POPULATION ENGINE

OWNER'S MANUAL

Thank you for purchasing Population Engine!

Population Engine makes your game world come alive – it *breathes life into your cities, villages, and dungeons.* You can potentially place hundreds of characters and objects into a scene in minutes.
It can help you build entire levels in hours, not weeks or months!

INTELLIGENT OBJECT PLACEMENTS
PE uses intelligent methods to “paint” your population around obstacles such as buildings, tables, rocks, etc… It then powers your NPCs to automatically move around your scene to simulate a living population.

NOT JUST FOR NPCs or PEOPLE
Remember, this tool can be used to mass-generate anything you want, including Foliage, Rocks, Props, Buildings, etc… not just People and NPCs. You can even load Building prefabs into it and generate a village with a few clicks!

RUNTIME & PERFORMANCE MANAGEMENT
Generate objects at runtime and keeps track of them. Use triggers to only spawn objects when the player moves into a specific area, increasing performance.

MANUALLY PAINT IN THE EDITOR
Even if you're not using the runtime generation tools, you can still use this as part of your level design workflow by painting right into the editor. For example, you're designing a forest and want to fill it with 100 randomly placed rocks, 50 animals, 50 people, etc... It would take a long time to insert each by hand. Simply load those prefabs into the Population Generator, set the area of effect, and click [GENERATE], right in the editor!

- FOR VIDEO TUTORIALS, GO TO OUR WEBSITE: www.AIBotSystem.com -

CONTENTS

{1} HOW DO I... OTHER CREATIVE USE CASES / EXAMPLE USAGE

None of these require any coding. All assume you understand the Basic Setup steps in later chapters.

How do I spawn a fleet of ships on the water?
Give your water plane a mesh collider (or you can place a transparent collider under it). Tag it something like “Water”. Then enable Obstacle Awareness and insert the water tag. Place your spawner on the water.

How do I mass-generate objects on a terrain, right at the edge of a river or water?
Create waypoints along the edge of the water. Using the waypoint lines as a visual guide, try to match the curves and shape of the edge of water. Insert many waypoints along the edge.

- On the Population Generator, set Area Type to “Spawn At Waypoints”
- Set “Spawn Radius” to a small number, like 3, to avoid spawning into the water.
- Insert your houses and people
- Set Object Height Position to “Use Raycast Height”

Your objects will spawn randomly along the edge with a random radius variance.

How do I mass-generate an entire village on a terrain?
Pick a flat area on the terrain (though a mountainous area is totally fine).

- Set your Terrain's tag to “Terrain” or something similar
- Enable Obstacle Awareness and set the “Spawn on Tags” to your terrain tag
(This is to make sure you only spawn on terrain, and not on top of other buildings)
- Set the “Spawn Radius” to a large number – basically the size of your village.
- Set “Area Type” to “Spawn at Generator”
- Set Object Height Position to “Use Raycast Height”

How do I add a TRAP to my dungeon that drops falling rocks on the player?
First, create your rock prefab. Add a collider. Add a Rigidbody with gravity enabled.

- Add a Sphere Collider to your Population Generator object. Set it as TRIGGER. This collider is the area that will detect your Player.
- Add the rock to your Population Generator slot.
- Set “Spawn Radius” to a small number.
- Set “Area Type” to “Spawn on Generator”
- Set “Total to Spawn” to a small number like 5
- Set Object Height Position to “Use Spawner Height”
- Set “Runtime Method” to “Trigger Instant”
- Set “Trigger Tags” to the tag of your player, and anyone else you want the rocks to drop on

How do I simulate flying cars, birds, etc... in my scene?
The same way you'd set up a normal moving NPC.

- Set up the waypoint system. Place waypoints ABOVE your city, in the sky.
- On your prefabs, turn off gravity on any rigidbodies

How do I create a Meteor Shower / Randomly falling meteorites?
Place a Population Generator up high in the sky. Set a large radius. Enable Gravity on your meteors. Set Object Height Position to “Use Random Height”

How do I generate random noises when the player moves into an area?
Easy! Same way you'd spawn anything else. Create a BLANK gameobject. Add an AudioClip that plays automatically. This can be a sound of birds, forest ambience, etc... Drag the gameobject into the Assets folder to make it a prefab. Insert that prefab into the Population Generator.

{2} INSTALLATION

Installation is simple. Just import, and be done.

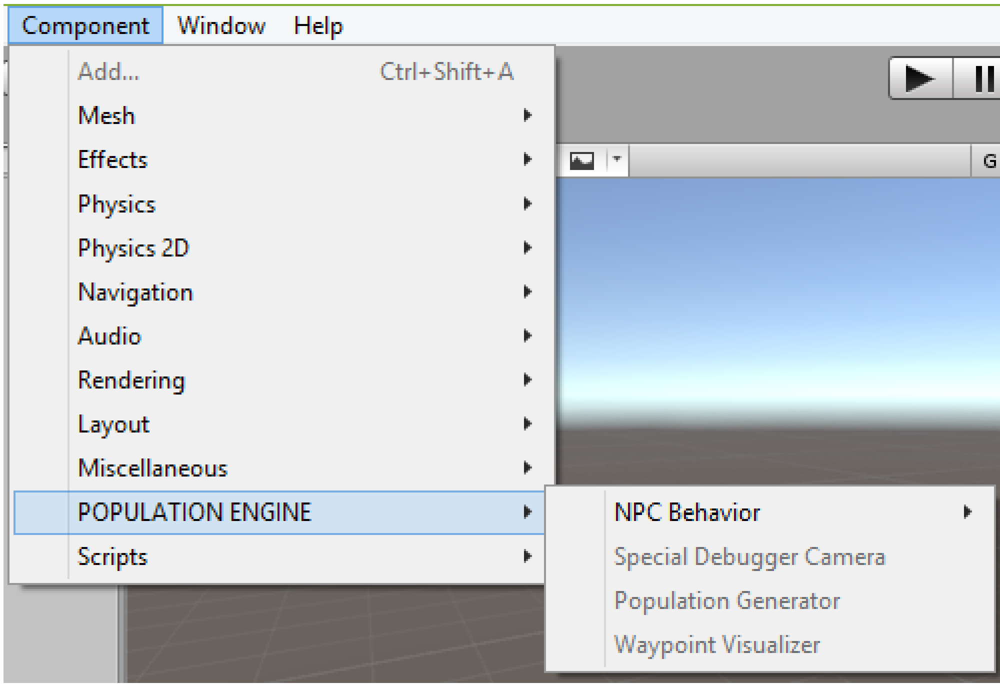
FIRST TIME INSTALLATION: Please create a new blank project. Then import our Asset. This is so you can go through the files and make sure there are no similar file names that might overwrite your existing files. After you've made sure there are no file naming conflicts, you can import this asset into your existing project.

FUTURE INSTALLATIONS: After you've checked out the demos and learned how to use it, you probably don't need all the demo resources. For future use, the only things you need to import are:

1) /Core/
2) /Scripts/
3) /Plugins/

If you're using our built-in animation controllers, you may also need:
4) /Animations/
5) /Animation Controllers/

CORRECT INSTALLATION: If everything installed correctly, a new sub menu will appear under the COMPONENTS menu, named "POPULATION ENGINE". All the tools you need are in this menu. → → →



{3} PREPARING YOUR CHARACTER MODELS, OTHER OBJECTS & ANIMATIONS

This section will show you how to prepare your characters for spawning. These steps are all highlighted in our tutorial video, “Basic Setup”

NON-CHARACTER OBJECTS

If you're spawning trees, buildings, rocks, etc… You do not need to do anything extra.
You can skip this section.

CHARACTER OBJECTS (NPCs)

If you're spawning people, AI, animals, pets, – we'll call these “NPCs” from now on – and you want to integrate movement with your existing systems, simply do the following:

For existing NPC prefabs (not setting up a raw model from scratch)

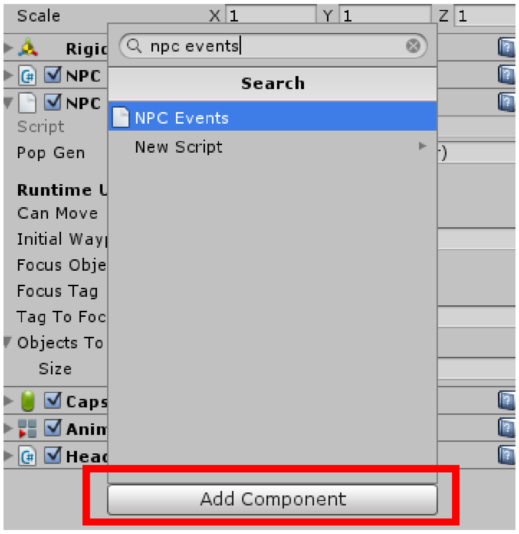
1) Select your NPC prefab.
2) ADD COMPONENT (type in “NPC Events”)
COMPONENT → POPULATION ENGINE → NPC Behavior → NPC Events

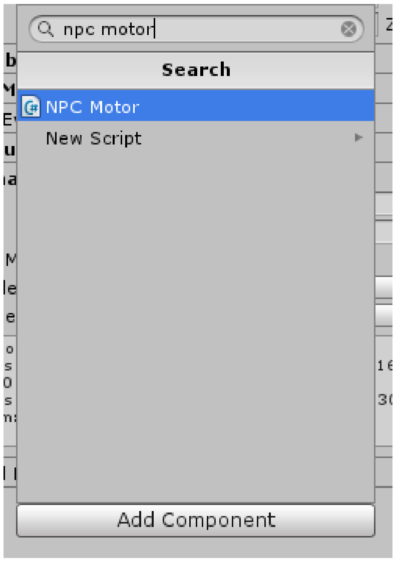
3) ADD COMPONENT and type in **“NPC Motor”.**
Go to Chapter 2 for details on NPC Motor setup.

4) Drag your character back into the Assets folder to create a prefab out of it. Later you will use this prefab as a spawn type.

You're done.

→ *For setting up new NPCs from scratch, read next page.*



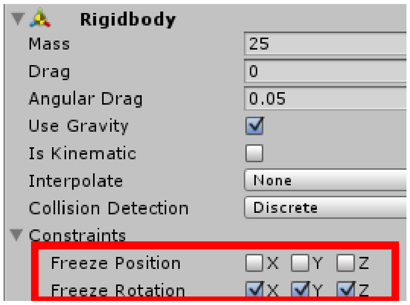


For new NPCs (setting from scratch)

1) Create a blank scene. Insert your Raw Model into the scene. For out-of-box integration with animations, make sure it's a Humanoid model with a humanoid avatar.

2) Add a COLLIDER. For humanoid models, we recommend a Capsule Collider with radius around 0.3.

3) Add a RIGIDBODY. Make sure all Rotational Constraints are checked.

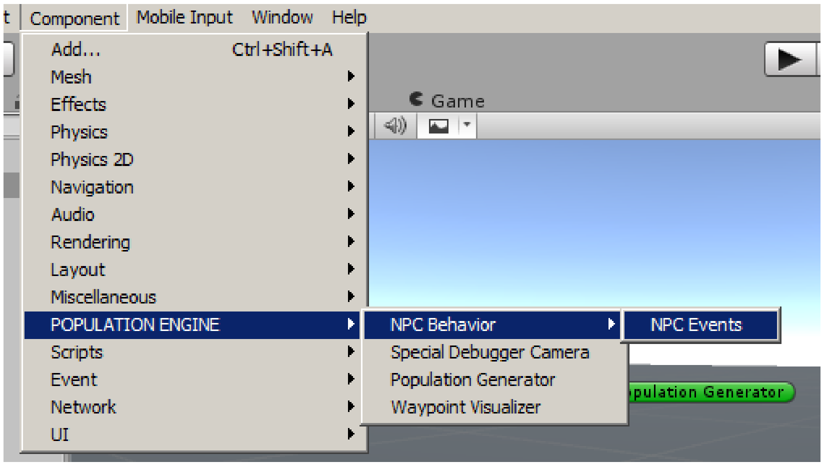


Rigidbody

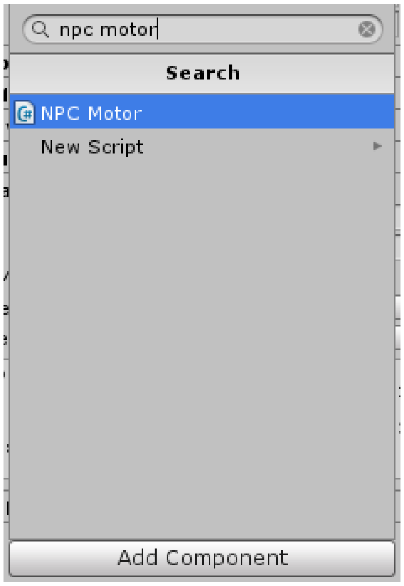
Mass	25
Drag	0
Angular Drag	0.05
Use Gravity	☑
Is Kinematic	□
Interpolate	None
Collision Detection	Discrete
Constraints	
Freeze Position	□X □Y □Z
Freeze Rotation	☑X ☑Y ☑Z

4) Add an Animator (usually Unity has already done for you). In the Animator Controller slot, add our pre-built controller named POPULATION ANIMATOR

5) In the Menu, click:
COMPONENT → POPULATION ENGINE → NPC Behavior → NPC Events



6) Now click ADD COMPONENT and type in **"NPC Motor".** Add the NPC Motor component to your prefab. See next section on NPC MOTOR SETUP. You also need the NPC Events component, but adding the NPC Motor will automatically add the NPC Events.



npc motor

7) To turn this character into a usable Prefab, you need to drag this character back into your ASSETS folder. Name it something like "My New Character".

After you've made your character into a prefab, delete the character from your scene.

You're done.

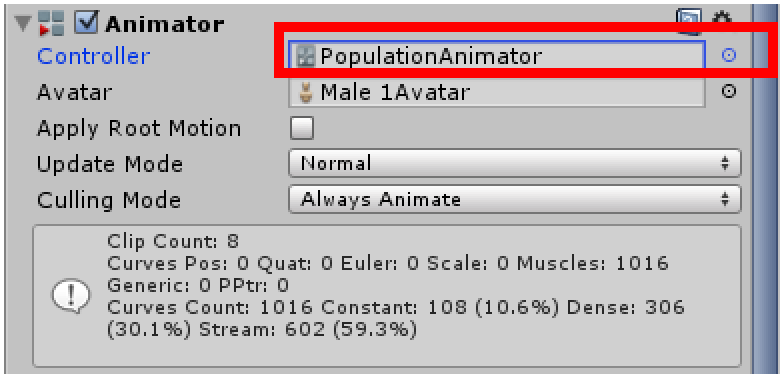
CHARACTER MODEL / ANIMATIONS SETUP CONTINUED

If you're using our included Animation Controller, the animations works like this:

There is a Parameter called “Speed” in the Animator.

In the transitions between Idle, Walk, and Run, there are logical conditions for Speed.

