

ProGen

Getting Started Guide

By Dilmer Valecillos

Last Updated January 2, 2020

Table Of Contents

[Overview](#)

[Requirements](#)

[Getting Started](#)

[ProGen Parameters](#)

[Demo Scenes](#)

[Important Classes](#)

Overview

ProGen is a custom Unity grid system that allows creating procedural building by providing a scriptable object containing a set of parameters that defines how a building should be generated. Some parameters include how tall, wide, depth, a building should be and many additional options which are identified under **ProGen Parameters**

Requirements

- Unity 2019.2.15f1 or greater
- Unity PostProcessing Version 2.1.7 or Greater (Manifest Requirement shown below)
"com.unity.postprocessing": "2.1.7"

Getting Started

To Get Started with ProGen follow the next steps:

1. Create a new scene with an empty game object. Click on "Add Components" and search and select "Pro Gen" script as one of the options (See Fig 1.0)

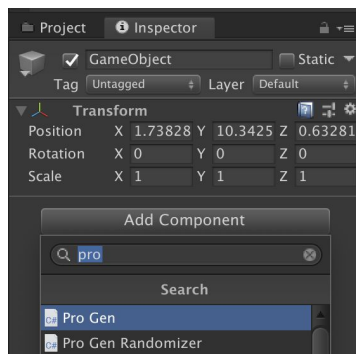


Fig 1.0

2. Click On Assets > Create > ProGen > CreateProGenTheme to generate a new theme (See Fig 1.1)

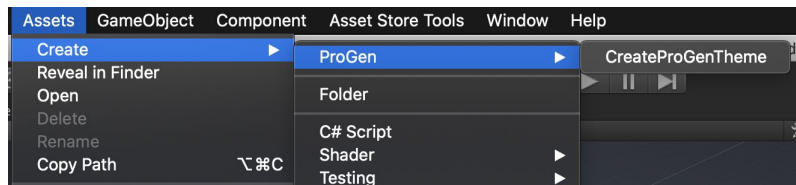


Fig 1.1

3. At this point a new scriptable object is available so go back to "Pro Gen" script and associate the **Theme** property with the newly generated theme (See Fig 1.2)

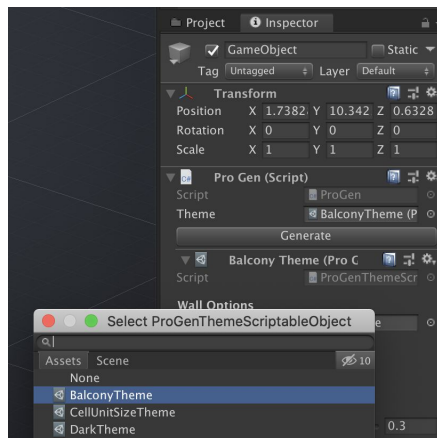


Fig 1.2

4. Now you need to assign prefabs to various areas of the ProGen Theme.
 - a. For Property "WallPrefab" search for "WallSimple" and assign it.
 - b. For Property "WindowPrefabs" increment the array index to 1 and assign the "WindowSimple" prefab.
 - c. For Property "RoofPrefabs" increment the array index to 1 and assign the "RoofSimple" prefab.
 - d. For Property "CornerPrefab" is **optional** but if you want to use it simply set it to use "WallCornerSimple" and you can toggle whether to use it or not by changing Property "Allow Corner Walls".
 - e. For Property "DoorPrefab" search for "DoorSimple" and assign it.
5. Now the final part is to click on the "Generate" button in the inspector to generate a new procedural building. Feel free to change any of the options available in ProGen to re-generate the building (See Fig 1.3)

