NPC Populator for Unity
User Manual

Clockworks Games, info@clockworks-games.com

Introduction

Non-player characters (NPCs), especially free-roaming ones, add a lot of ambiance to your game environment. However, for many developers, it may not be a priority to add NPCs as compared with game elements that are more central to theme and action of the game, especially because adding NPCs may require significant developer effort.

The NPC Populator for Unity makes it easy and quick to add many free-roaming NPCs to your environment, without writing any code. Only some configuration is necessary.

The approach for the NPC Populator has been not to reinvent the wheel. Critical, complex capabilities including animation and navigation are already built into Unity. What is required are scripts that make these capabilities work smoothly together. This is what the NPC Populator provides. By using the NPC Populator, you do not need to write any code to add NPCs to your game.

Each game will have its own unique combination of environment and characters, whether created specifically for the game or obtained through the Asset Store. As long as these follow some straightforward guidelines, they will work well with the NPC Populator. In particular, humanoid characters need to be compatible with Unity's humanoid animation type, so that humanoid animation clips can be used.

There are two options for using the NPC Populator: The supplied NPC Factory creates many NPCs at once, having a variety of models (appearances), speeds, and locations. Alternately, you can create one NPC at a time.

Mass producing NPCs using the NPC Factory

These eight steps will guide you through creating many NPCs using the supplied NPC Factory. A video tutorial is available at https://www.youtube.com/watch?v=4AQlgeys_Pw.

Step 1: Download and Install the NPC Populator.

This is done in the same way as any asset that you would obtain from the Unity Asset Store. Assuming you have received this Manual as part of the NPC Populator, you have probably done this step already!

Step 2: Download and Install Unity's Standard Package for Characters.

From the Unity Assets menu, select Import Package and then Characters. Click Import on the dialog that pops up.

The Characters package includes animation clips for humanoid walking, running, turning, and idle that are used by the NPC Populator.

At this point you can verify that the NPC Populator is installed properly by running the Sample Scene.

- Under the File menu, choose Open Scene.
- Navigate into the NPCPopulator folder, then the SampleScene folder, then select and open the SampleScene Unity scene file. This opens a scene with a number of Space Robot Kyles and a variety of obstacles to walk around.
- Click on the play button, and the robots should all walk around the scene.
- In case the robots move but without walking animation, sometimes Unity has an issue importing an animation controller before its animation clips. This may have happed here by installing the Characters Standard Package after the NPC Populator. If this does happen, please exit and restart Unity. In our experience, this fixes the problem. (Please contact us if you have any issues.)

Step 3: Configure your environment with a navigation mesh (NavMesh).

You can of course use any environment of your choosing. The Unity Asset Store contains over 4000 environment assets, or you may create one of your own.

This 5-minute video from Unity teaches what you need to know in most situations for setting up the NavMesh: https://unity3d.com/learn/tutorials/topics/navigation/navmesh-baking?playlist=17105

For more detailed information, there are other videos in that series, and the details are described in the online Unity Manual: https://docs.unity3d.com/Manual/Navigation.html

One tip that we discovered is that you will get more lifelike results if you choose an asset in which easily walkable areas (such as sidewalks) and less desirable walkable areas (such as roads) are modeled as separate objects, so that the less desirable areas can be assigned a higher navigation cost. Otherwise, the NPCs will have no preference between walking on, for example, sidewalks vs. roads.

Step 4: Place Waypoints

The NPC Populator includes a waypoint prefab. Drag this into your scene and duplicate as many times as you would like, so that you place a waypoint at key junctions for the travels of your NPCs, for example at intersections, entrances to buildings, etc. All of your waypoint objects should be under an empty

object called Waypoints in the hierarchy. When roaming, NPCs will head for a waypoint. When reaching the destination, the NPC randomly chooses another waypoint as the next destination, with a preference for choosing a waypoint in the direction already facing if possible, to avoid more drastic changes in direction.

Step 5: Place a Center Point and Estimate the Radius of your Environment.

Create an empty GameObject and rename it to something like "Center." Position your Center object at approximately the center of your environment, in a walkable area, at ground level.

Your randomly placed NPCs will be placed within a specified radius of your Center object. Estimate the size of this radius. One way to do this is to create another empty GameObject and place it in your environment as far from your Center object as would be safe to place NPCs. Compare the coordinate of your Center and new GameObjects to get the distance between them.

Step 6: Import Humanoid Models

The Unity Asset Store contains over 1500 humanoid model assets. Many of these will be compatible with NPC Populator. The key is that they must use or be compatible with the humanoid animation type.

Usually the humanoid models will come as prefabs within your Project window. Click on the prefab to bring up the Import Settings dialog in the Inspector window. Then check on the Rig tab.

If the Animation Type is already Humanoid, this model should be ready to use.

If the Animation Type is not Humanoid, for example it may be Generic, try changing the Animation Type to Humanoid, setting Avatar Definition to “Create from this Model,” and click Apply. If that is successful, a checkmark will appear next to the Configure button.

