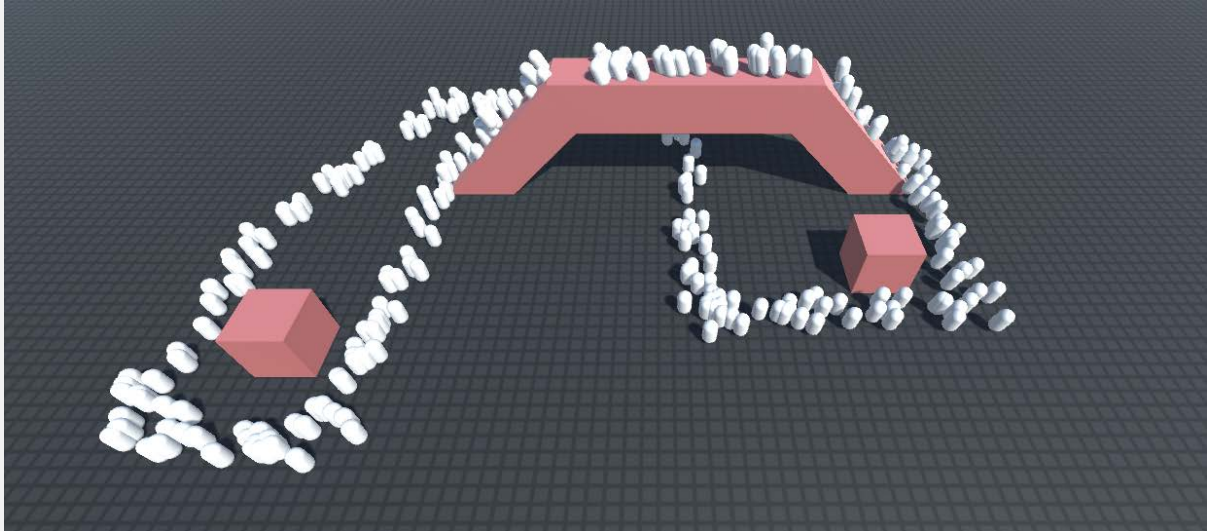


# RAPID WAYPOINT SYSTEM

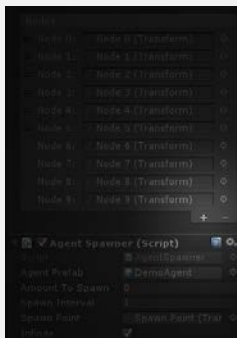
BY HENRY BELL AND FREDERIC BABORD



## QUICK START

1. Drag the waypoint manager prefab from RWS / Prefabs into the scene view or the hierarchy panel
2. [Add nodes](#)
3. [Configure the manager](#)
4. [Add agents](#)

## ADDING NODES



To add a node, click the '+' icon. This will add a new game object into the scene as a child of the current waypoint manager game object as well as append that newly created game objects' transform to the list.

If the node isn't in the right place in the list, it can be moved by dragging the handle to the right to the desired location.

The nodes can be placed on the floor as the agents will go to its location + half its colliders' height.

## REMOVING NODES

To remove a node, click the '-' icon at the bottom of the list. This will delete the corresponding game object from the scene.

## CONFIGURING THE MANAGER

Below are the properties that are in the manager. These properties demote how agents linked with the system will behave.

Property	Description
<b>Should Loop</b>	Joins the first and last nodes to create a circuit
<b>Update Interval (Obsolete)</b>	The amount of times to update the system per second
<b>Node Proximity</b>	The minimum distance before the agent moves to the next waypoint. <i>Note: this should not be set to 0!</i>
<b>Target Spread</b>	The distance around nodes the agent can chose to simulate randomness. At a value of 0, the gent will move directly to the centre of the target node.
<b>Agent Rotation Mode</b>	<b>None</b> The agent will <i>not</i> rotate to face its next waypoint
	<b>Snap</b> The agent <i>instantly</i> face its next waypoint
	<b>Slerp</b> The agent will smoothly interpolate to face its next point
<b>Rotation speed</b>	This is only visible if slerp is selected. This controls how fast the agent rotates to face the next waypoint.
<b>Nodes</b>	This is a reorderable list of the node transforms

## ADDING AGENTS

To add an agent you need to add this line of code **AT RUNTIME**:

```
m_waypointManager.AddEntity((GameObject)Instantiate(m_agentPrefab, m_spawnPoint.position, m_spawnPoint.rotation));
```

You can specify their target node by specifying the agents "CurrentIndex".

## CREATING A CUSTOM AI AGENT

The system was built with polymorphism in mind. So when you create a new agent class you need to inherit from **WaypointAgent**, so it inherits the base logic for traversing the system. Below is an example script of extending the agent.

## BUILDING A WAYPOINT PATH AT RUNTIME

You need to have a 2D array of GameObjects that have the “Tile” script attached before the point graph is generated. Once you have this, then the waypoint manager has a creation function:

`BuildNavigationMap (GameObject [,] map, int width, int height);`

A demo of how to both create and use this can be seen in the “Dynamic” scene in the demo folder.

## EXPLORING THE DEMO SCENES

### DEMO

In the demo scene there is a waypoint graph already set up. Agents will spawn in a constant rate and traverse the graph. On the Waypoint Manager GameObject, there is the waypoint manager script along with the agent spawner script.

### DYNAMIC

In this demo scene, a simple tile grid is generated at runtime, (this can be loaded in from a text / xml file if you prefer). The navigation map is then built and the agents are then created.

## LICENCE

We are happy for you to be use this system in commercial projects. We only ask that you credit the developers (Henry Bell & Frederic Babord).

## KNOWN BUGS

- When deleting a node from any place other than the reorderable list in the waypoint manager, it is not removed from the list and Unity throws an editor UI error.