

CITYGEN3D BETA 0.81

QUICK START GUIDE



1. Introduction

Thank you for purchasing CityGen3D!

After reading this guide you should have a basic understanding of how to generate scenes based off real world map data using CityGen3D within Unity, all without writing any code.

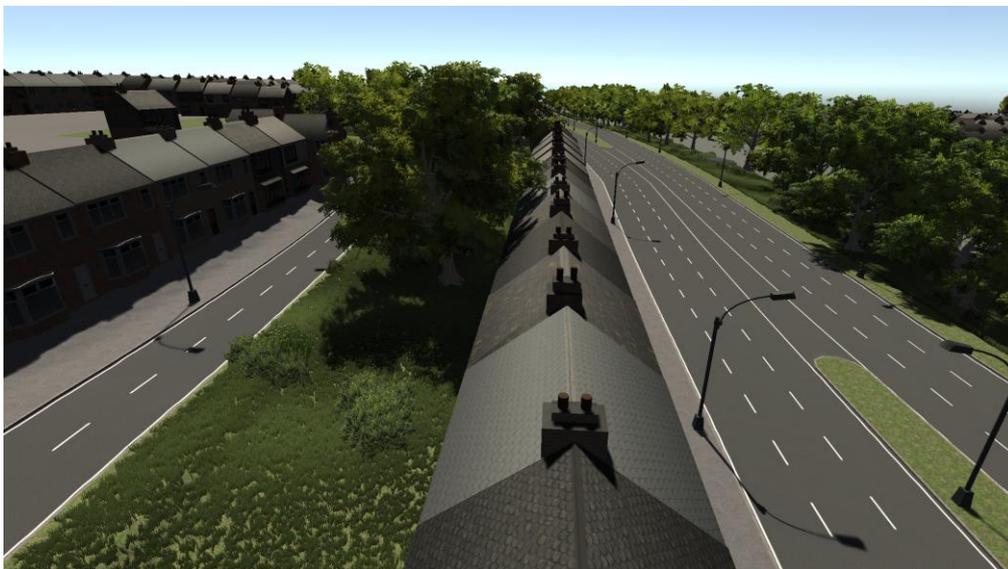
This guide will eventually mature into a complete manual covering all aspects of CityGen3D, but for now just concentrates on the fundamentals so you can get up and running as easily and as quickly as possible.



Please remember that CityGen3D is still work-in-progress and some features are missing or incomplete. Report any bugs you find so CityGen3D can be improved before final release.

You can visit the website at www.citygen3d.com and follow this thread on the Unity Works-In-Progress Forum for the latest information:

<https://forum.unity.com/threads/beta-released-citygen3d-procedural-city-generation-from-map-data.514677/>



2. Installation

CityGen3D Beta is now targeting Unity 2019.1 or later, to optimize some of the processes and take advantage of the very latest Unity features, including Unity Jobs with Burst Compiler and Terrain Editing on the GPU.

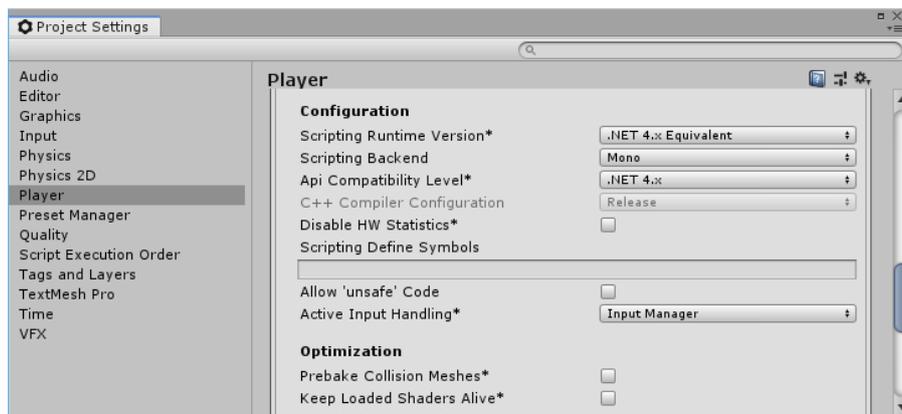
2.1 Installation - Unity

It is recommended to install Unity via the Unity Hub. Further information on can be found here: <https://docs.unity3d.com/Manual/GettingStartedInstallingHub.html>
Please install Unity 2019 and start a new Unity 3D project.