1) The Idle animation plays if Speed = 0
2) The Walk animation plays if Speed < 2
3) The Run animation plays if Speed > 1.99



☑ Animator

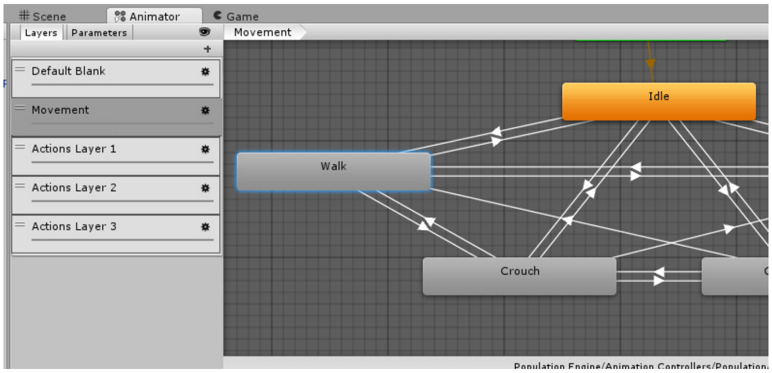
Controller: PopulationAnimator
Avatar: Male 1Avatar
Apply Root Motion: □
Update Mode: Normal
Culling Mode: Always Animate

Clip Count: 8
Curves Pos: 0 Quat: 0 Euler: 0 Scale: 0 Muscles: 1016 Generic: 0 PPtr: 0
Curves Count: 1016 Constant: 108 (10.6%) Dense: 306 (30.1%) Stream: 602 (59.3%)

Idle, Walk, Run can be found in the MOVEMENT layer.
You may notice there are other layers, Actions Layer 1, Actions Layer 2, Actions Layer 3
These layers are non-functional and are there for your reference and as a recommended way of setting up the Animator Controller. Some of them may be included as integrated in a future update (for example, we may have “Drink” as an action and integrate that with our system out of the box).

To change animations, simply swap them out:

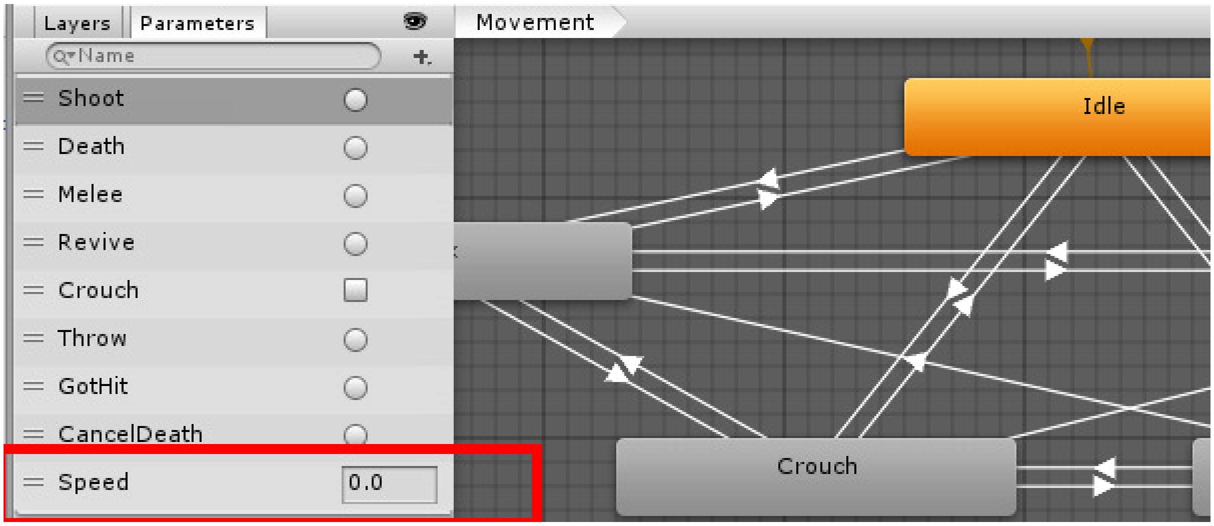
Idle, Walk, and Run animations will need to be set to LOOP and LOOP POSE.



CHARACTER MODEL / ANIMATIONS SETUP CONTINUED

SCRIPTING / SPEED UPDATING:
Our included script, NPCMotor.cs already auto-updates the Speed parameter for you and you can use it without any coding on your part. But if you decide to use custom scripting, you can either create your own Animator Controller, or fill in the Animator Float Speed in your code.

Please see the scripting section at end of manual.



Shoot	○
Death	○
Melee	○
Revive	○
Crouch	□
Throw	○
GotHit	○
CancelDeath	○
Speed	0.0

{4} NPC MOTOR AND NPC EVENTS COMPONENTS

NPC EVENTS COMPONENT
This component serves as the bridge between the Population Generator, Waypoint System, and your own code. When your NPC is spawned, the Population Generator pushes some data into the NPC Events component that will be used by your own scripts. Additionally, you will query the NPC Events component to get waypoint positions in order to move your NPCs.

Do not manually fill in any of the slots on this component. It will cause confusion to the NPC.

The NPC EVENTS component is **REQUIRED** on your spawned objects if:

1) You are using the NPC Motor to power your population's movements
2) You wish to integrate existing code with our waypoint system.

If you have no use for these, such as when you already have a big movement framework in place and want minimal intrusion by the Population Engine or if you're simply spawning static objects like rocks, trees, etc..., then you can skip this entire chapter.

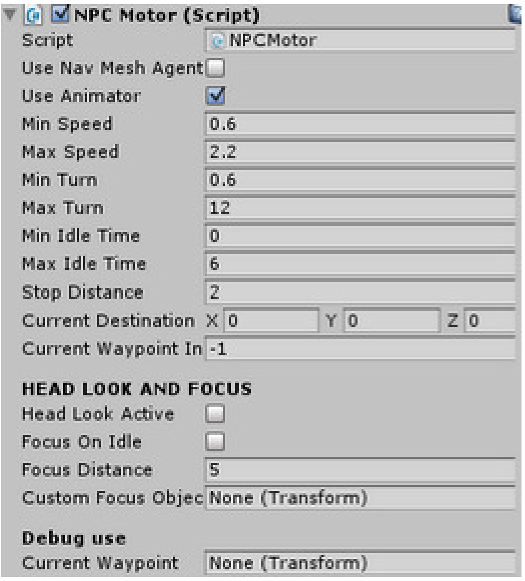
NPC MOTOR COMPONENT
The NPC Motor is a fully editable C# script that shows how to communicate with the Population Generator, retrieve objects to look at, retrieve waypoint positions, and move your character on your waypoint route. Additionally, it integrates with the Head Look Controller. All of these work together to create NPC simulations that both look good and are performance-focused.

You do not need coding experience to operate the NPC Motor. Simply attach it to your NPC and watch it run to your waypoints and look at specific objects. Only edit the code if you need more advanced integration.

This component is NOT required to use Population Engine. If you're spawning static objects such as rocks and buildings, this component should never be attached.

→ *See next page for details about each NPC Motor setting.*

THE NPC MOTOR COMPONENT:



☑ NPC Motor (Script)	
Script	NPCMotor
Use Nav Mesh Agent	□
Use Animator	☑
Min Speed	0.6
Max Speed	2.2
Min Turn	0.6
Max Turn	12
Min Idle Time	0
Max Idle Time	6
Stop Distance	2
Current Destination	X 0 Y 0 Z 0
Current Waypoint In	-1
HEAD LOOK AND FOCUS	
Head Look Active	□
Focus On Idle	□
Focus Distance	5
Custom Focus Objec	None (Transform)
Debug use	
Current Waypoint	None (Transform)

Use Nav Mesh Agent: This has no function as of now. It's a new feature for a potential future update. You can certainly add your own NavMesh code if you like. However, we do not recommend Pathfinding for very large masses of NPCs.
Use Animator: Check to use Animator. Uncheck to ignore animations. If Checked, it will auto-update the "Speed" variable of the Animation Controller. The Idle-Walk-Run animations depend on the Speed variable.
Min / Max Speed: The NPC will spawn with a random speed between min/max. This creates variation in your population. To set the same speed for all, simply set both min/max to same number.
Min / Max Turn: Random rotational speed range. If your NPC runs in circles around a waypoint, set this to a higher number, or set speed to lower number.
Min / Max Idle Time: Each time the NPC arrives at a waypoint, it will stop and go Idle for some time. It randomly picks a time from this range, at every waypoint. If Head Look or Object Focus is enabled, the NPC will stop and turn towards the object.
Stop Distance: The distance the NPC will reach before stopping at a waypoint. If your areas are very crowded, it's good to set this to a higher number. If your scene isn't crowded, set this to a lower number.
Current Destination: This simply tells you, at runtime, where the NPC is currently headed. You do not fill this out. It's good for debugging your NPCs.
Current Waypoint Index: This tells you, at runtime, the current index of the waypoint chain that it's moving towards. It uses arrays, so an index of 0 represents Waypoint #1...index of 1 represents Waypoint #2, etc… This is a debugger tool for coders.
Head Look Active: Check this if you are using Head Look integration. Your NPC must have a HEAD LOOK controller attached, or this will automatically be disabled. When moving, the NPC will always turn its head towards the nearest object of interest. You define the distance to turn and which object to focus.
Focus On Idle: Check this if you want the NPC to stop and turn towards a specific object (such as a car in a showroom). See Showroom demo scenes.
Focus Distance: The distance to object of interest to activate body focus or head look controllers.
Custom Focus Object: the object to focus at. If you're spawning this NPC at runtime, the objects to focus are set in the Population Generator component itself, under "Auto Focus Settings"

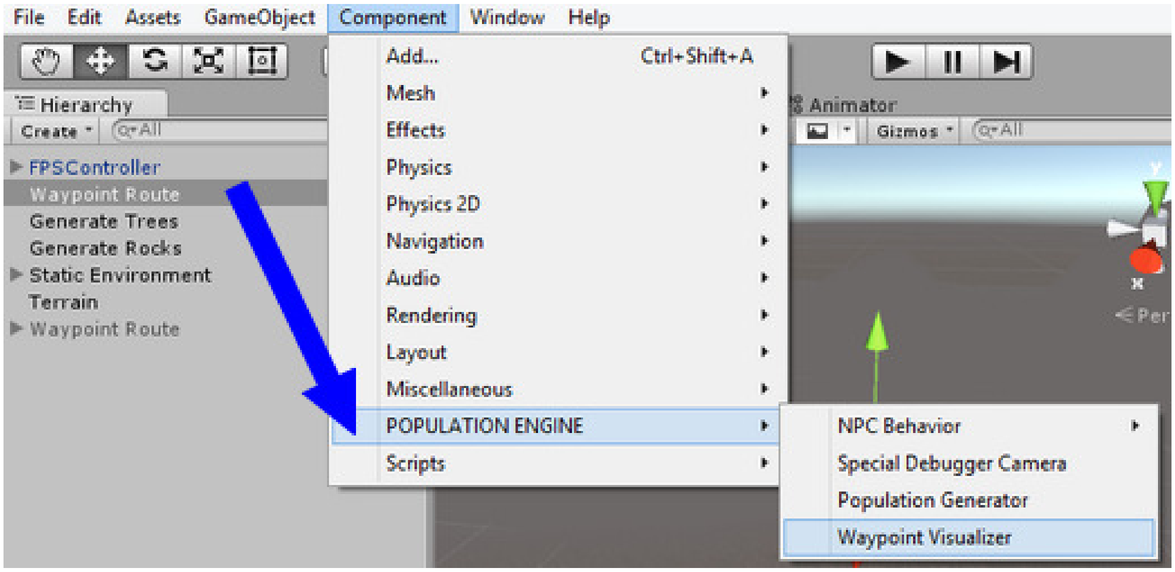
{5} SETTING UP WAYPOINTS, ROUTES, DIRECTIONAL TRAFFIC

The waypoint system is very simple to use. The entire workflow involves drag and drop. Basically, you create a waypoint system, place your waypoints, then load the waypoint system into the Population Generator.

CREATING A NEW WAYPOINT SYSTEM:

Make sure your scene has a "ground" object such as a plane or terrain. Make sure this ground has a collider. Your NPCs need something to stand on.

1) Create an empty GameObject. Name it something like "Waypoint Route"
2) Select this object. Click: **COMPONENT → POPULATION ENGINE → Waypoint Visualizer**



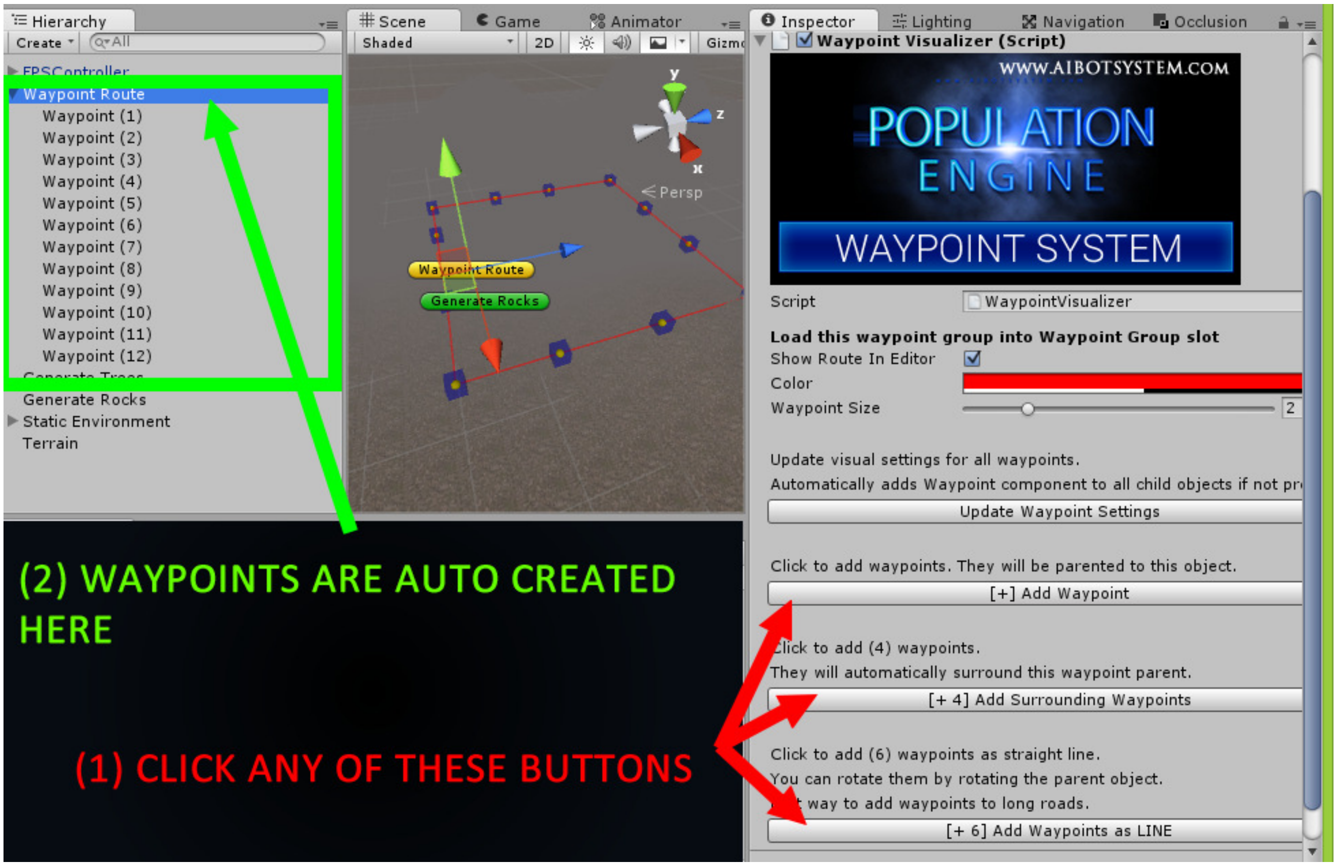
Now you're ready to add waypoints.