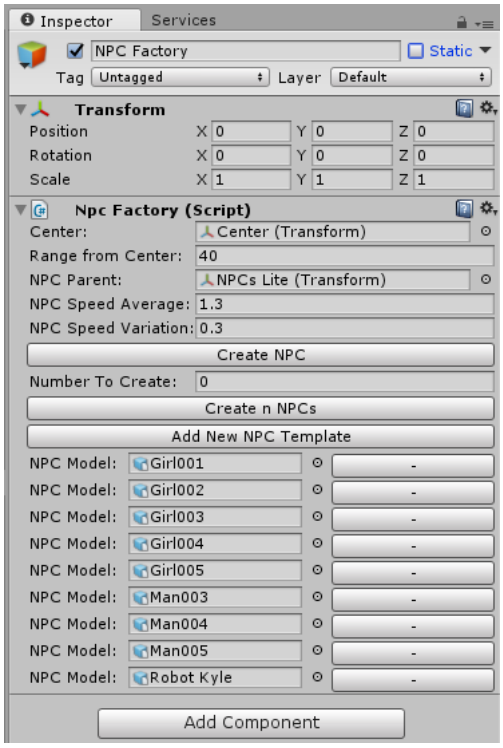
We have tried the following humanoid models with success:

- Space Robot Kyle from Unity – Need to switch from Generic to Humanoid Animation Type as above.
- Modern People and other packages from davidbinformatique.
- All models from Morph3D – Be sure to use version 1.6 or later, which has much better performance than earlier versions.
- Models that you create yourself using Autodesk Character Generator https://charactergenerator.autodesk.com/

Please let us know if you try other humanoid model assets and whether the packages are compatible or have issues. We will keep a list on the Clockworks Games web site for the benefit of all NPC Populator users.

Step 7: Create and Configure an NPC Factory.

Drag an NPC Factory prefab into your hierarchy window. The NPC Factory allows the rapid creation of as many NPCs as you want. Of course, there is a practical limit because your system performance and frame rate will degrade at some point. We have experimented with varying numbers of NPCs on different types of computers and have found the performance impact to be small for ~100 NPCs on even an average laptop.



The fields for the NPC Factory are filled in as follows:

- Center: Drag in the Center game object that you created in step 5.
- Range from Center: Type in the distance that you estimated in Step 5.
- NPC Parent: This field is optional. If you copy in a game object, all NPCs will be children of that game object. You may want to create an empty game object in your hierarchy named "NPCs" and drag that in.
- NPC Speed Average and Variation: Generated NPCs will be assigned a speed of the average that you specify plus a random value between minus and plus the variation. This avoids the effect where all the NPCs seem to be marching at exactly the same speed. 1.3 and 0.3 are reasonable values.
- Add New NPC Template: Click this button as many times as you have humanoid models that you would like to use. The NPC Factory will randomly choose among those models for each NPC that it generates.
- NPC Model: Drag an NPC model or prefab into each of these slots.
- "-" (minus button): Click on this to remove a slot from the list of NPC Models.
- Create NPC button: This will generate an NPC, choosing its speed and model randomly as described above.
- Number to Create: The number of NPCs to create all at once using the button below.
- Create n NPCs button: Click on this to create a number a NPCs.

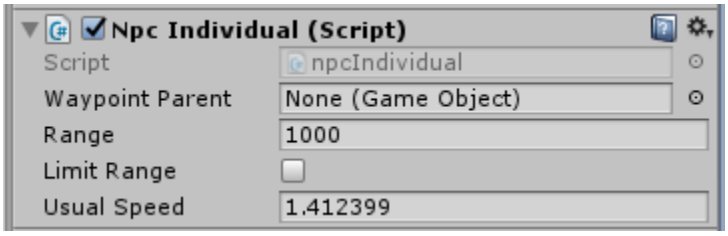
Step 8: Generate the NPCs!

Continuing to use the NPC Factory above, type the Number to Create, and click on Create n NPCs. This will randomly generate the specific number of NPCs from the variety of models and range of speed that you specified, with random locations and orientations based on the Center and Range that you specified, and also with a random avoidance priority so that one NPC will tend to have priority over another when they approach each other.

Create a Single NPC

As an alternate to the above, you can create NPCs individually without using the NPC Factory.

- Perform steps 1-4 and 6 from the instructions above.
- Drag your humanoid character model or prefab to your scene or hierarchy.
- Drag the npcIndividual script onto your character in the Hierarchy.
- In the Inspector, you have the following fields that you can adjust:



- The Waypoint Parent will be set automatically if all of your waypoints are children of an empty object called Waypoints as specified above.
- You can optionally set a range limit for how far away each scan for waypoints will look.
- You can set the Usual Speed. 1 is a slow walk; 5 is a fast run.

Acknowledgements

Space Robot Kyle is a free asset from Unity Technologies and is included here only as part of the sample scene.

Thanks to Morph 3D, davidbinformatique, All*Star Characters, and Autodesk whose humanoid character models work well with the NPC Populator. We expect that many other models in the Unity Asset Store are also compatible, but there are over 1500, so it is not feasible for us to test them all.