You then need to setup Unity to be compatible with a few Preview Packages:



From the Player tab on the Project Settings panel in the Unity Edit menu, change the Scripting Runtime Version to “.NET 4.x Equivalent” and API Compatibility Level to “.NET 4.x”.



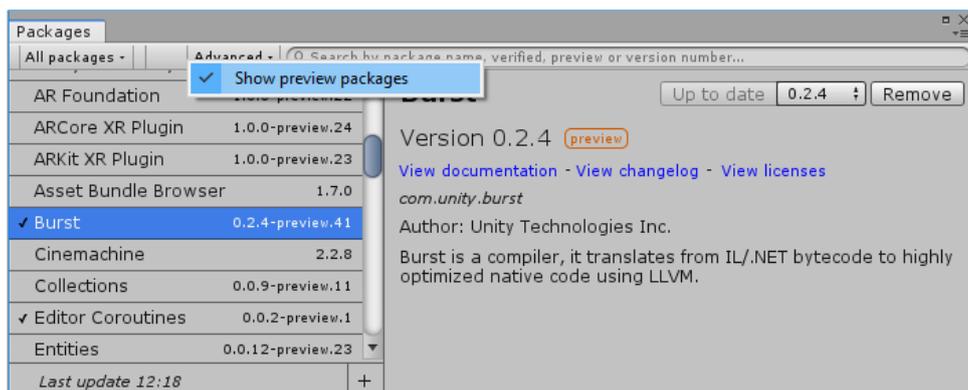
CityGen3D uses several Unity features that are still in Beta. These are called Preview Packages and should be installed via the Package Manager:



To ensure the required packages are listed, you need to make sure Preview Packages are not hidden by clicking the Advanced button and ticking “Show Preview Packages”. Then check that “All Packages” is listed using the filter at the top of the package list.

In turn, please install the following packages from the Package Manager:

- Burst
- Editor Coroutines
- Mathematics



2.2 Installation – Setup

CityGen3D is distributed in a zip file, which contains everything you need to get started. First extract the contents of the zip file to a suitable location on your computer.

If this is your first time using CityGen3D it is highly recommended that you use it in a brand new project, so you can experiment and learn the basic functionality separate to any existing project you may be working on.

Import the CityGen3D unity package into your project via the Unity menu:



Make sure all files are selected and click the Import button.

More information on Unity packages can be found here:

<https://docs.unity3d.com/Manual/AssetPackages.html>

Once CityGen3D is imported you can verify the installation by observing a new menu category called CityGen3D in your Tools menu in Unity. Don't click anything just yet though, there's one more step to finish setup!

In order to function correctly, CityGen3D uses specifically defined Layers that you will need to specify.

16: Map Surface	
17: Map Road	
18: Map Building	
19: Map Tree	
20: Map Entity	
21: Map Feature	
27: Feature	
28: Entity	
29: Building	
30: Highway	
31: Landscape	

To save you a bit of time you may have noticed another file in your extraction alongside the Unity Package. This has the layers all defined for you. Simply copy the supplied TagManager.asset file over the file of the same name in the ProjectSettings folder in your project.

Alternatively, you can add the layers manually using the Edit Layers window in Unity:

<https://docs.unity3d.com/Manual/Layers.html>

Once the layers have been added you are ready to start using CityGen3D! Note that the use of these layers is only required while CityGen3D is active in the Editor. Once you have generated your environment using CityGen3D there is no obligation to retain this layer structure in your project, so using CityGen3D does not restrict your layer usage at run-time.



Layers 9 to 15 are best used for any custom layers in your project. Future CityGen3D features may make use of layers 22 to 26

If you get the following error after importing CityGen3D it means you do not have required Preview Packages installed in your project. Please follow the instructions in section 2.1 for installing Preview Packages and ensure you are using Unity 2019.

```
! Assembly 'Assets/CityGen3D/Plugins/CityGen3D.dll' will not be loaded due to errors:  
Unable to resolve reference 'Unity.Mathematics'. Is the assembly missing or incompatible with the current platform?  
Assembly 'Assets/CityGen3D/Plugins/CityGen3D.dll' will not be loaded due to errors:  
Unable to resolve reference 'Unity.Mathematics'. Is the assembly missing or incompatible with the current platform?  
Unable to resolve reference 'Unity.Burst'. Is the assembly missing or incompatible with the current platform?  
Unable to resolve reference 'Unity.EditorCoroutines.Editor'. Is the assembly missing or incompatible with the current platform?
```

2.3 Installation - Updating

CityGen3D will be periodically updated with new features and bug fixes. Please note that while every effort is made to ensure compatibility between CityGen3D versions, this isn't always possible, especially after big upgrades.