CREATING WAYPOINTS

1) In the Inspector, click [ADD NEW WAYPOINT]. Notice a new object is created as a child. This is a waypoint.
2) Click [ADD NEW WAYPOINT] again. This adds another waypoint.
3) Click the 2nd waypoint in the hierarchy. Now, move it to another position. You'll see a line is automatically drawn. This is your route. If you were to use this system now, your NPC will go in a straight line, back and forth between the 2 waypoints.

Feel free to add as many waypoints as you want. The order of the waypoints in the hierarchy tells the NPC the route to follow.

→ *See next page for details*



☑ Waypoint Visualizer (Script)

Script: WaypointVisualizer

Load this waypoint group into Waypoint Group slot

Show Route In Editor ☑

Color 🟥

Waypoint Size 2

Update visual settings for all waypoints.
Automatically adds Waypoint component to all child objects if not pr

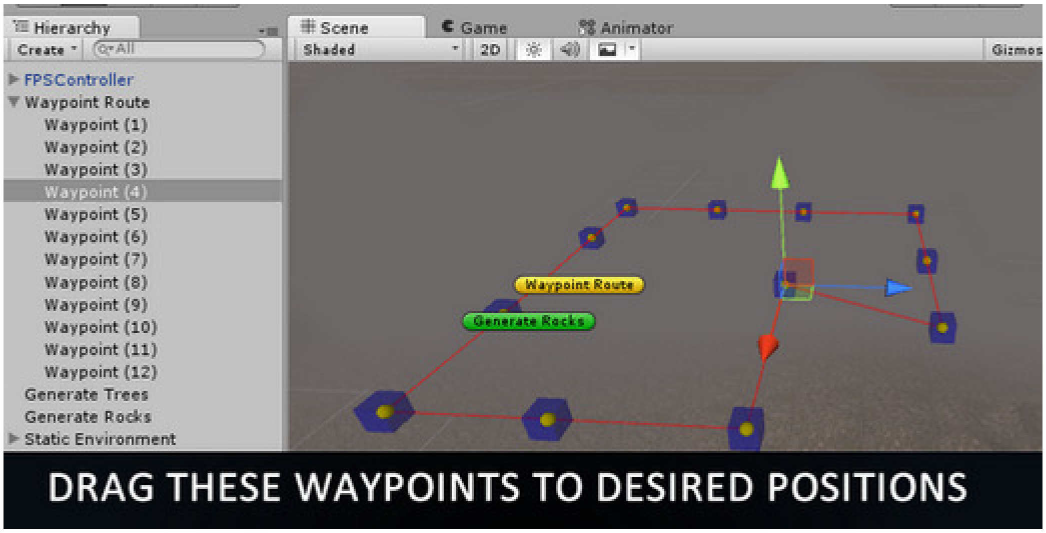
Click to add waypoints. They will be parented to this object.

Click to add (4) waypoints.
They will automatically surround this waypoint parent.

Click to add (6) waypoints as straight line.
You can rotate them by rotating the parent object.
t way to add waypoints to long roads.

(2) WAYPOINTS ARE AUTO CREATED HERE

(1) CLICK ANY OF THESE BUTTONS



DRAG THESE WAYPOINTS TO DESIRED POSITIONS

Colors: You can color code your waypoint systems. To change the color, click on the main waypoint parent, and set the color to Red. Then click [Update Waypoint Settings]. You can do this with waypoint sizes, etc… Just click [Update] afterwards.

To Change Colors of All Waypoints: See above.

To Change Color of individual waypoint: click on the waypoint itself. Change the color in the inspector Color setting.

Auto Waypoint Formations: Under the [ADD NEW WAYPOINT] button, there are 2 more buttons. These buttons will automatically generate many waypoints to save you time.

The "Add Surrounding Waypoints" button will generate 4 waypoints that surround the position of your waypoint system. Example: You can place an empty waypoint system at the center of a house. Then click "Surround" and 4 waypoints will be generated around the house.

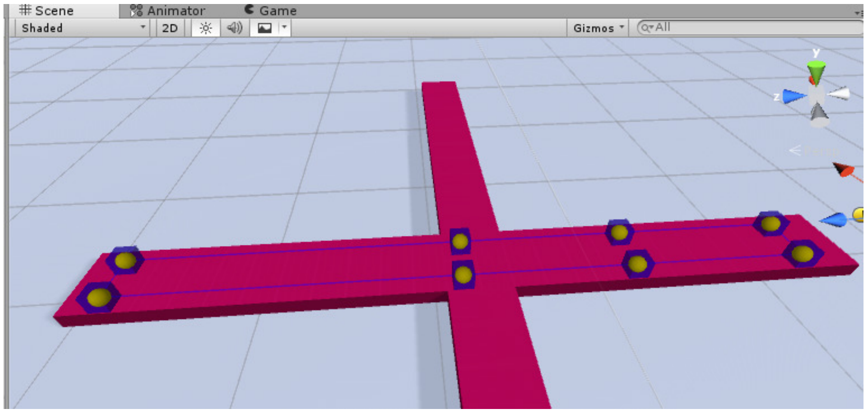
The "Add Waypoints as Line" button will generate 6 waypoints in a straight line. This is good for long roads.

DIRECTIONAL TRAFFIC: You can easily simulate multi-directional traffic. When you create your waypoints, drag the waypoints around so that they form a circular route so when your NPCs follow it, then will automatically go the reverse direction.

<u>In the same Waypoint Route:</u>
Create waypoints in one direction:
[x] → [x] → [x] → [x] → [x] → [x] →
Then create waypoints in opposite direction, and place them alongside the first group:
[x] ← [x] ← [x] ← [x] ← [x] ← [x] ←

Example: There are 2 streets here. Look at the center street. A single waypoint system simulates a 2-way street.

When played, this waypoint setup will have NPCs moving in opposite directions.



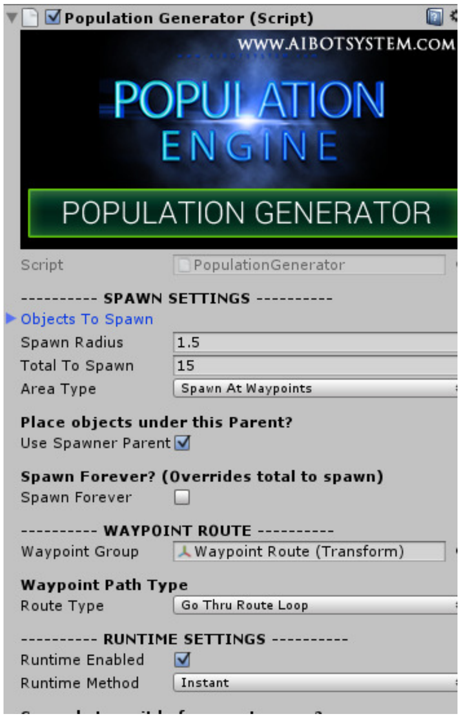
{6} GENERATING POPULATION: WORKFLOW FOR RUNTIME AND MANUAL

With your characters setup and waypoint route created, you're ready to start generating population.

Note: The Population Engine component is fairly big and a full screenshot cannot fit this page. So we will include snapshots of parts of the component as we discuss them.

At this point, you don't need anything in your scene except your environment and waypoint route created in previous step. Characters are not needed in the scene since we will be generating them automatically.

STEP 1) Create an empty GameObject. Name it something like "Population Generator 1"
STEP 2) Select this object. Click: **COMPONENT → POPULATION ENGINE → Population Generator**



Script: PopulationGenerator

---------- SPAWN SETTINGS ----------

▶ Objects To Spawn

Spawn Radius: 1.5

Total To Spawn: 15

Area Type: Spawn At Waypoints

Place objects under this Parent?

Use Spawner Parent ☑

Spawn Forever? (Overrides total to spawn)

Spawn Forever □

---------- WAYPOINT ROUTE ----------

Waypoint Group: Waypoint Route (Transform)

Waypoint Path Type

Route Type: Go Thru Route Loop

---------- RUNTIME SETTINGS ----------

Runtime Enabled ☑

Runtime Method: Instant

All the magic happens in this single component. We put all the tools you need into one component for your convenience. Think of this component as a "swiss knife" that can be used for multiple purposes. This component powers both the runtime generation and manual tools which you can use to "paint" groups of NPCs into your scene.

You can have multiple Population Generators in a scene. For example, in a large RPG map, you might create a new Population Generator for every town on the map. Some will generate elves, some will generate humans, etc… Your dungeon might have a different Generator on each level. Later, we show you how to use TRIGGERS, which you can to activate a Generator when a player steps into its area. So when your player steps into a new dungeon area, it can trigger traps or spawn new enemies.

STEP 3) Load the Waypoint System into the Population Generator. Simply drag your Waypoint Route from your scene into the "Waypoint Group" slot.

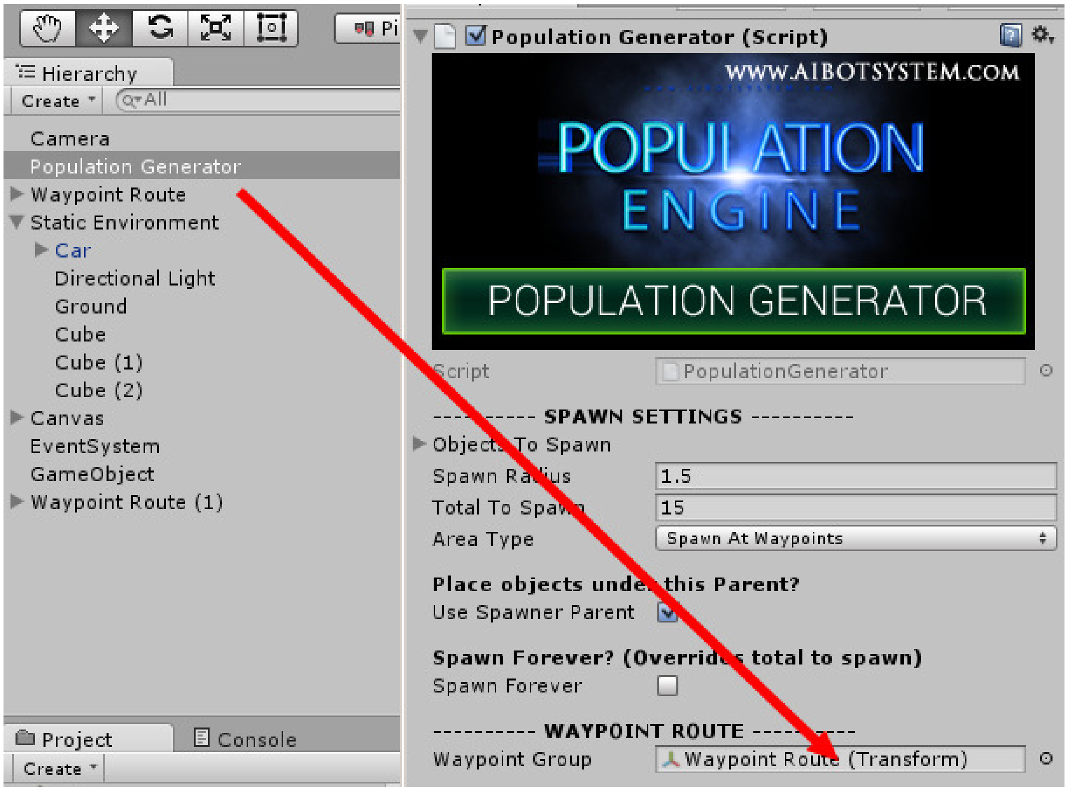
Now select the Waypoint Path Type:

Go Thru Route Loop: The NPCs will move through the waypoints, then go back to the first.
A → B → C → D → A → B → C → D
If you want the NPC to reverse directions, simply place additional waypoints in the opposite direction.

Go Thru Route Once: The NPCs will move through the waypoints and stop at the last.
A → B → C → D

Go Random Waypoint: The NPCs will randomly pick waypoints and go out of order. Note: If you have a lot of obstacles in your scene, make sure the waypoints are all in sight of each other.

Shown: STEP 3. If you're creating NPCs that move around, you should immediately drag your Waypoint Route object into the “Waypoint Group” slot of the Population Engine. *It's easy to forget this step and can result in becoming confused about where errors are coming from later.*



☑ Population Generator (Script)

Script: PopulationGenerator

---------- SPAWN SETTINGS ----------

Objects To Spawn

Spawn Radius: 1.5

Total To Spawn: 15

Area Type: Spawn At Waypoints

Place objects under this Parent?

Use Spawner Parent ☑

Spawn Forever? (Overrides total to spawn)

Spawn Forever □

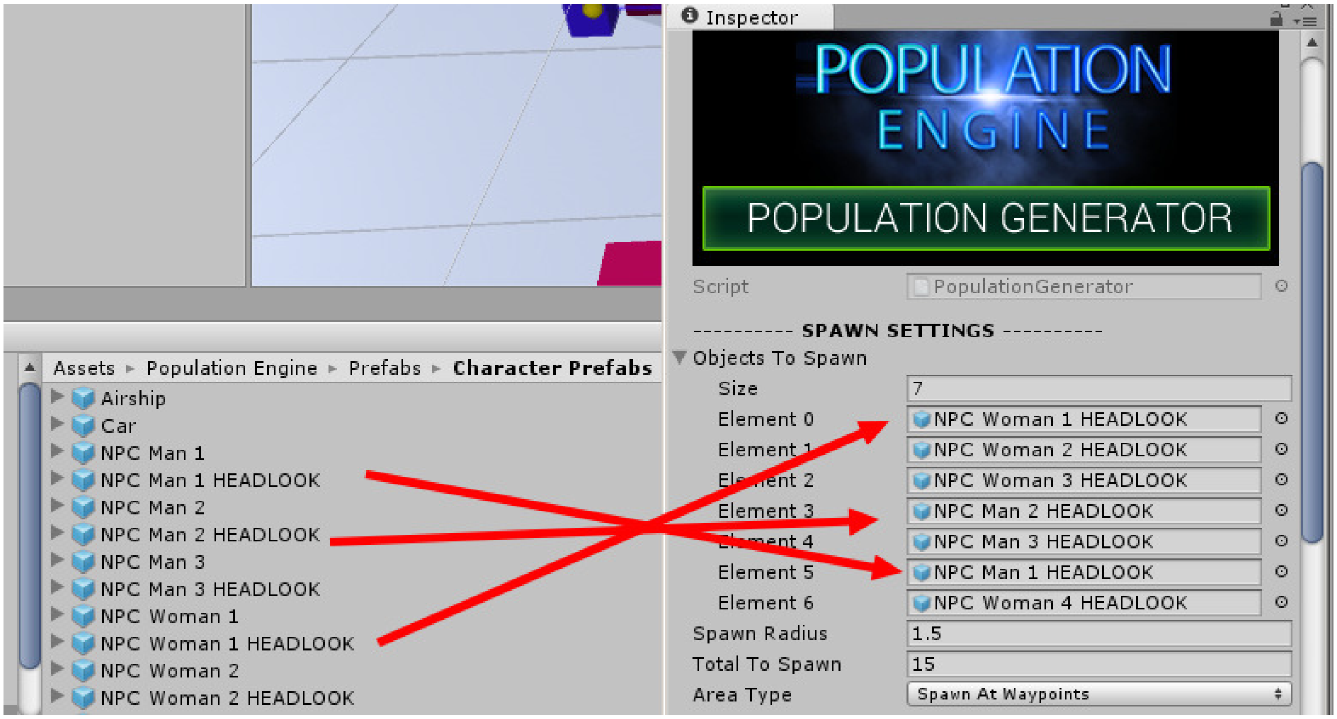
---------- WAYPOINT ROUTE ----------

Waypoint Group: Waypoint Route (Transform)

→ *Go on to next page.*

STEP 4) INSERT OBJECTS. Now, insert your objects to spawn into the “Objects to Spawn” slot. These are the character prefabs you created in the first section “Preparing Your Character...”, but it can be anything you want – including trees, buildings, rocks, props, particle effects, enemy AI, etc...