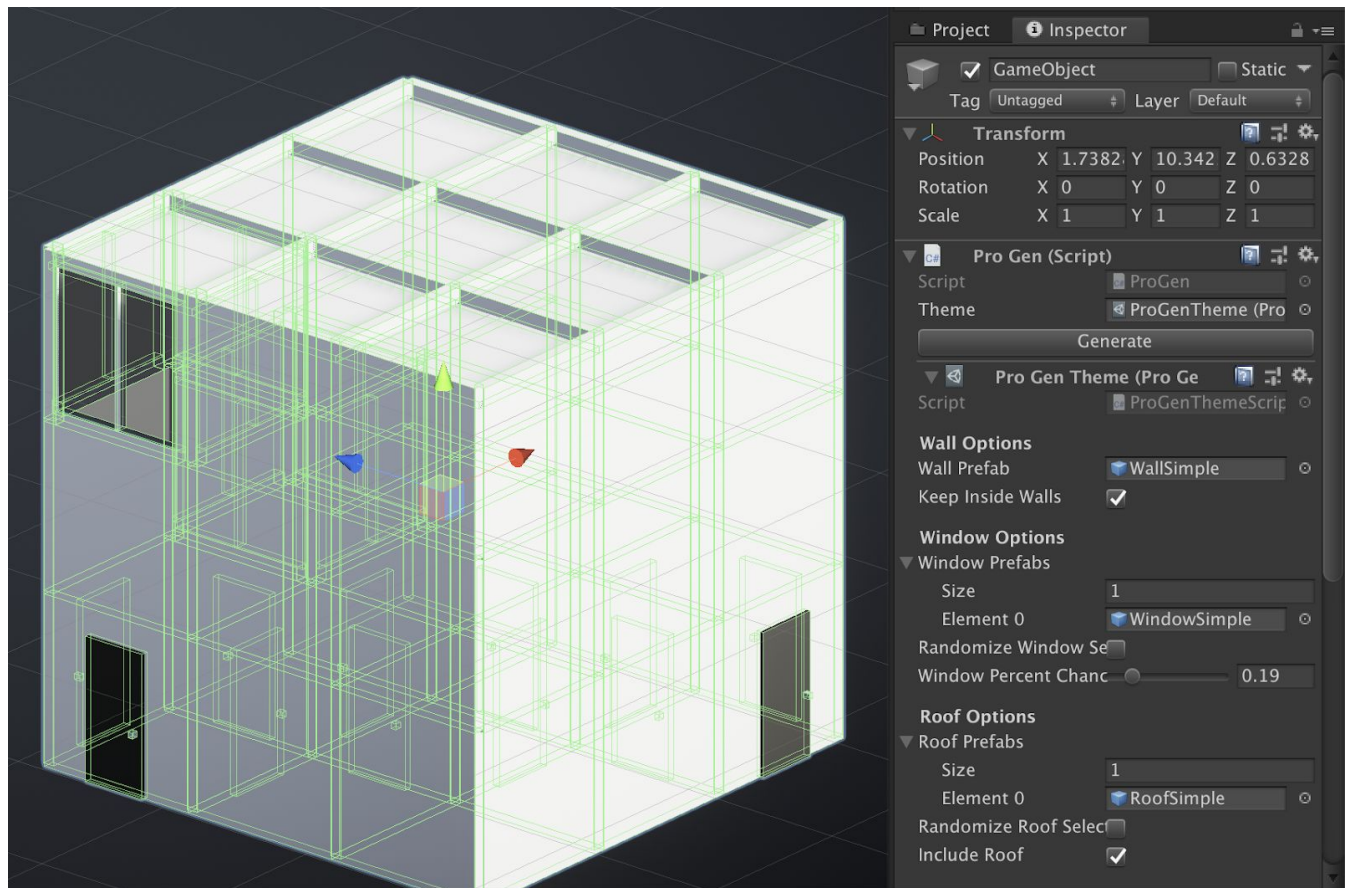


Fig 1.3

ProGen Parameters

All parameters in ProGen.cs are named with each purpose and are all listed below:

- Wall Options
 - Wall Prefab
 - Keep Inside Walls
- Window Options
 - Window Prefabs
 - Randomize Window Selection
 - Window Percent Chance (***Determines % of the time windows will be added***)
- Roof Options
 - Roof Prefabs
 - Randomize Roof Selection
 - Include Roof
- Wall Corner Options 2x2 Min
 - Corner Prefab
 - Allow Corner Walls
- Door Options
 - Door Prefab
 - Door Percent Chance (***Determines % of the time doors will be added***)
- Grid Options
 - Rows
 - Columns
 - Randomize Rows
 - Randomize Columns
 - Cell Unit Size
 - Number Of Floors

Important Classes

- ProGen.cs - This is the main class responsible for generating the buildings data structure and also rendering of all the building prefabs. To generate a building through code:

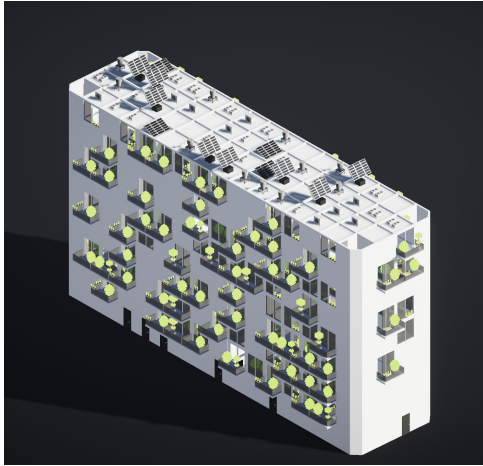
GetComponent<ProGen>().Generate()
- ProGenRandomizer.cs - This class allows you to randomize all parameters available in ProGen.cs and it provides options to set minimums and maximums values for almost every property.
- ProGenEditor.cs - This is an editor extension that allows you to keep track of changes in ProGen.cs and it incorporates ProGen Theme's as an option to a custom editor.

- ProGenExtensions.cs - This class contains a IsValid method to ensure ProGen is setup correctly, if you see a warning in the Unity console then it means you are missing a property.