Please back-up your project before attempting to upgrade Unity or CityGen3D! This is especially important while CityGen3D is in Beta.

While in Beta it is highly recommended where possible to only use a new version of CityGen3D in a brand new project. However, if you need to update CityGen3D in an existing project, here are some tips to try and make updating as simple as possible:

- It is recommended to completely remove the CityGen3D folder from the Assets in your project and re-import CityGen3D Unity Package.
- Before re-importing, delete the Data & Generator gameobjects from the hierarchy.
- After import, you should then add these to your scene as normal. This ensures you are getting up to date versions compatible with latest code.
- You can then load the database you are working on and reprocess the data before making further changes to your scene via the new generator.

3. Tutorials

The easiest way to get started is to load the New City scene from Assets\CityGen3D\Scenes. This will load a blank scene in Unity comprising only of two GameObjects: FreeCam (camera) and Environment (light and wind). You now have a suitable setup ready to use CityGen3D in your project.

This video demonstrates the basic workflow showing how to create an environment from start to finish in under ten minutes using default settings, for a really quick introduction.

It's recommended to watch this video as it covers a lot of what's being described in this manual.

<https://www.youtube.com/watch?v=YclH3N84FMY>

More videos will be available in future detailing some of the more advanced features.

For now, see if you can recreate this scene following the steps in the video and reading this guide.

4. Location

CityGen3D has two main functions. The first is to download and process the map data (“Data”), the second is to generate the 3D scene based on this data (“Generator”).

Both Data and Generator have their own Window within Unity for setting options.

So first thing to do is to add a CityGen3D data prefab to the scene via the menu bar so we can interact with CityGen3D via the Data window:



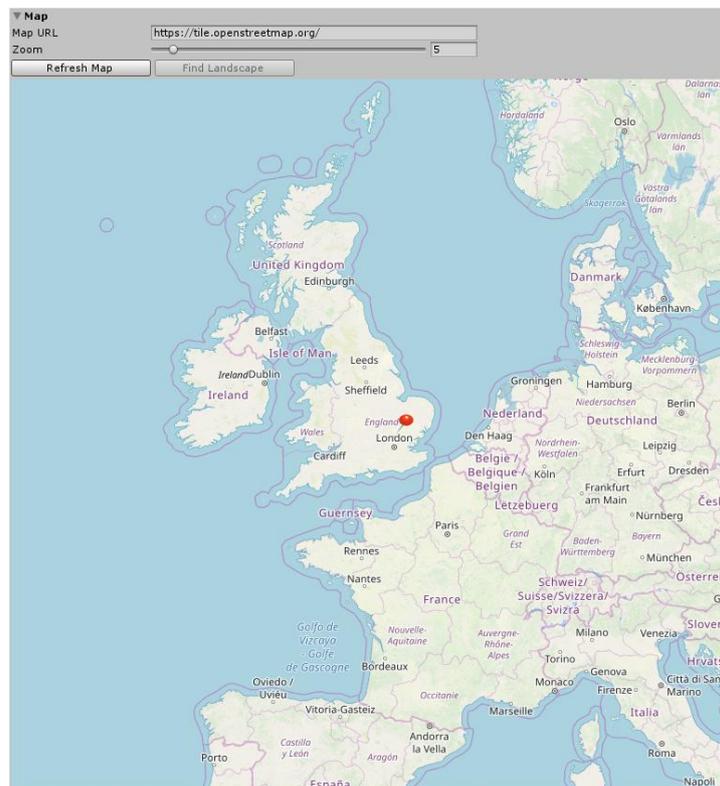
Clicking the Data menu item or pressing Ctrl+T will create a Data window in Unity. You can drag it around the screen and position it just like another other Unity window such as the Inspector, but it’s a good idea to make it fairly large so you can see the embedded map.

The Location tab is the primary panel on the Data window and is where you tell CityGen3D what data to download from OpenStreetMap (OSM). More information on OpenStreetMap can be found here: https://wiki.openstreetmap.org/wiki/Main_Page

4.1 Location - Map

The easiest way to select your location is via the map. You can drag the map around using Right-Click and select a new location using Left-Click.