Shown: Step 4. Insert as many prefabs as you want. **The system will randomly pick objects from this pool to spawn.** They all have the same chances of being spawned. If you only want to spawn 1 object type, you can just insert 1 object. Simply drag and drop.



Script: PopulationGenerator

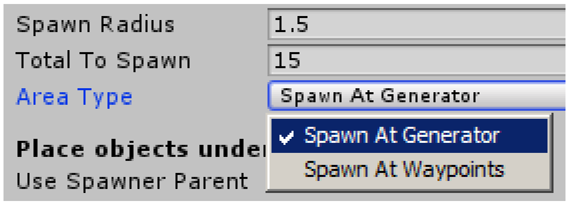
---------- SPAWN SETTINGS ----------

Objects To Spawn
- Size: 7
- Element 0: NPC Woman 1 HEADLOOK
- Element 1: NPC Woman 2 HEADLOOK
- Element 2: NPC Woman 3 HEADLOOK
- Element 3: NPC Man 2 HEADLOOK
- Element 4: NPC Man 3 HEADLOOK
- Element 5: NPC Man 1 HEADLOOK
- Element 6: NPC Woman 4 HEADLOOK

Spawn Radius: 1.5
Total To Spawn: 15
Area Type: Spawn At Waypoints

STEP 5) Set Spawn Radius and Total to Spawn. The Spawn Radius is the area around the spawner from which to generate objects. The location of spawning is determined at random.
If you're spawning at the Population Generator location itself, it's the radius from center of its position. If you're spawning at waypoints, it's the distance from center of each waypoint.

For this example, set the Area Type to “Spawn at Generator”
This will turn on the Editor visuals. You'll see a blue wire sphere around the object that represents the spawn radius.
You can click and drag your mouse over the “Spawn Radius” label text in the Inspector and see the sphere change size.
The sphere will not be drawn if you select “Spawn at Waypoints”



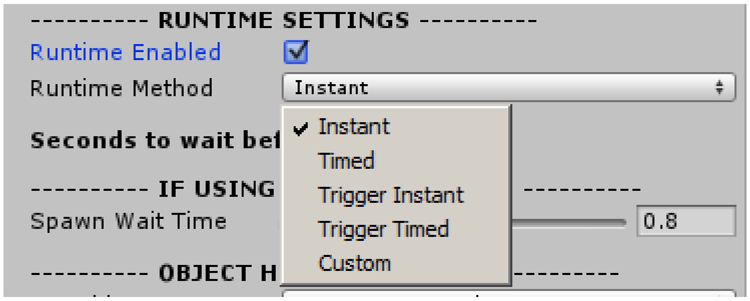
Spawn Radius: 1.5
Total To Spawn: 15
Area Type: Spawn At Generator
Place objects unde
Use Spawner Parent

Set Total to Spawn is the total amount of objects to spawn. It is not the total of *each* object, but the total of ALL objects combined. In later sections, we'll show you how to set enemy waves (example: You can want a maximum of 5 objects in scene simultaneously, but generate a total of 100. When 1 object is destroyed, auto spawn another).
Check “Spawn Forever” if you want to spawn infinite objects.

STEP 6) Set Runtime or Manual. Do you want the Generator to activate automatically at runtime, or will you manually spawn them in the editor?

UnCheck "Runtime Enabled" if you're manually using this tool to place population into your scene. Unless, you want to both create objects manually and let the Generator work at run time.

Check "Runtime Enabled" to let the Generator randomly spawn objects at run time.



---------- RUNTIME SETTINGS ----------

Runtime Enabled ☑

Runtime Method Instant

Seconds to wait bef

---------- IF USING

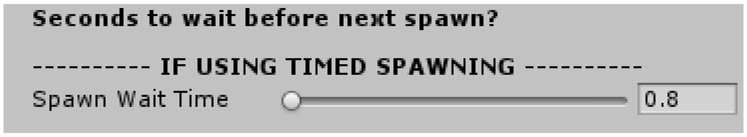
Spawn Wait Time 0.8

---------- OBJECT H

If Checked, Pick a Runtime Method:

Instant: Objects will be spawned instantly, as soon as the Generator loads (unless this GameObject is deactivated). For mass amounts of objects, your scene might have a slight delay when it loads.

Timed: Each object will be spawned according to an interval. This will not be instant.
Set this to # of seconds you want to wait before spawning next object.
This is useful if you want NPCs to "gradually" appear in your town or prevent enemy units from overwhelming the player.

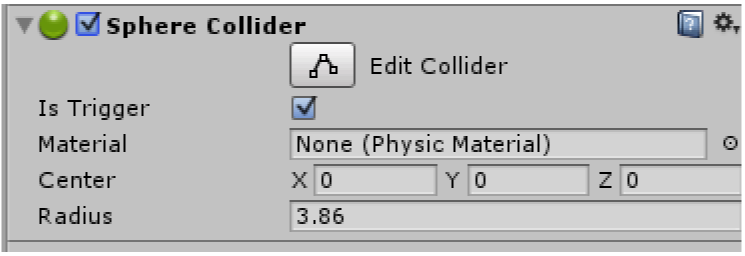


Seconds to wait before next spawn?

---------- IF USING TIMED SPAWNING ----------

Spawn Wait Time 0.8

Trigger Instant: Same as "Instant" except the Generator will wait until the player walks into its collider trigger. You'll need to add a collider and set to Trigger. Works with any collider type that is trigger. This is useful if you want to spawn NPCs or enemies only when a player walks into that area.



☑ Sphere Collider

Edit Collider

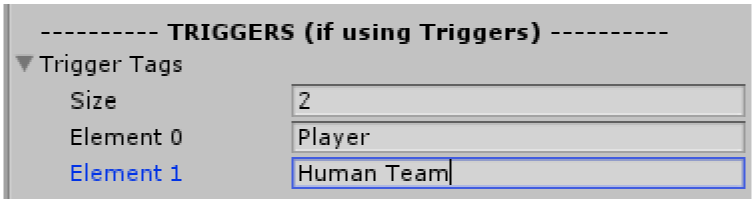
Is Trigger ☑

Material None (Physic Material)

Center X 0 Y 0 Z 0

Radius 3.86

This also requires you to set up Trigger Tags →
Set unlimited tags to detect to activate the trigger.



---------- TRIGGERS (if using Triggers) ----------

Trigger Tags

Size 2

Element 0 Player

Element 1 Human Team

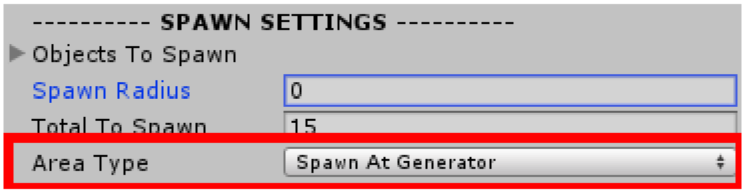
Trigger Timed: Similar to "Trigger Instant" except the Generator will use a timer instead of spawning everything instantly.

Custom: This setting is for your custom scripts. When set, the Generator will not do anything until your script tells it to, at runtime. You may still use it for manual generation in the editor. See the section on Scripting.

STEP 7-A) Set Spawn Position. Where you do you want your objects spawned?

a. Set Spawn Radius above 0.

b. Set Area Type: Do you want to spawn at the Generator's position or at randomly selected Waypoint positions?



---------- SPAWN SETTINGS ----------

▶ Objects To Spawn

Spawn Radius 0

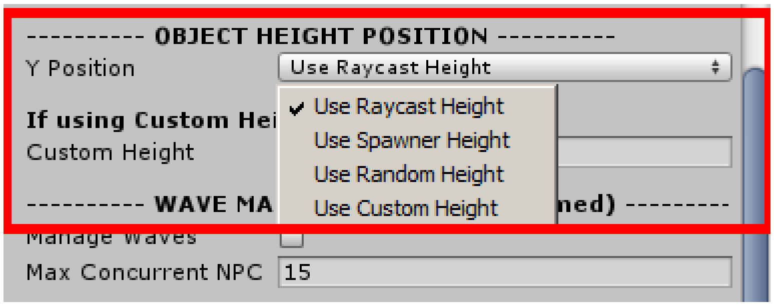
Total To Spawn 15

Area Type Spawn At Generator

c. Set Y Position: How to determine the Y position of each spawned object.

Use Raycast Height

Picks the closest collider from top down and spawns on there. Use this to spawn more accurately on the ground or surface.



---------- OBJECT HEIGHT POSITION ----------

Y Position Use Raycast Height

✓ Use Raycast Height
Use Spawner Height
Use Random Height
Use Custom Height

If using Custom Hei

Custom Height

---------- WAVE MA ned) ---------

Manage Waves □

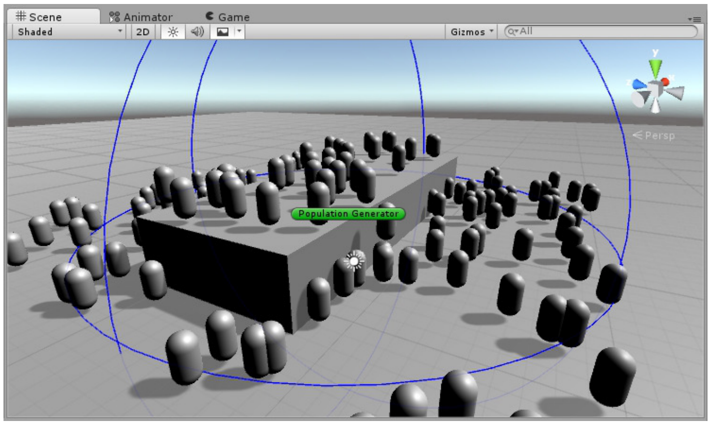
Max Concurrent NPC 15

For example, say you have a FLOOR and a TABLE. The table is above the floor. So if you place the generator above the table, the object will spawn ON the table, and not the floor. If you remove the table, the object will spawn on the floor.

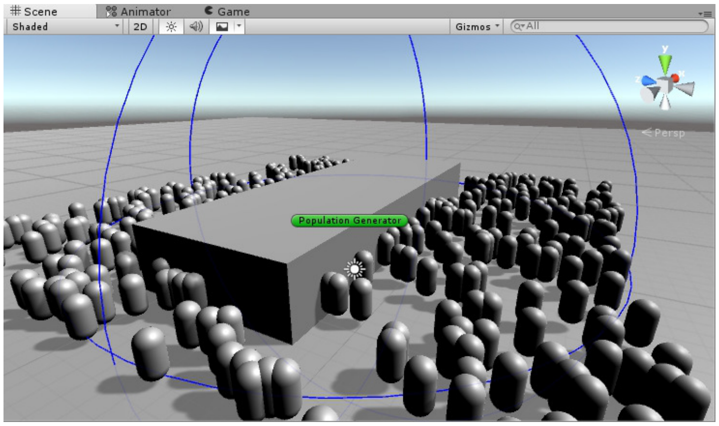
Use Spawner Height

All objects will spawn at the Y position of the spawner itself, depending on the Area Type setting. If you're spawning at waypoint positions, then it will use the Y position of each waypoint.

Using Raycast Height setting
The objects spawn on the highest collider it lands



Using Spawner Height setting
The objects all spawn at same Y position



Use Random Height: Pick a random Y position within “Spawn Radius”

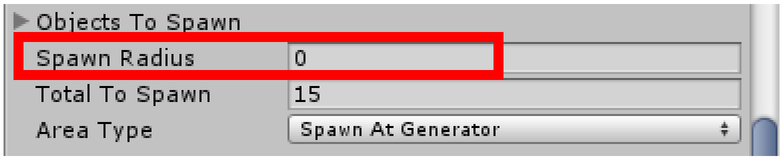
Use Custom Height: Enter a custom Y position. Also fill in “Custom Height”. These are World Positions, not local.

STEP 7-B) Do you want Random or Constant spawn positions?

For random positioning: Follow the above steps for Random Spawning.

Non-Random: If you want your objects to be spawned exactly at your spawner position (Generator or Waypoint) with no randomness, then simply set:

Spawn Radius: 0



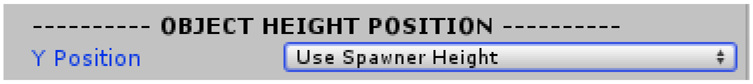
▶ Objects To Spawn

Spawn Radius: 0

Total To Spawn: 15

Area Type: Spawn At Generator

Y Position: Spawner Height



---------- OBJECT HEIGHT POSITION ----------

Y Position: Use Spawner Height

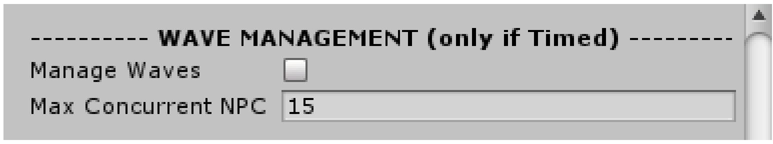
Now you're done.

There are many more settings but these are the basic requirements. The other settings are all optional.

→ *See next page for explanations on the other settings.*

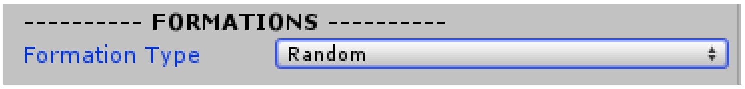
POPULATION GENERATOR OPTIONAL SETTINGS

These are all optional settings, not required for basic functionality.
This is only a summary and we will go into some of these settings in detail later.



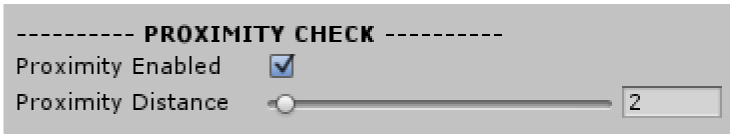
---------- WAVE MANAGEMENT (only if Timed) ---------
Manage Waves □
Max Concurrent NPC 15

Manages waves of enemies or objects. It automatically keeps track of all spawned objects for you. When they are destroyed, it will spawn more. Limits the max number of objects that can exist simultaneously.



