- Transitions between states are determined by factors such as events in game and position of the AI relative to the player(s). Random probabilities are used to transition states with higher probabilities assigned to “meaningful” states based on game conditions.

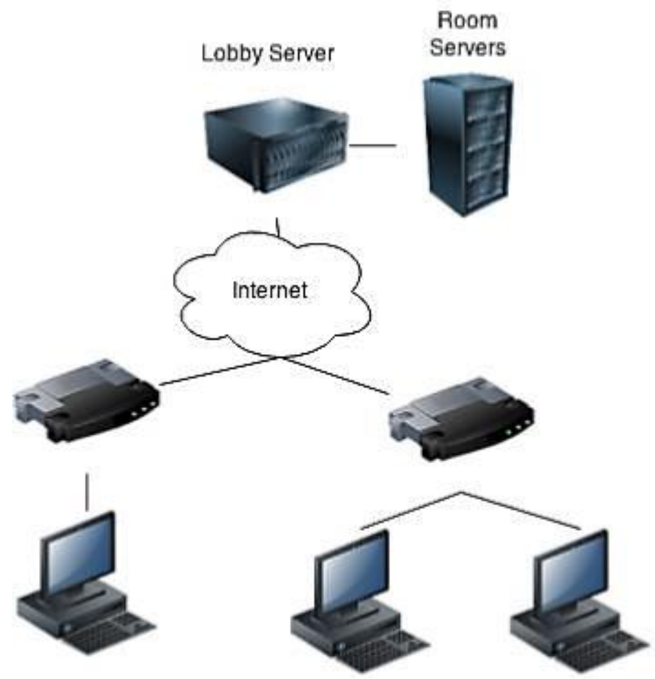
Sensing the Environment:

- Environment sensing is controlled via a simple algorithm
that generates an arbitrarily sized grid of points around any given point on the map.
- Generated points are used to test for certain heuristics such as visual cover from enemies, distance from enemies, and height advantages.
- A large sparse grid is generated to get a basic layout of the environment. After which a smaller, denser sampling grid is created around points of interest using the same algorithm. As a result the AI is able to efficiently sample points because only points of interest are being expanded and closely examined.

Modular Weapon System

- Weapons are managed by a modular system which at its base handles instantiating, displaying and network synchronization.
- Each weapon plugged into the system is extended from a base class which handles states, bullet spawning and basic UI.
- Individual weapons can be created by simply extending the base class and adding special considerations for the weapons properties and special effects.
- Weapon UI is the final component of the system which can be dropped into the individual weapon class slot and animated appropriately

Networking



Sound

- Each sound event in game has three to eight variances which are handled by FMOD and propagated using RPC calls to all clients ensuring all players hear the same sounds.

The Ohio State University
College of Engineering

SUPACOOL GAMING