The **Map URL** allows you to specify the server address from which the map is downloaded from in case this changes sometime in the future. Otherwise you should just be able to leave this on default setting.



When viewing the map from far away, the location is marked on the map by a red pin.

Zoom in further using the slider to see the map in more detail.

At higher zoom levels, a shaded grey area shows the precise extent of your currently selected location and area.



This button helps you to find your currently selected location on the map and forces a redraw.



If you have already generated a landscape, you can select it and click this button to apply origin and tile settings from the selected landscape. This can be very useful if you want to download another data set and have it align correctly with existing terrains.

4.2 Location - Settings

Make sure **Data Source** is set to *Download* and give an appropriate **Name** to your location, such as a town, village, or city name.

If you already have a database for a specific location, having downloaded the data in CityGen3D previously, set your Data Source to *Database* and load the appropriate asset from Resources\Locations folder.

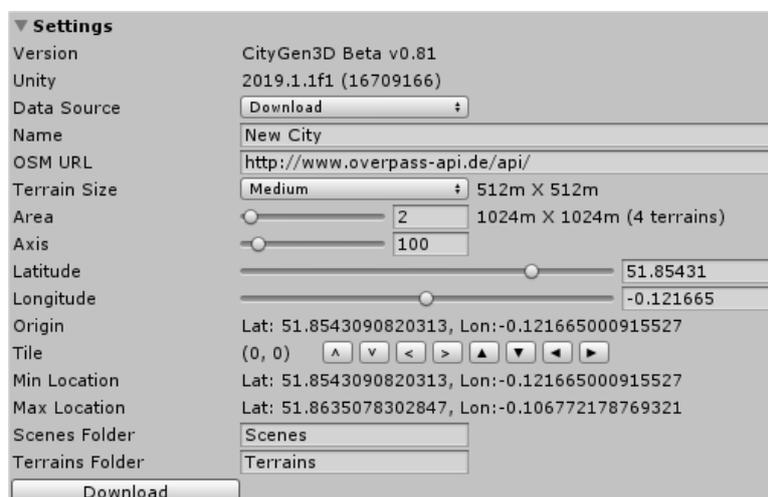
The *XML* data source shouldn't be needed for most users, but allows you to import raw OSM data from an external source if you are not able to access the OSM servers from within Unity.

It's possible in future that OpenStreetMap may change their API and you may need to update the **OSM URL** to point to a new location, or to try an alternative server. Otherwise simply leave it on the default web address.

In addition to the map, you can also select a location using the **Latitude** and **Longitude** sliders, or enter a specific location in the text boxes. The map will auto update as you move the sliders, although you may need to click the Refresh Map button to find your new location.

Note that when selecting a new location via the map, or using the Latitude/Longitude sliders, you are choosing the South-West extent of your environment and a new reference origin.

Having set your origin you should subsequently move around the map using the **Tile** arrow buttons only, instead of clicking on the map or moving the sliders.



Preserving the origin in a project is an important concept in CityGen3D because it allows you to download other data sets and have them align to previously created terrains. You could even have multiple users in your team working on different areas of your environment knowing that the terrain tiles will later align seamlessly with each other because of the shared origin setting.

The tile values represent how many terrains to the East and North of the origin the selected location is, so the origin is always considered to be tile [0,0].

Clicking the chevron icons will shift your location exactly one tile in chosen direction. The solid arrow buttons will shift your location enough distance to cover the current area size.

So you could start a project by selecting a new location on the map, which would set the origin coordinates and reset the tile to [0,0]. You then download your chosen area size and generate these terrains using the Generator.

You could later expand your environment by using the arrow buttons to shift the location to be adjacent to your existing terrains, rather than attempting to manually select a new location that aligns with it. Retaining the same origin across different groups of terrains within your project ensures that terrain heightmaps and textures seamlessly align with each other.



If you accidentally select a new origin in your project and want to create a new group of terrains to align with existing terrains, you can use the Find Landscape button to reset the origin based on an existing terrain and use the arrow buttons to select a new location.

Use the **Area** drop-down to select the data you wish to download and process. Note that CityGen3D will also download any OSM data outside of this area within a one hundred metre margin. It will also download all nodes and ways outside of this area if they belong to ways or relations that appear partially inside it. This helps to ensure that CityGen3D has all the data required to create an accurate map for your area.

The origin, tile values, and area size all contribute to the currently selected data area, which at higher zoom levels appears shaded on the map. The bounds of this shaded area are also specified with the **Min Location** and **Max Location** labels. Naturally if the current tile value is [0,0] then the Min Location is identical to the origin.