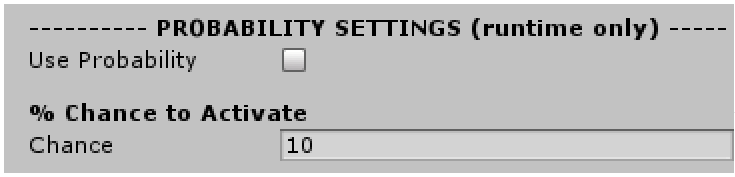
---------- FORMATIONS ----------
Formation Type Random

Spawn randomly or spawn in a line?



---------- PROXIMITY CHECK ----------
Proximity Enabled ☑
Proximity Distance 2

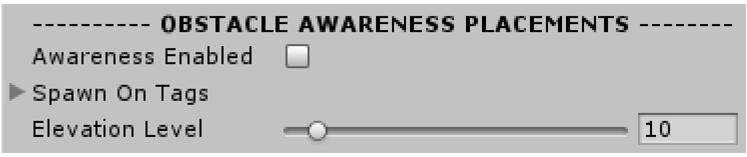
Checks if the spawn area is already taken by another object. If so, it will not spawn at all. Use this to prevent NPCs from spawning "into" each other. The bigger your objects' colliders, the higher you should set the Proximity Distance. This can result in having lower number of spawns than you specified. For example, if you set 25 spawns but only see 10 generated, it probably means some were not spawned because they would have spawned on top of each other. If this happens often, you should increase your spawn radius, or lower the # of spawns in small areas.



---------- PROBABILITY SETTINGS (runtime only) -----
Use Probability □

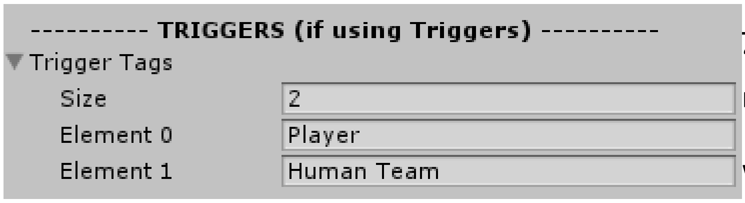
% Chance to Activate
Chance 10

The chance of spawning something at all. For example, if you set 10%, then that means every time an object is *about* to be spawned, there is a 10% chance it will actually spawn. So there is a 90% chance it will not spawn at all. Example use: In RPGs you spawn random enemies in specific areas – in some areas enemies spawn 50% of the time.



---------- OBSTACLE AWARENESS PLACEMENTS --------
Awareness Enabled □
▶ Spawn On Tags
Elevation Level 10

Objects will **only** spawn on top of another collider that has the specified tags ("Spawn On Tags"). **WORKS BEST WITH "Y-POSITION: USE RAYCAST HEIGHT" SETTING!** A good example is when you have a city scene.   Your spawn radius is big, and covers the entire city. But you don't want people accidentally spawned on top of fences, fire hydrants, etc... You ONLY want people spawned on the city streets. So you tag your street with something like "My Street" and insert it into "Spawn On Tags". Now all objects will skip everything else and only be placed onto the streets with the specified tags. Objects must have a collider that is not trigger.



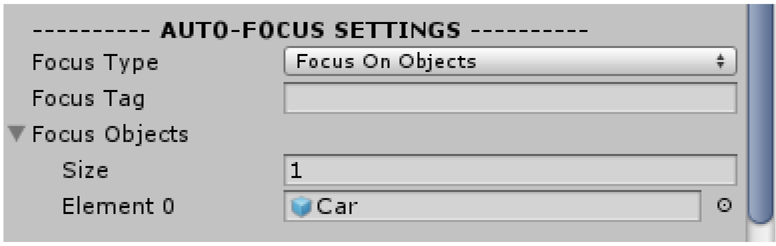
---------- TRIGGERS (if using Triggers) ----------

▼ Trigger Tags

Size	2
Element 0	Player
Element 1	Human Team

These come into play if you selected "Trigger Instant" or "Trigger Timed" as the runtime methods.

When an object with the specified tags move into the collider area of either the spawner or waypoint, (depending on which you picked), it will begin generating objects. If no object with these tags move into it, then it will not spawn anything.

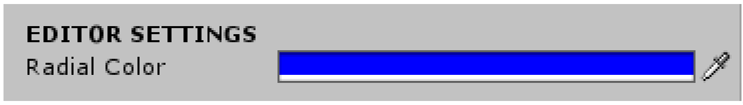


---------- AUTO-FOCUS SETTINGS ----------

Focus Type	Focus On Objects
Focus Tag	
▼ Focus Objects	
Size	1
Element 0	Car

Have your NPCs focus on specific objects either when they stop on Idle, or with their Head Look Controllers.

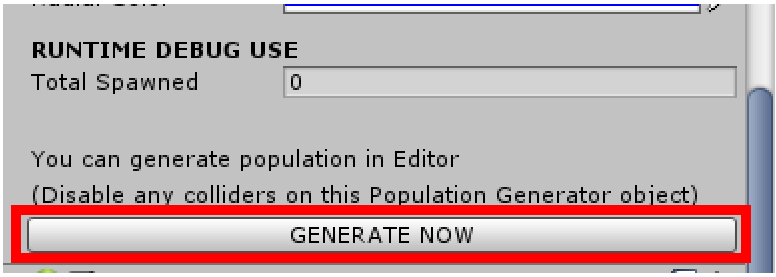
Set Focus Tag to focus on nearest tagged object. So if you have a tag "Team 1", your NPCs will always attempt to find the nearest player tagged "Team 1" and look at it. You can also set unlimited specific objects to focus.



EDITOR SETTINGS

Radial Color 🟦

Set the color of the wired sphere in your editor. This represents the spawn radius. Set different colors if you have multiple Population Generators in a scene. Note: This will not be shown if your Area Type is set to "Spawn at Waypoints"



RUNTIME DEBUG USE

Total Spawned 0

You can generate population in Editor

(Disable any colliders on this Population Generator object)

"GENERATE NOW" button.
This is for manual use, right in the editor.

You can generate objects in the editor and fine tune them yourself. It uses the same settings as the Runtime generation.

Click this button to generate. If you do not need manual generation, do not touch this button.

Becareful of accidentally pressing this when you don't want to generate. You might have set Spawn count to a large number like 500 and this will immediately generate 500 objects in your scene! Undo will not work and will not delete the objects. Very large numbers of unoptimized prefabs can crash your editor.

→ *You can find more details on Manual generation later in the manual.*

{7} OBSTACLE AWARENESS TOOLS / Intelligent Object Placement

- This applies to both Runtime and Manual generation -

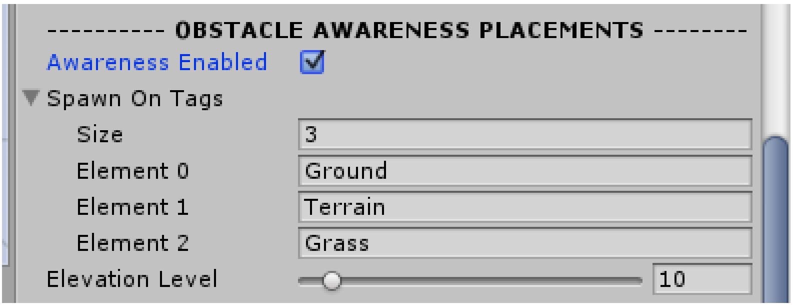
Obstacle Awareness Tools allows you to *quickly* fine tune where you want the objects spawned.

Let's say you have 2 Population Generator components in your medieval village. One randomly generates PEOPLE and the other spawns BIRDS. Both have the same spawn radius that covers your entire village. You want 300 PEOPLE always spawned on the ground, but never on top of buildings/rooftops. You want 50 BIRDS always spawned on rooftops, but never on the ground. Can you imagine having to do all this by hand? This tool helps you do that quickly.

<u>SETUP:</u>

1) Insert the tags of the objects you want to spawn **on top of.**

2) Set the Elevation Level. This tells the Generator how high to detect platforms to spawn on. 10 is fine for most cases and don't need to set this at all. If you want to spawn on a very tall SkyScraper, for example, you might want a high number.



---------- OBSTACLE AWARENESS PLACEMENTS --------

Awareness Enabled ☑

▼ Spawn On Tags

Size 3

Element 0 Ground

Element 1 Terrain

Element 2 Grass

Elevation Level 10

3) – Optional – **Y-Position setting to "Use Raycast Height" - this works best.**

4) Check "Awareness Enabled" to enable these settings. That's it.

So back to our example. Lets say you want PEOPLE to always spawn on the ground plane, terrain, and grass areas. So you tag your ground as "Ground", your terrain as "Terrain" and grass area as "Grass". Then put all 3 tags into the panel. Now, your people will only spawn on these 3 tags.

And you want BIRDS to always spawn on rooftops. So you create a new Population Generator for the birds. Tag all rooftops of your houses as "Rooftop" and insert that tag into the new panel. Now your birds will only spawn on objects tagged "Rooftop"

And if you want boats only spawned on the water, simply give your Water plane a mesh collider and tag it something like "Water" and insert that in.

{8} SETTING UP TRIGGERS

Triggers come into play if you selected “Trigger Instant” or “Trigger Timed” as the runtime methods.

When an object with the specified tags move into the collider area of either the spawner or waypoint, (depending on which you picked), it will begin generating objects. If no object with these tags move into it, then it will not spawn anything.

Example uses:

1) When the player walks into a specific area, start generating random enemies at that location.

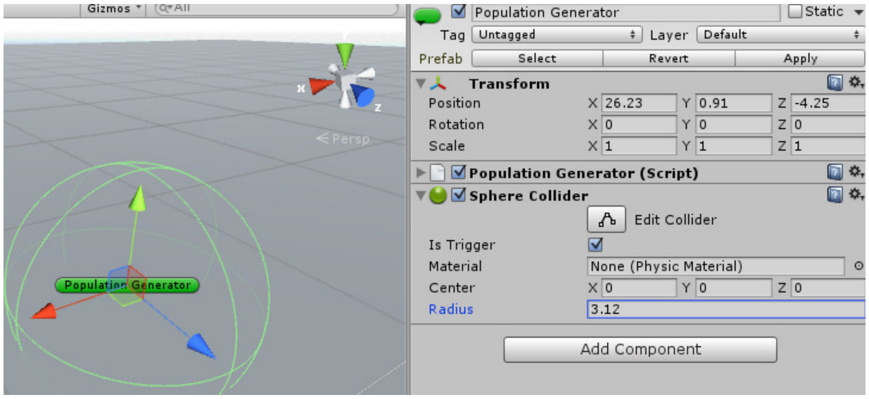
2) When the player walks into a specific area, start generating random enemies at *another* location.

3) When the player walks into a specific area, generate random traps and falling rocks on top of the player.

4) When a monster moves into an area, generate random guards to attack it.

SETUP:

1) Add a Collider to your Population Generator. Mark it as a Trigger. The size of this collider is the area of effect. Any collider works – box, sphere, capsule, mesh, etc...



☑ Population Generator ☐ Static

Tag Untagged Layer Default

Prefab Select Revert Apply

Transform

	X	Y	Z
Position	26.23	0.91	-4.25
Rotation	0	0	0
Scale	1	1	1

☑ Population Generator (Script)

☑ Sphere Collider

Edit Collider

Is Trigger ☑

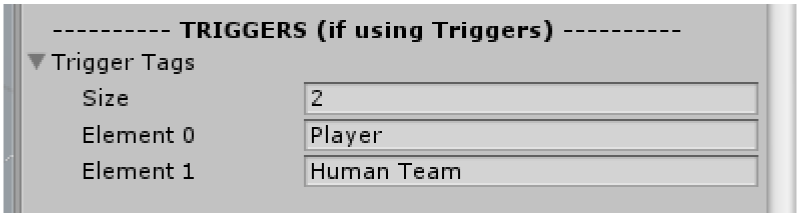
Material None (Physic Material)

Center X 0 Y 0 Z 0

Radius 3.12

Add Component

3) Make sure your target object (such as Player) has both a collider and rigidbody.

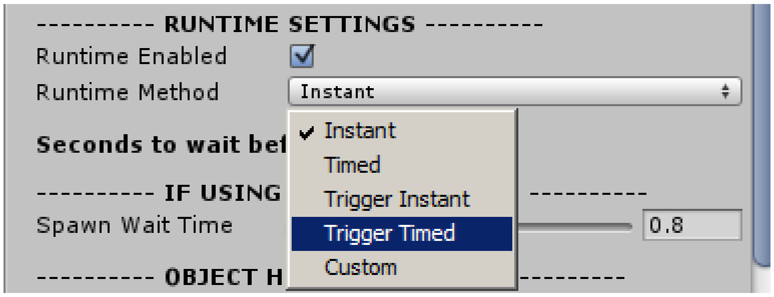


---------- TRIGGERS (if using Triggers) ----------

Trigger Tags

Size	2
Element 0	Player
Element 1	Human Team

3) Set the tags of the objects that will trigger this effect.



---------- RUNTIME SETTINGS ----------

Runtime Enabled ☑

Runtime Method Instant

Seconds to wait bet

---------- IF USING ----------

Spawn Wait Time 0.8

---------- OBJECT H ----------

4) Set Runtime Method to either “Trigger Instant” or “Trigger Timed”

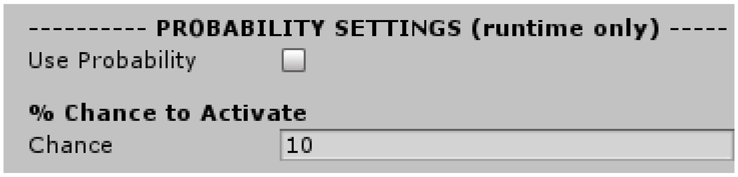
{9} PROBABILITY-DRIVEN SPAWNING

Let's say you have an RPG and you want random enemies spawned every 3 seconds in a forest, but only a 50% chance of spawning. So this means every 3 seconds, there is a 50% chance an enemy will spawn. This will help you do that.

<u>SETUP:</u>

1) Set % Chance to Activate:
This is the chance of spawning something at all.

2) Check "Use Probability"



---------- PROBABILITY SETTINGS (runtime only) -----
Use Probability ☐

% Chance to Activate
Chance 10

You're done.

<u>This affects:</u>

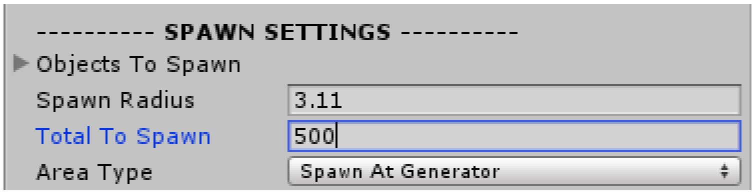
1) Runtime Instant spawning
2) Runtime Timed spawning

{10} CREATING ENEMY WAVES / CONCURRENT OBJECTS MANAGEMENT

The Population Generator can automatically keep track of all objects it has spawned. When it detects that an object has been destroyed, it will automatically spawn another, up to the max amount you specified.

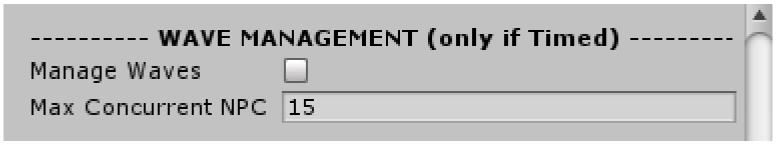
Example: You want 500 total enemies in your level. But you only want 15 max enemies on the map at once.