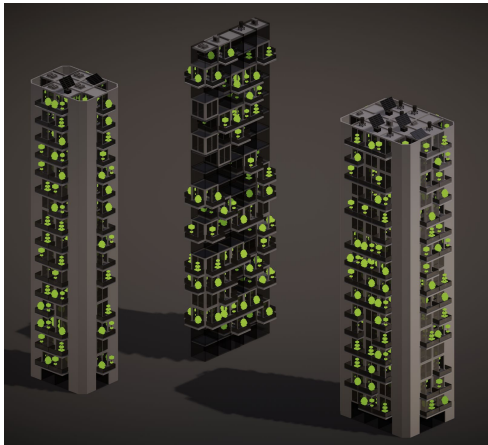
Demo Scenes

This project includes 8 different scenes to help you get started with ProGen. The demo scenes are listed below and each scene name describes its purpose.

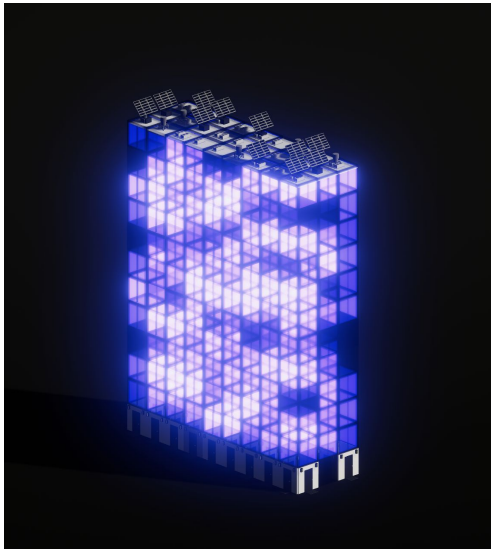
1. BuildingRandomizer



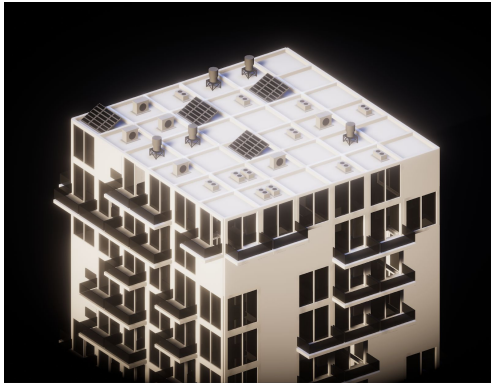
2. BuildingWithTreesCorners



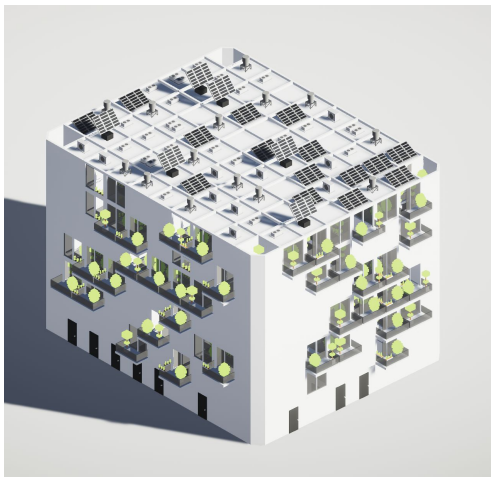
3. DarkDayBuilding



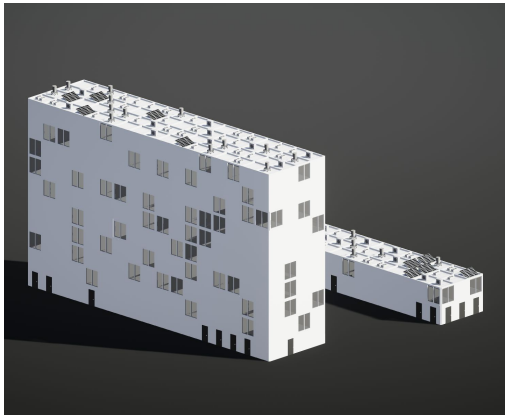
4. FocusOnRoofBuilding



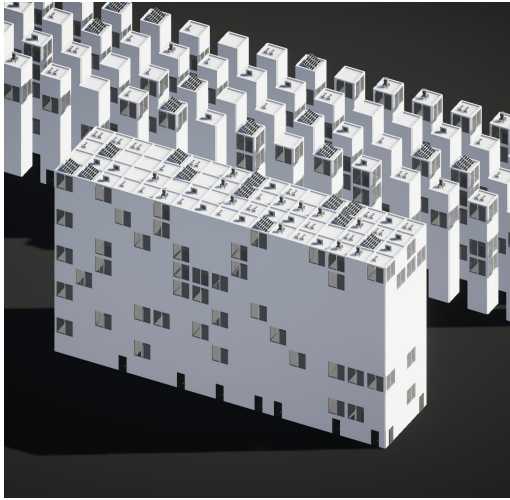
5. RoundedCorners



6. TwoBuildingsSimple



7. TwoBuildingsViewFromAbove



8. TwoBuildingsWithBalconies