Each landscape is assigned an index based on their tile value where the terrain at origin has index of 0 and is therefore called "Landscape 0".

The **Axis** integer is used to auto number landscapes and specifies the maximum number of terrains that can be created across the horizontal axis in your current project.

So landscape indices are incremented by one for every tile increment across x-axis and incremented by Axis value for every y-axis increment. This means that the Axis setting must be at least equal to the Area size, and this will be set for you if needed when you increase the Area slider.

Although CityGen3D allows you to download and process areas as big as 16km, it is recommended to create big environments using multiple smaller areas and the tiling feature as detailed above.

In other words, ensure your Axis setting is large enough to accommodate your entire environment, but only work on small areas within that at any given time. This reduces the overhead on the Editor and makes it easier to work with.

Each database you create will save a tile reference as part of the filename so you can easily find different portions of your environment.

Later, your CityGen3D scenes will be saved to the specified **Scenes Folder**.
Terrain data and terrain layers are saved to the specified **Terrains Folder**.

A blue rectangular button with rounded corners and a white border, containing the word "Download" in white text.

When you are happy with your location and area size, click the Download button to start the download process from OpenStreetMap. If you can't see the Download button, ensure the Data Source is set to Download.

You must be connected to the internet for this to work. This will auto save the location database as an asset in Resources\Locations, so you can load it again anytime without having to re-download the data. Your Data Source will also change automatically to Database to prevent you accidentally selecting a new location while working on your scene.



Densely mapped locations with lots of data can take a long time to process. It is highly recommended that you start working with small areas while learning the tool so you can experiment with different options and learn the workflow much quicker.

4.3 Location - Processing

Once you have successfully downloaded data from OpenStreetMap, you need to process it so CityGen3D can use it. You have full control over how data is processed via the other tabs in the Data window, but for now we are just going to use the defaults



In future this manual will go into more detail on the OSM data format and how you can use the different Data panels in CityGen3D to customise your data processing, specifically to fit the needs of your project.

Surfaces and Roads are always processed by CityGen3D as they are fundamental to any scene. Use the **Filter** to select which of the other data types are processed. Buildings are not currently used by CityGen3D so you can speed up processing by ensuring these are not selected. You can always process them at a later date, once the Generator can utilize the data.

Roads are defined in OSM data by a polyline known as a "way". During processing, CityGen3D offsets these polylines with rounded edges to describe the extent of each road as a polygon. The **Arc Tolerance** setting determines how many points are used to round off the offset at the end of the road, therefore controlling how smooth road corners are.

The default is low for smoother curves in the map data. You can increase this value for a larger tolerance and fewer polygons, which may be preferred in a low-poly rendering style.



A variety of land uses are mapped in OSM, such as farmland, residential, meadow, and many more. But in some places there won't be any such surface in the data. The **Land Surface** setting allows you to tell CityGen3D what surface should be used as the default, where this occurs.

The **Sea Surface** setting tells CityGen3D which of your defined surfaces should be used where there is no land.

In order for CityGen3D to know where land and sea is in your location, a separate coastline data file should be loaded. **Coastline JSON** should be set to *Default Land* if your chosen location doesn't have any sea. This tells CityGen3D to create a Land Surface covering your whole location and requires almost no additional processing.

If your location does include a coastal boundary you will need an appropriate coastline data file, so CityGen3D can create suitable Land and Sea Surfaces.

Coastline files are named to include their latitude and longitude boundaries for easy identification. The "United_Kingdom_-11_49_2_63" coastline file indicates that it should be used for locations between Longitude -11 to 2 and Latitude 49 to 63, which is large enough to accurately process any location in the United Kingdom and Republic of Ireland.



Additional coastline data files will be made available in future releases, eventually for worldwide coverage.

Processing coastline data can take a while, so it is advised to untick the Coastline checkbox if you know your terrains are inland. Unticking the Coastline checkbox has the same function as having the Default Land coastline file active, but allows you to have a reference to a proper coastline file for your location and turn it on and off at your convenience.

For example, when generating an environment in England it's good to have the aforementioned UK coastline file assigned to Coastline JSON so it can be referenced as needed. You can then toggle the processing off using the Coastline checkbox if you are processing inland terrains.

Process

Click the Process button to start the processing. Once it has finished you should see some new GameObjects in the Unity Hierarchy under CityGen3D.