So you set “Total to Spawn” to 500



---------- SPAWN SETTINGS ----------
▶ Objects To Spawn
Spawn Radius 3.11
Total To Spawn 500
Area Type Spawn At Generator

and set “Max Concurrent NPC” to 15.

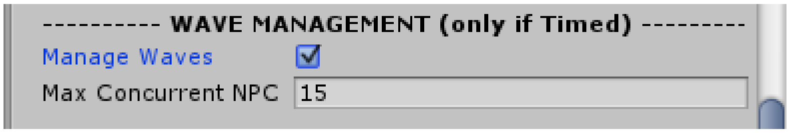


---------- WAVE MANAGEMENT (only if Timed) ---------
Manage Waves □
Max Concurrent NPC 15

Now when you play the game, the system will make sure that there will only be 15 max enemies on the map. Each time an enemy is destroyed, it will spawn another, until it reaches 500.

SETUP:

1) Set Max Current NPC
2) Check “Manage Waves” to turn this on.



---------- WAVE MANAGEMENT (only if Timed) ---------
Manage Waves ☑
Max Concurrent NPC 15

You're done.

Other Examples:

Performance Management: Your mobile game can only have 10 cars on the map, or the performance will drop. Each time a car crashes and explodes, spawn another to keep the number of cars always at 10.

Enemy Waves & Triggers: Combine this with the Trigger option. Create multiple Population Generators, name each like “Wave 1” “Wave 2” Wave 3”, etc... When the player walks into “Wave 1” trigger, activate an enemy wave of 15 enemies. When the player walks into “Wave 2”, activate an enemy wave of 30 enemies.

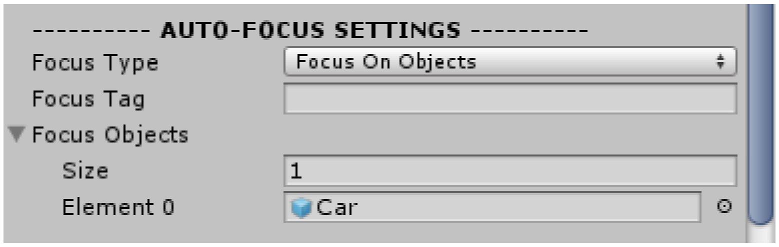
{11} OBJECT FOCUS

Have your NPCs focus on specific objects either when they stop on Idle, or with their Head Look Controllers.

1) On the Population Generator component:

Set Focus Tag to focus on nearest tagged object. So if you have a tag “Team 1”, your NPCs will always attempt to find the nearest player tagged “Team 1” and look at it.

Or Set Focus Objects. You can also set unlimited specific objects to focus.



---------- AUTO-FOCUS SETTINGS ----------

Focus Type	Focus On Objects
Focus Tag	
▼ Focus Objects	
Size	1
Element 0	Car

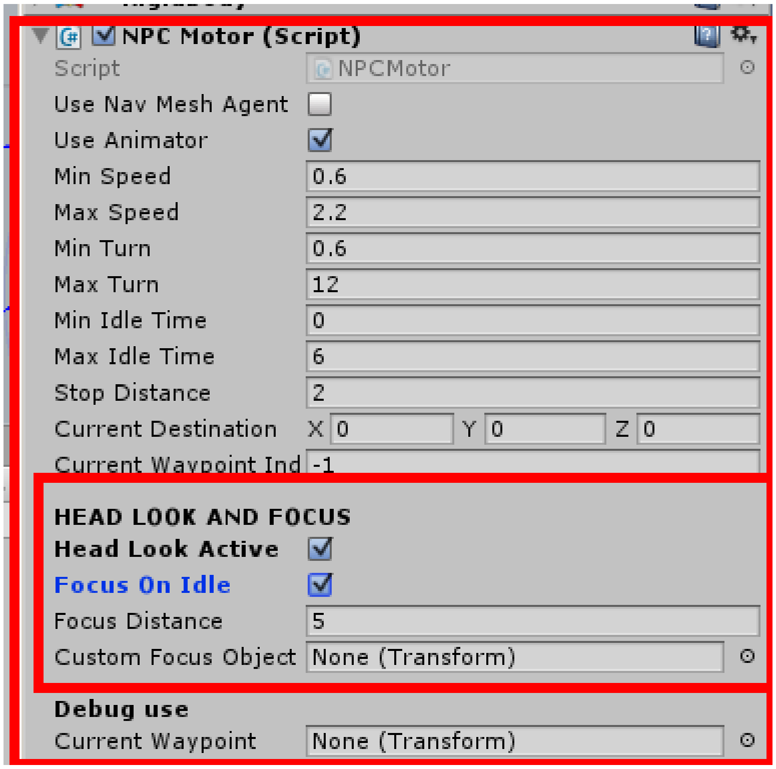
This requires the NPC Events component attached to your NPCs.
This works out-of-box with the NPC Motor. (See Showroom Demo)

2) Go to your Character. Find the NPC Motor component, or add one to him.

3) Check one or both:

Head Look Active: Will turn head towards specified objects of interest, as long as objects are within “Focus Distance” (See Showroom Headlook Demo). Requires Head Look setup.

Focus On Idle: Each time the NPC stops at a waypoint, it will turn its entire body towards the specific objects, before moving on. (See Showroom Demo)



☑ NPC Motor (Script)

Script	NPCMotor
Use Nav Mesh Agent	☐
Use Animator	☑
Min Speed	0.6
Max Speed	2.2
Min Turn	0.6
Max Turn	12
Min Idle Time	0
Max Idle Time	6
Stop Distance	2
Current Destination	X 0 Y 0 Z 0
Current Waypoint Ind	-1

HEAD LOOK AND FOCUS

Head Look Active	☑
Focus On Idle	☑
Focus Distance	5
Custom Focus Object	None (Transform)

Debug use

Current Waypoint	None (Transform)

→ To set up HEAD LOOK functionality, see next page.

{12} SET UP HEAD LOOK CONTROLLER

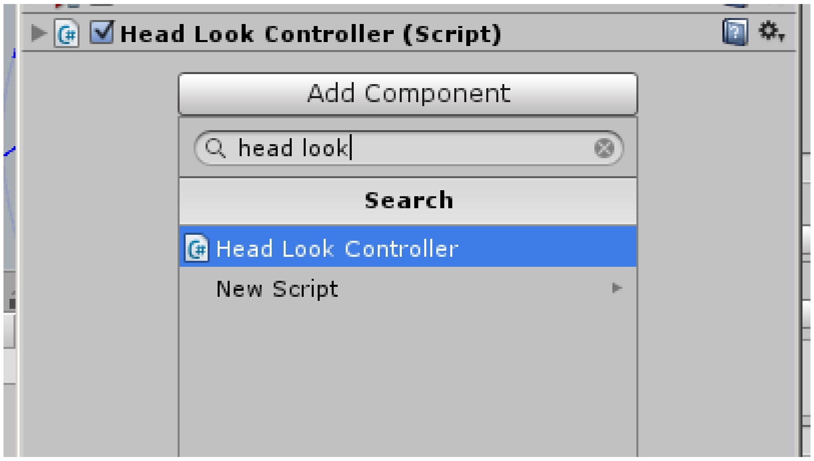
- Make sure you have first completed the steps in above SECTION 9: OBJECT FOCUS -

We included a bonus Head Look Controller integration with this product. This enables your NPCs to turn their heads to look at specific objects automatically. It will increase the level of realism in your scene.

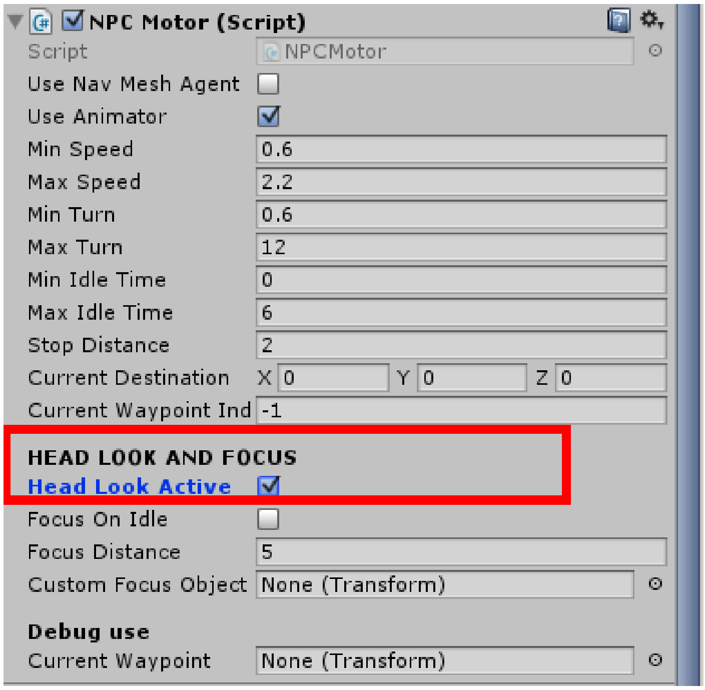
This works out-of-box on Humanoid models with Mechanim enabled, and using our provided Animation Controller. (See the demo prefab characters with HEAD LOOK and Showroom Scene)

1) Add the NPC Motor and NPC Events component to your character. NPC Events should automatically be added by NPC Motor.

2) Add the component "Head Look Controller".



3) Enable Head Look functionality in NPC Motor.


NPC Motor (Script)

Script	NPCMotor
Use Nav Mesh Agent	☐
Use Animator	☑
Min Speed	0.6
Max Speed	2.2
Min Turn	0.6
Max Turn	12
Min Idle Time	0
Max Idle Time	6
Stop Distance	2
Current Destination	X 0 Y 0 Z 0
Current Waypoint Ind	-1
HEAD LOOK AND FOCUS	
Head Look Active	☑
Focus On Idle	☐
Focus Distance	5
Custom Focus Object	None (Transform)
Debug use	
Current Waypoint	None (Transform)

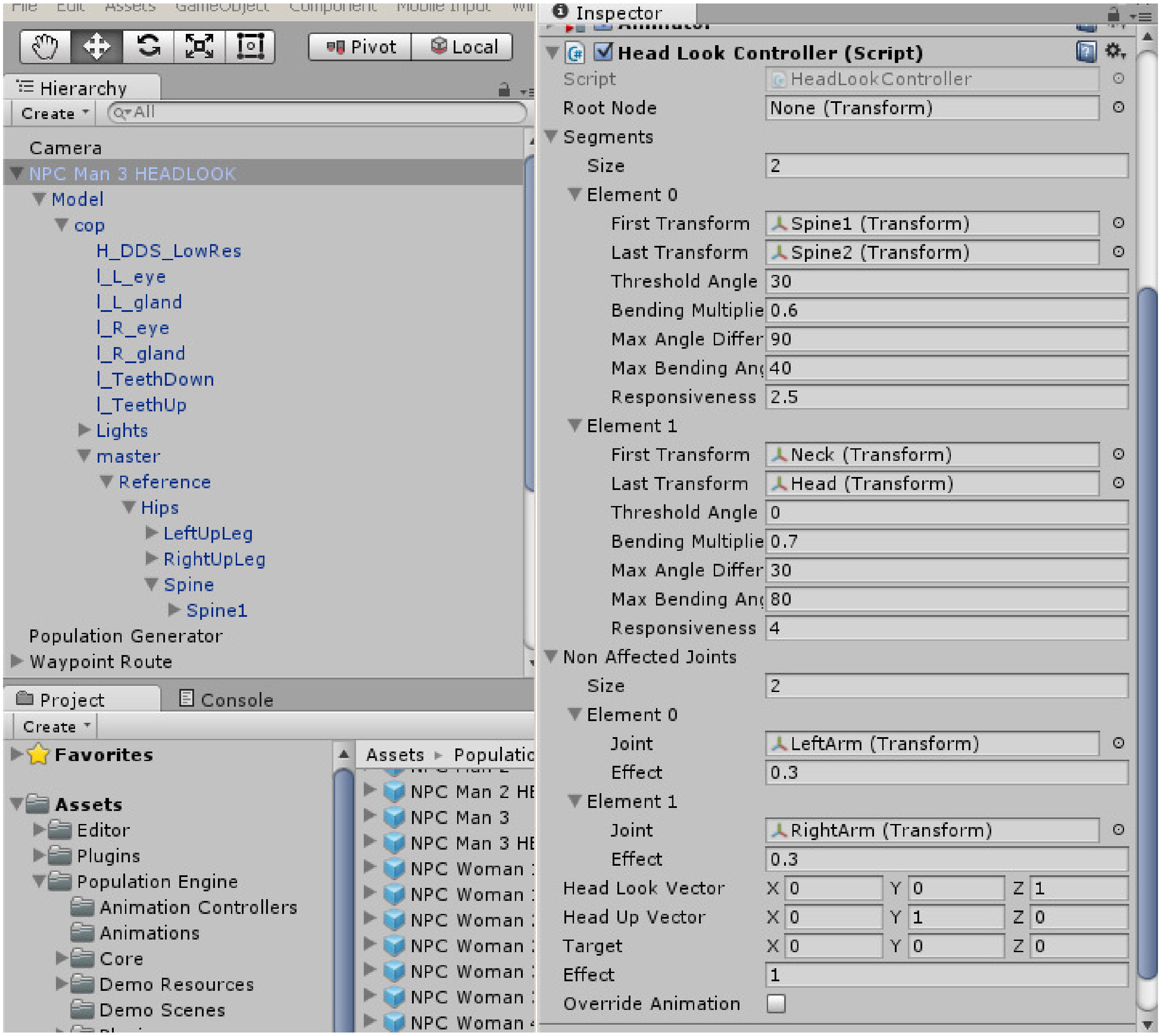
4) Insert your model's bones into the Head Look Controller slots.
→ - See next page for details -

5) Drag your character back into your /Assets/ folder to create a prefab out of it. Then insert this new prefab into the Population Generator.

You're done.

HEAD LOOK CONTROLLER BONE SETUP (continued):

Open up your character's model hierarchy and find the similar bone names. Insert the proper bones into the slots in this screenshot. Don't do anything to the numbers or anything else.
You only need to drag/drop the bones.



Head Look Controller (Script)

Script	HeadLookController
Root Node	None (Transform)
Segments	
Size	2
Element 0	
First Transform	Spine1 (Transform)
Last Transform	Spine2 (Transform)
Threshold Angle	30
Bending Multiplie	0.6
Max Angle Differ	90
Max Bending Ang	40
Responsiveness	2.5
Element 1	
First Transform	Neck (Transform)
Last Transform	Head (Transform)
Threshold Angle	0
Bending Multiplie	0.7
Max Angle Differ	30
Max Bending Ang	80
Responsiveness	4
Non Affected Joints	
Size	2
Element 0	
Joint	LeftArm (Transform)
Effect	0.3
Element 1	
Joint	RightArm (Transform)
Effect	0.3
Head Look Vector	X 0 Y 0 Z 1
Head Up Vector	X 0 Y 1 Z 0
Target	X 0 Y 0 Z 0
Effect	1
Override Animation	□

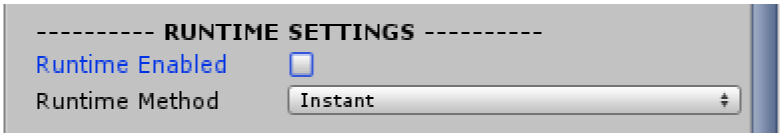
{13} MANUALLY PLACING OBJECTS IN EDITOR / PAINTING OBJECTS

The Population Generator component allows you to paint mass amounts of objects onto your scene in one click. Additionally, it can use Obstacle Awareness to intelligently paint objects around obstacles.