Load Generator

As you become more experienced with CityGen3D, you may wish to save different Generator prefabs for different projects. For instance you could have a Generator profile setup to utilise 3D graphics from different visual themes (eg Western, Cartoon, Sci-Fi). The **Generator Profile** field is where you will tell CityGen3D which profile to use. Then you can click the Load Generator button to activate it in your project.

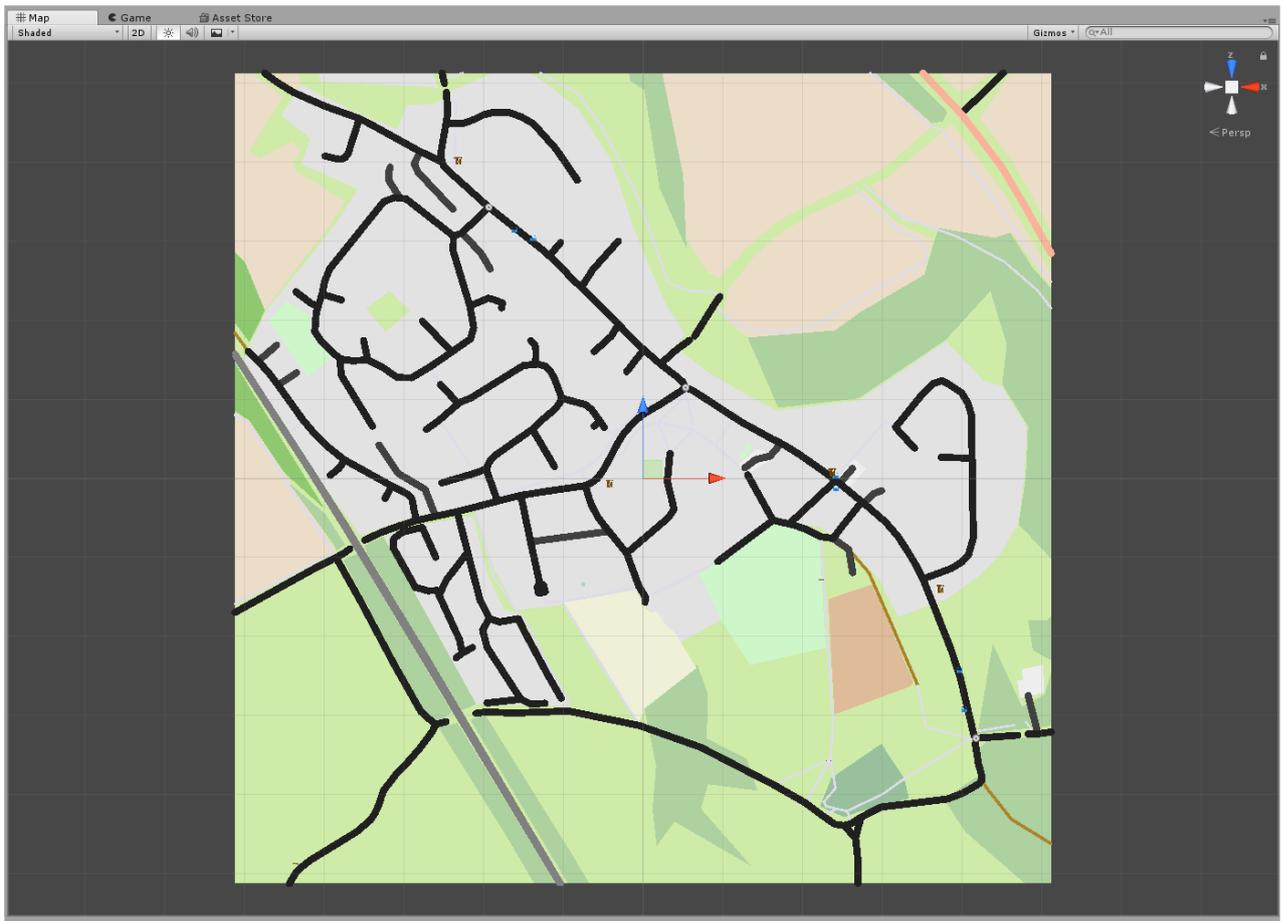
5. Map View

Now that we have processed some map data we can view the data in 2D. Press Ctrl+M or use the Unity menu:



Your Scene view will be renamed to “Map” and will be fixed to a locked Top-Down perspective. Select the CityGen3D GameObject in your Hierarchy and with focus on the Map window (the Scene view) press the [F] key.

This will zoom out the scene camera so you can see a rendering of the downloaded data in 2D. You can also use your mouse wheel to zoom in and out.



6. Heightmap

Now you have a Generator prefab in your project you can start creating a 3D scene. The Generator is split into modules, each with their own interface and you select the module you want to work on by clicking the relevant tab in the Generator window.

It's best to work through them from left to right, so we start with the Heightmap panel.

6.1 Heightmap - Terrains

The first thing that needs to be done is to generate the terrains on which everything else is built. A CityGen3D environment is split into one or more equally sized terrains, the size of which you already selected before downloading the data. So if your downloaded data is larger than one terrain in size, you will have the option of selecting which terrains to generate using the checkboxes.

Each terrain is attached to a Landscape object, which will later act as a parent object for all other prefabs generated above it at that location. This approach results in an environment that is broken down into smaller more manageable chunks.

If your environment is particularly large, then you'll want to consider saving each Landscape as a separate scene, which will allow you to load only have the portions of your world that you actually need in memory.

CityGen3D has a [Runtime](#) solution built in for managing this process for you, but for now simply ensure the **Save As Scenes** checkbox is enabled to have each Landscape saved separately, to make your scene compatible with it.

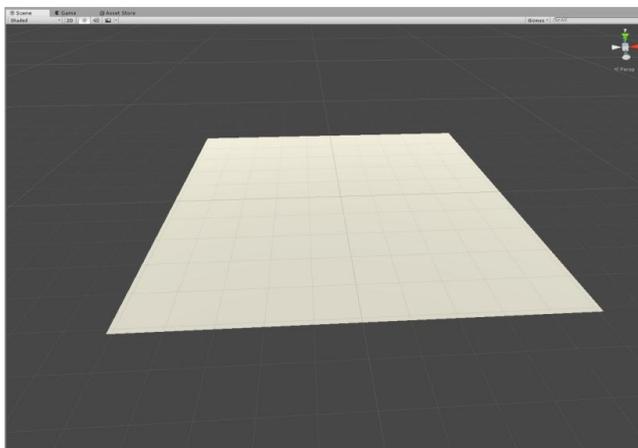
Generate Terrains

Click Generate Terrains which will make each of your terrains and they will appear in your Scene view as well as your Hierarchy named Landscape 0, 1, 2, etc

Convert To Scenes

Click this button to convert pre-existing landscapes into scenes for compatibility with scene streaming. This is redundant if your landscapes were saved as scenes on creation using the Save As Scenes option.

By default, your terrains will be flat planes, and currently won't have a texture applied to them. It should look something like this:





If you can't see your terrain, ensure your Scene view is active and not still in 2D Map view mode. Press Ctrl+M to toggle between 2D and 3D view at any time.

6.2 Heightmap - Settings

OpenStreetMap doesn't have detailed topographical data, so CityGen3D uses another data source for automatic heightmap generation based on real world data.

Select your chosen **Data Provider**.

NASA SRTM requires manual download of NASA's Shuttle Radar Topography Mission data, but is the most reliable method as it doesn't require communication with a third party server each time you use it.

SRTM data is freely available from several online sources, but the easiest way to identify the files you need is to download via this website: <http://dwtkns.com/srtm30m/>

You will need to register for free on the NASA website in order to access the files, please follow the instructions on the website. Once you have signed in, simply find your location on the map and download the required data files.

After downloading the zip files you need, you should unzip them and put them in your Unity project here: `Assets\CityGen3D\Resources\Data\SRTM`

Bing Maps and *MapQuest* do not require a manual download of data. Instead, CityGen3D communicates with the chosen server to download the required data on demand.

Not all locations are covered by all providers, so try a different one if your chosen area is not covered or if you have difficulty accessing the server.

The web address of the API can be updated if necessary using the **API URL** setting. But for now, the default should be fine.



In order to download heightmap data for your terrains you need to obtain a free Bing Maps Key from <https://msdn.microsoft.com/en-us/library/ff428642.aspx> or free MapQuest Key from https://developer.mapquest.com/plan_purchase/steps/business_edition/business_edition_free/register

Ensure you enter a valid key into the **Bing Key** or **MapQuest Key** field if using one of those services. An invalid key will result in data not being downloaded.

Offset allows you to align the downloaded heightmap data to OSM data if you find there's a slight mismatch. The defaults of 0,0 should be fine in most instances.