Example uses: Manually spawn people around cars and buildings, and avoid spawning on top of cars or inside buildings. Mass- generate flowers only on grass areas and not on top of rocks that may also be in the grass area. Generate trees only on terrain, and avoid overlapping buildings that are also on the terrain.

SETUP:

If you're only using Manual placement and not runtime, remember to first UNCHECK "Runtime Enabled"



---------- RUNTIME SETTINGS ----------

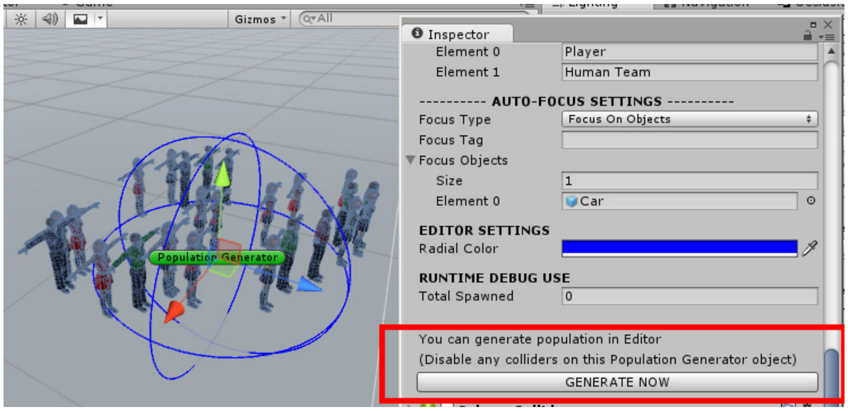
Runtime Enabled □

Runtime Method: Instant

Steps are the same as you would setting up for runtime generation.

1) Insert the prefabs you want to spawn
2) Set the Number to Spawn and Spawn Area Radius
3) Select whether you want to spawn objects at the Population Generator location or at waypoints.
4) If spawning at waypoint locations, remember to load the Waypoint Route into the waypoint slot.
5) Set how you want the the Height / Y Positioning
6) Set the Formation type (random? Or in straight line?)
7) Enable / Disable and set Proximity Check (checks whether next object will overlap existing object)
8) Set Auto-Focus Settings (these will give your manually-placed NPCs the ability to look at specific objects)
9) Enable / Disable Obstacle Awareness – this allows you to place objects around other objects, people around obstacles, trees around buildings, etc... Turn it off if not using.
10) Click GENERATE NOW. Objects will be painted right in your editor.

Warning: Setting a large Spawn number can crash your editor. Remember, if you insert 2000 objects, it's similar to copy/pasting 2000 of those objects into your scene, which can crash Unity if the prefabs are complex. You can set it to a lower number like 25 or 100 and click GENERATE multiple times instead.



Element 0: Player

Element 1: Human Team

---------- AUTO-FOCUS SETTINGS ----------

Focus Type: Focus On Objects

Focus Tag:

Focus Objects

Size: 1

Element 0: Car

EDITOR SETTINGS

Radial Color

RUNTIME DEBUG USE

Total Spawned: 0

You can generate population in Editor
(Disable any colliders on this Population Generator object)

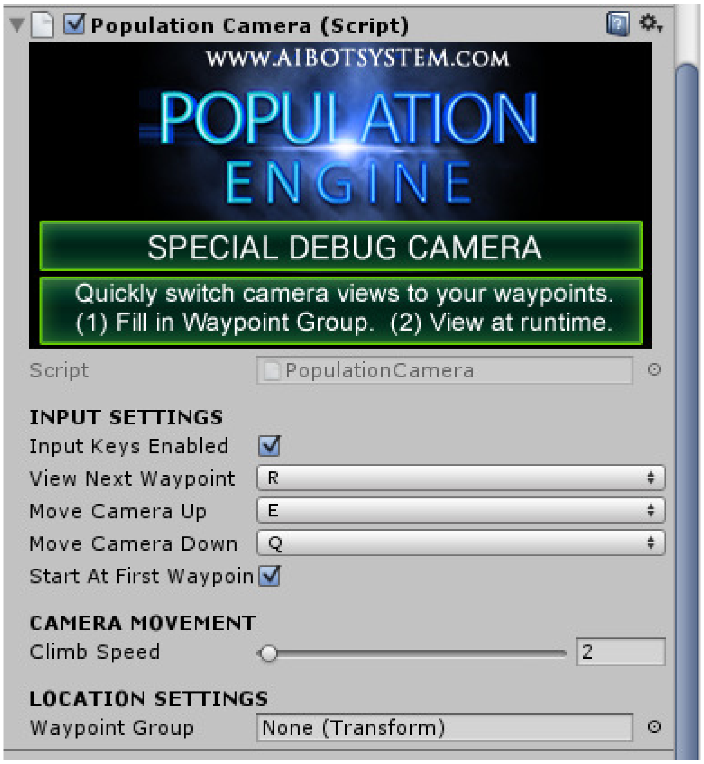
{14} SPECIAL DEBUGGER CAMERA (POPULATION CAMERA)

We created an optional Debugger Camera that you can use to observe your population and make sure things are running the way you want. This camera is **not** required for the system to work. It simply exists as a free bonus to help you edit your scene.

WHAT THIS CAMERA DOES:
Observe population activity at each of your waypoints without any additional set up / coding.

When you play your scene, it automatically moves the camera to your first waypoint. Then, you can use your mouse to look around the area and press UP/DOWN to move the camera up/down.

You can press the “View Next Waypoint” key to automatically move to the next waypoint, like a slideshow.



☑ Population Camera (Script)

WWW.AIBOTSYSTEM.COM

POPULATION ENGINE

SPECIAL DEBUG CAMERA

Quickly switch camera views to your waypoints. (1) Fill in Waypoint Group. (2) View at runtime.

Script: PopulationCamera

INPUT SETTINGS

Input Keys Enabled ☑
View Next Waypoint: R
Move Camera Up: E
Move Camera Down: Q
Start At First Waypoin ☑

CAMERA MOVEMENT

Climb Speed: 2

LOCATION SETTINGS

Waypoint Group: None (Transform)

You can use this to go through each waypoint to make sure that your population is moving around as they should. It's a fast way to check up on all the hotspots, instead of having to manually move your player into that area.

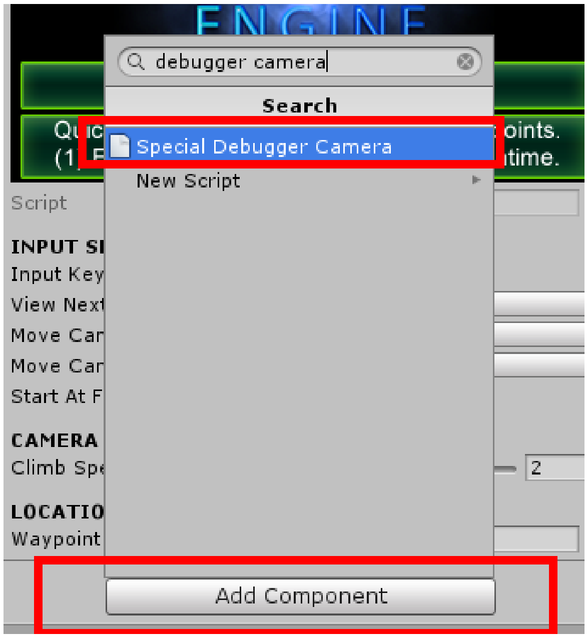
SETUP:

1) Add the Debugger Camera component to your Camera. →

2) Set the Waypoint Group. Simply drag your scene's Waypoint Route parent object into this slot. This is the Waypoint group you will observe. You can change waypoint groups by simply switching it out with another.

3) Set up the Input Keys. Make sure they don't interfere with the Input Keys for your game.

4) Make sure all other movements scripts on your camera are disabled.



Now run the scene and press the View Next Waypoint input key.

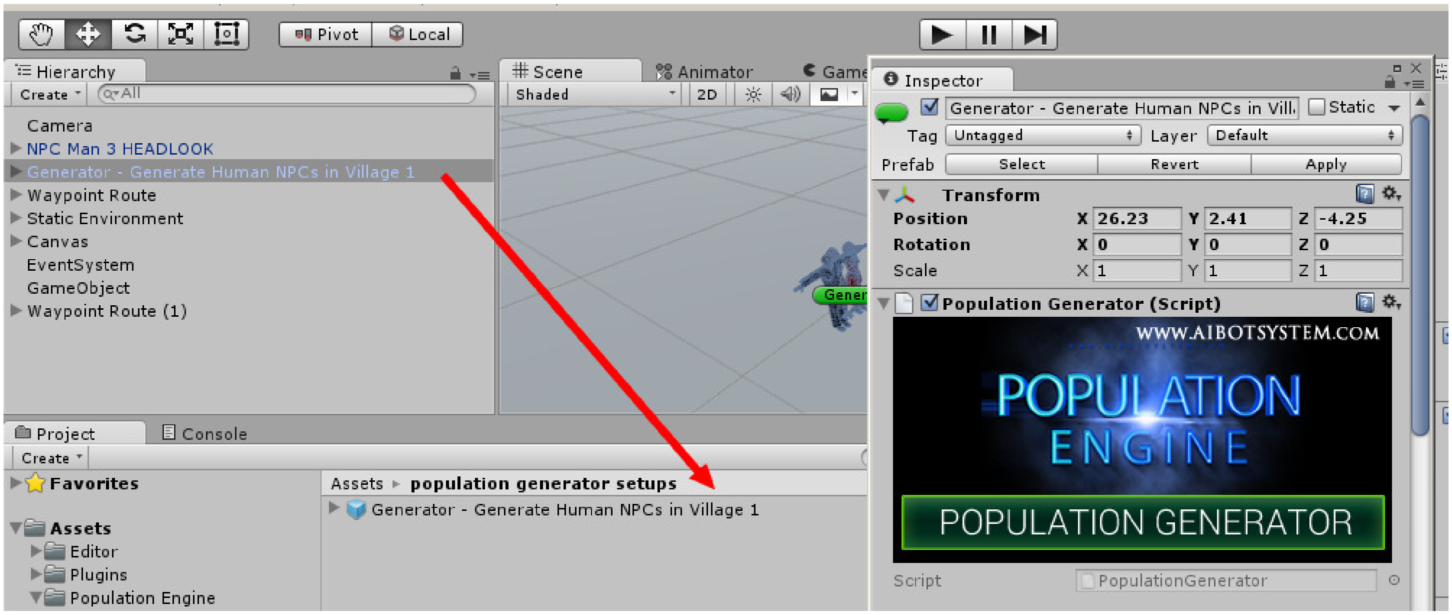
{15} HOW TO SAVE YOUR SETUPS / TRANSFERRING PROJECTS

So you loaded 50 character prefabs into your Population Generator and want to use the same characters in another scene, but don't want to insert 50 prefabs all over again.

The good news: We designed this tool with Unity's own Prefab system in mind. So, **to save your settings, all you have to do is drag your Population Generator object into your /Assets/ folder to create a prefab from it.** Then, if you want to use it again in another scene, simply drag it from the /Assets/ folder back into your scene. All your settings should stay intact.

1) Name your Population Generator something descriptive, like “Generate Human NPCs in Village 1”.
2) Drag it into the /Assets/ folder to create a prefab from it.
3) Drag this prefab into another scene.
4) Set up your waypoints again for the new scene and load up the new waypoint route.

Saving your settings basically requires drag and drop.



☑ Generator - Generate Human NPCs in Vill ☐ Static

Tag Untagged Layer Default

Prefab Select Revert Apply

	X	Y	Z
Position	26.23	2.41	-4.25
Rotation	0	0	0
Scale	1	1	1

Script PopulationGenerator

TRANSFERRING SETTINGS TO ANOTHER PROJECT

To transfer settings to another project:

1) Follow the above steps, turning your Population Generator into a reusable prefab.
2) Copy / Import all your character files to your new project.
3) Open your new project in Unity and wait for all the new files to finish processing.
4) Import Population Engine.
5) Copy the prefab from Step 1 to your new project folder. *Sometimes the prefab import is corrupted in the new project and shows up blank. Simply copy/paste again. This is not the fault of our product.*

{16} SCRIPTING INTEGRATION (C#)

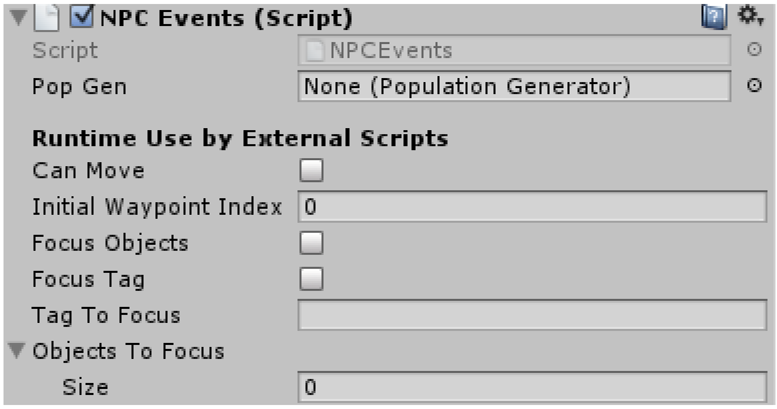
All of the above mentioned functionality can be done without scripting on your part. If you want further integration, the set up is quite simple.

There are 2 types of scripting integrations: 1) Individual NPCs. 2) Population Generator integration.

INDIVIDUAL NPC SCRIPTING

If you are not spawning NPCs, but are spawning non-movable objects such as Trees, buildings, props, etc... you can ignore this section.

Your NPC characters can run any existing scripts you had. Our system is **non-intrusive** and should not interfere with your existing character scripts. The Spawning phase instantiates your character prefab into the scene, just as you would instantiate any character. Therefore, you can safely attach all the scripts you need to your NPC.



☑ NPC Events (Script)

Script: NPCEvents

Pop Gen: None (Population Generator)

Runtime Use by External Scripts

Can Move □

Initial Waypoint Index: 0

Focus Objects □

Focus Tag □

Tag To Focus:

Objects To Focus

Size: 0

NPC Events: If you want your NPC scripts to integrate with the Population Generator that spawned it, your NPCs will need the NPC Events component attached.
You usually do not fill in any of these settings manually. The Population Generator spawner tries to detect this component on spawned objects and will use it to communicate with your custom scripts.
The **popGen** slot references the Population Generator that spawned it, and **is automatically filled out by the system.** Your code will need to reference this. All code should reference popGen when calling the methods of Population Generator.

Basic Script Setup:

```
        using PopulationEngine;
        public class YourClass : MonoBehavior{
                private PopulationEngine.NPCEvents npcEvents;     // refers the PopGen slot above

                void Start(){
                        npcEvents = GetComponent<PopulationEngine.NPCEvents>();
                        // usage: npcEvents.popGen.MethodName()
                }

                // rest of your code...

        }
```

SCRIPTING CONTINUED

The Population Generator component has a few public methods you can call from your scripts. For this manual, we will assume that you used the above code to set things up and will use the “npcEvents” variable name in our examples.

GETTING WAYPOINT DATA

You'll need to retrieve waypoint data in order to know where to move your NPC to.

You also need to keep track of the current waypoint index in your scripts. Depending on the Waypoint Path Type you chose, the waypoint route will not always begin at Waypoint Index 0, especially if you selected Random.

To get the initial waypoint index:

```
int firstWaypointIndex = npcEvents.initialWaypointIndex;
```

Allowing Movement: When the Population Generator has spawned the NPC and finished setting up its parameters, it will let you know that all is clear by setting NPCEvents.canMove to true.
To check if you're allowed to move:

```
bool canWeStartMoving =  npcEvents.canMove;
if (canWeStartMoving == true) { // move my NPC }
```

To Check if the Route ends (no more waypoints after this), given current waypoint:
(useful to know when your NPC should stop moving to next waypoint, especially if your Waypoint Path Type is LOOP ONCE)

```
bool doesRouteEnd = npcEvents.popGen.DoesRouteEnd(currentWaypointIndex);
```

To Get the Waypoint Transform:

```
Transform waypoint = npcEvents.popGen.GetWaypointByID(waypointIndex);
```

To Get position of waypoint at specific index:

```
Vector3 waypointPosition = npcEvents.popGen.GetCurrentWaypointPosition(waypointIndex);
// returns Vector3.zero if waypointIndex does not exist
```

To get position of NEXT waypoint in the route, given current index:

```
Transform nextWaypoint = npcEvents.popGen.GetNextWaypointPosition(currentWaypointIndex);
```

To get the index number of NEXT waypoint in the route, given current index:
(This is useful for your script to keep track of current and next waypoint indexes in the route)

```
int nextWaypointIndex = npcEvents.popGen.GetNextWaypointIndex(currentWaypointIndex);
```

GETTING INFO ABOUT OBJECTS TO FOCUS ON / LOOK AT

We already have set up object focus and head look controllers for you so you can use it without scripting. But if you prefer to code your own, here's the setup:

Focusing Specific Objects:
Check that npcEvents.focusObjects is true.
Grab the GameObject Array npcEvents.objectsToFocus

The objects in objectsToFocus are not sorted and come in the same order you placed them in, on the Population Generator component. Our included sample script, “NPCMotor.cs” shows one way of sorting these objects by distance and looking towards it when it comes near.

Focusing Specific Tag:
Check that npcEvents.focusTag is true.
Grab the string npcEvents.tagToFocus

You can then retrieve all GameObjects by the tag, tagToFocus and pick individual ones to focus on, as you wish.

HEAD LOOK CONTROLLER INTEGRATION

We have already set up head look controls in the “NPCMotor.cs” script which can be used by people without any programming experience. If you prefer to code your own integrations:

Add the Head Look Controller component to your NPC.

Use it in your code:

```
// turn on:
GetComponent<HeadLookController>().target = new Vector3 (0,0,0);
GetComponent<HeadLookController>().effect = 1;

// turn off:
GetComponent<HeadLookController>().effect = 0;
```

ANIMATOR CONTROL

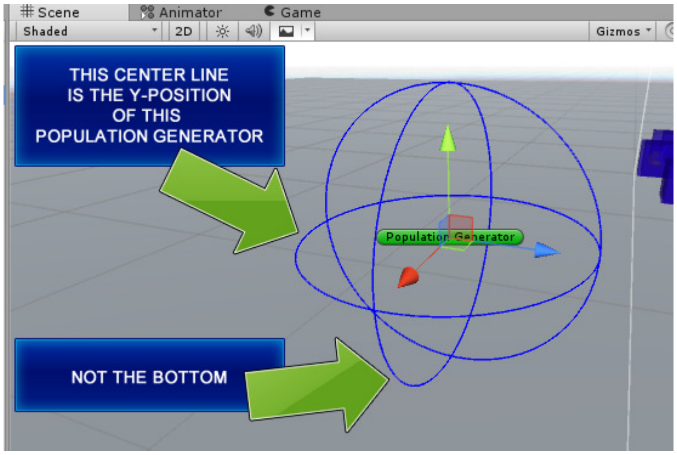
To integrate with our included Animator Controller, all you need to do is update the Animator Float “Speed” from your code:

```
GetComponent<Animator>().SetFloat("Speed", currentSpeed);
```

{17} TROUBLESHOOTING, NOTES, MOST FREQUENT QUESTIONS

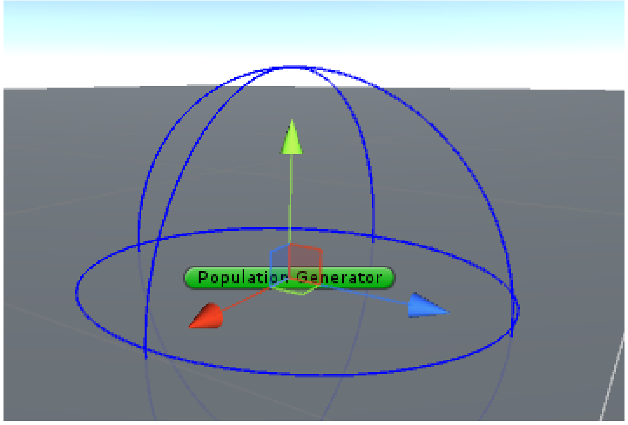
Note: Most common problems are caused by (1) Colliders and (2) The Y-position of the Population Generator or waypoints. Objects may not spawn if the system thinks that it is not supposed to spawn there. Please check your positioning and move it around above ground.

IMPORTANT: The Y position of the Population Generator is actually the CENTER of the sphere, not the bottom! This is the way object positions are done in Unity – they pick the center instead of the bottom to represent the Y position. Therefore, the real Y-Position is the center line of the sphere drawn in the editor of the Spawn Radius.

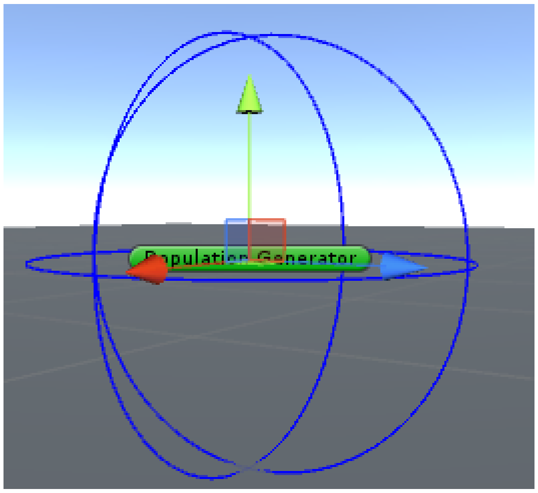


So, to spawn objects right at the ground level, you must place the Generator with the middle line at the edge (the grey area is the ground):

Right:



Wrong:

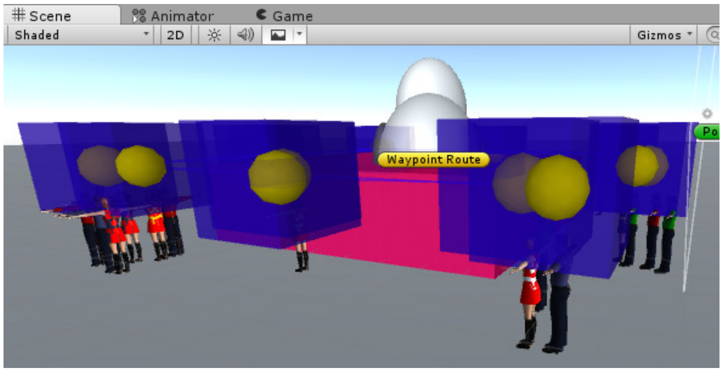


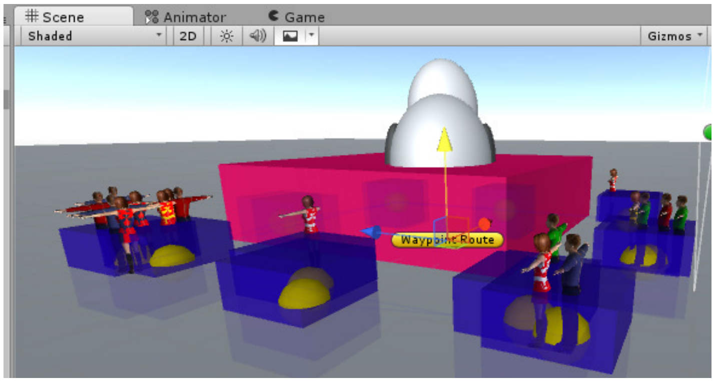
TROUBLESHOOTING & NOTES CONTINUED

My people are running around in circles, not going anywhere
This usually happens if your waypoints are placed at an unreachable spot. Make sure your waypoints are placed at the same ground level your characters are standing at.

Wrong: (waypoints placed too high)
(This will result in the NPCs running in circles)

Right: (waypoints placed at feet level of NPCs)



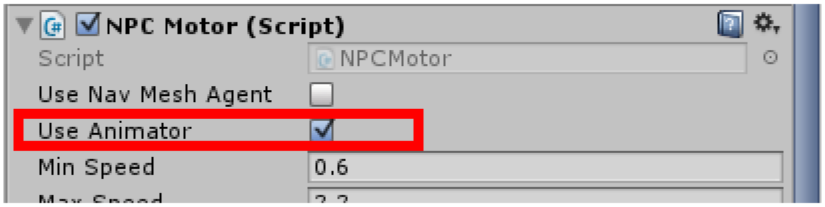


I have everything set up, but nothing is being spawned
1) If using Runtime, the “Runtime Enabled” setting on the Population Generator must be checked.
2) If you're Spawning at Waypoints, make sure your “Waypoint Group” slot isn't blank
3) Did you fill in “Objects to Spawn”?
4) Are you using Timed Spawning in the “Runtime Method”? Maybe your “Spawn Wait Time” is too long.
5) Disable Proximity Check. With this enabled, sometimes objects will not spawn because it detects another object within its proximity and wants to prevent overlapping each other.
6) Disable “Use Probability” or check that your “Chance to Activate” isn't too low.
7) Are you using Obstacle Awareness? Make sure you inserted tags in the “Spawn on Tags” slot.
Make sure your spawner's position in on top of a tagged area you can actually spawn on.
To check if this is causing the problem, disable Obstacle Awareness.
8) If your “Y-Position” setting is using Spawner Height, make sure your Population Generator or waypoints are above the ground you're spawning on.
9) If your “Y-Position” setting is using Raycast Height, make sure your spawner isn't overlapping too many other colliders.
10) Check any third-party scripts on your objects – make sure they are not destroying the object as soon as they are spawned.
11) If you've enabled Obstacle Awareness, also set “Y-Position” to “Use Raycast Height” as it works best with Obstacle Awareness.

TROUBLESHOOTING & NOTES CONTINUED

My character not animating. I have everything set up: Animator Controller, NPC Motor, Humanoid, Mechanim-enabled, etc...

1) Did you forget to enable “Use Animator” on the character's NPC Motor?



NPC Motor (Script)

Script NPCMotor

Use Nav Mesh Agent □

Use Animator ☑

Min Speed 0.6

Max Speed 2.2

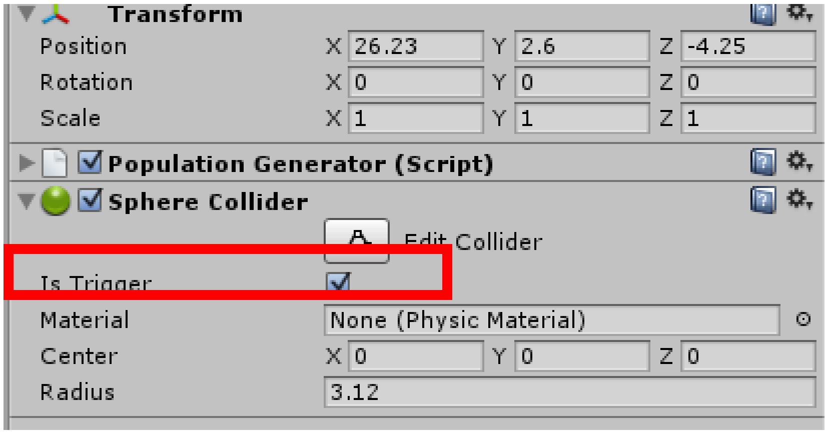
2) Are the Animations in the Animator Controller blank? Sometimes people forget to import animations so the Animation Controller has no animations loaded.

3) Make sure there is only ONE Animator Controller on your character, and that it's at the same level as the NPC Motor (not as a child or parent object).

TROUBLESHOOTING & NOTES CONTINUED

I want to use Triggers to spawn but when my player walks into the collider, nothing spawns.

Make sure your Population Generator object has a collider, and that "Is Trigger" is checked



Transform

Position X 26.23 Y 2.6 Z -4.25

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Population Generator (Script)

☑ Sphere Collider

Edit Collider

Is Trigger ☑

Material None (Physic Material)

Center X 0 Y 0 Z 0

Radius 3.12

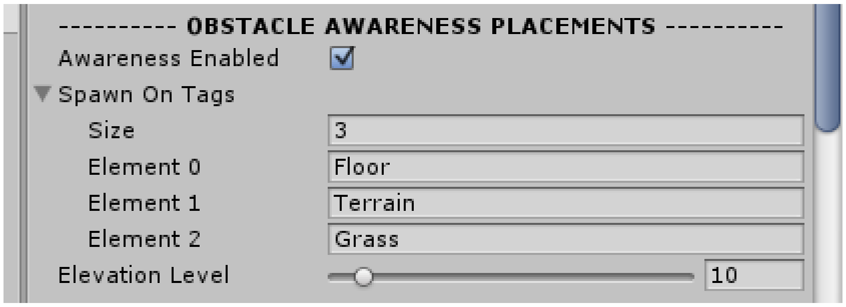
My objects are all spawning on top of each other

You're probably using "Use raycast Height" as the Y Position setup. Some users actually want to spawn objects on top of other objects (such as a Minecraft-type game), so we included this feature.

If you want to prevent objects from spawning on top of each other, you can simply do:

1) Enable Obstacle Awareness
2) Enter the tags of your ground area and anything else you only want to spawn on. Do not enter any tags of your actual objects to spawn.

Now the objects will only spawn on top of objects that have those tags, and not on top of each other.



---------- OBSTACLE AWARENESS PLACEMENTS ----------

Awareness Enabled ☑

Spawn On Tags

Size 3

Element 0 Floor

Element 1 Terrain

Element 2 Grass

Elevation Level 10

NEED ADDITIONAL HELP?

1) Visit our website: www.AIBotSystem.com for video tutorials. Click the "Population Engine" section.

2) Contact us at aibotsystem@gmail.com with your Unity Invoice # and your support question. You can also contact us on our website.

3) **Asset Store Reviews:** Please do not leave tech support questions as Asset Store reviews. This is because we are NOT auto-notified when we receive a review. We also do not constantly check the Reviews page. This means it will take a long time for your question to get answered. So the fastest way of reaching us is through email (it goes to the developers directly). With that said, this is an evolving product, so we would appreciate a good review to let us know what we're doing right, so that we continue to do things the way you want :-)