Water Depth adjusts how far below sea level the terrain can be.

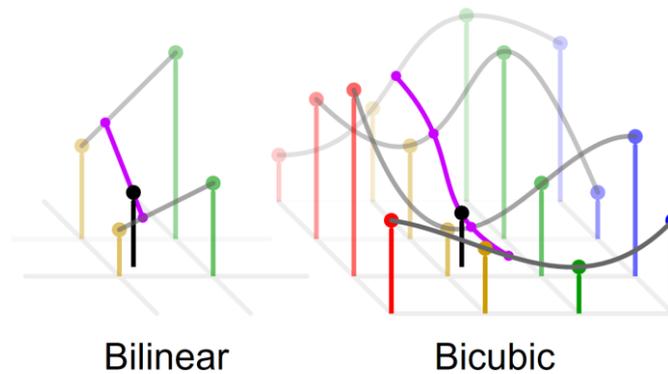
Peak and **Nadir** describe the highest and lowest points of your terrain in meters.

Tick the **Auto Range** checkbox to set these two values automatically based on the range of your downloaded heightmap data and your chosen Water Depth setting.
You should untick this option and set the Peak and Nadir manually if you intend to build your environment using multiple data sets with a shared origin. This is because all terrains in your project need to have the same height range in order for them to align with each other seamlessly.

Raw heightmap data is sampled at relatively low resolution. So in order to produce a high resolution heightmap CityGen3D interpolates the raw data in one of two ways, chosen via the **Interpolation** listbox.

Bicubic is the default and results in a smooth heightmap using spline interpolation of the surrounding raw samples.

Bilinear is a bit faster, but results in a more angled low-poly style heightmap due to linear interpolation between raw samples.



Apply Heightmap

Click the Apply Heightmap button to process SRTM elevation data and apply it to your terrains.

Download Heightmap

If using Bing Maps or MapQuest services, click the Download Heightmap button to download elevation data for your location.

Once the data has been downloaded a dialog should confirm the download has finished and you should see the heightmap applied to the terrain. Otherwise you will get an error message back from the server.

Common errors include:

Unauthorized – your key is not valid.

Internal Server Error – data not available for this area.

Gateway Timeout – likely a temporary problem with server.

Too Many Requests – server is blocking you.

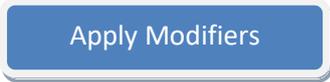


Bing and MapQuest can block users that make multiple requests in a short period of time. Because CityGen3D needs to make one data request per terrain tile, when working very large multi-terrain environments it is advised to download SRTM data so it can be processed without third party server access. Bing and MapQuest services are most useful for quick tests on smaller environments when you don't already have SRTM data.

6.3 Heightmap - Modifiers

You can apply per surface modifiers to the downloaded heightmap. It's recommended you have a negative **Altitude Offset** for each water surface type.

In order to prevent distortion of roads and pathways caused by modifying the heightmap, offsets are applied only when far enough away from them. Unticking the checkbox removes this protection and gives priority to the heightmap offset instead of nearby highways.

A blue button with rounded corners and a white border, containing the text "Apply Modifiers".

Click this button to apply the height offsets to the selected terrains.

6.4 Heightmap - Smoothing

After applying an offset through the Modifiers interface, you may wish to smooth the heightmap to remove vertical drops. This works in the same way as Unity's built in smoothing tool, but instead of having to paint manually on the terrain, the smoothing is applied uniformly across the heightmap.

To apply more smoothing, increase the **Iterations** setting, which controls how many times the smoothing is applied.

There are also options for adjusting the **Brush Size** and **Brush Falloff**.

Notice how the brush image changes as you move the Falloff slider, giving you control over the brush intensity.

A blue button with rounded corners and a white border, containing the text "Apply Smoothing".

Click this button to apply smoothing evenly across all selected terrains.

6.5 Heightmap - Flattening

Placing prefabs over uneven terrain can result in unwanted gaps appearing between the base of the object and the terrain mesh.

After placing prefabs over your terrain, either manually or automatically using the Roadside panel, you can use the Flattening feature to fix this for you.

For example, to flatten terrain under buildings you would set the **Apply To** layer mask to include the Buildings layer.

The **Radius** setting controls how far away from the prefab the flattening is applied. The lower the radius, the steeper the incline on the terrain.

A blue button with rounded corners and a white border, containing the text "Apply Flattening".

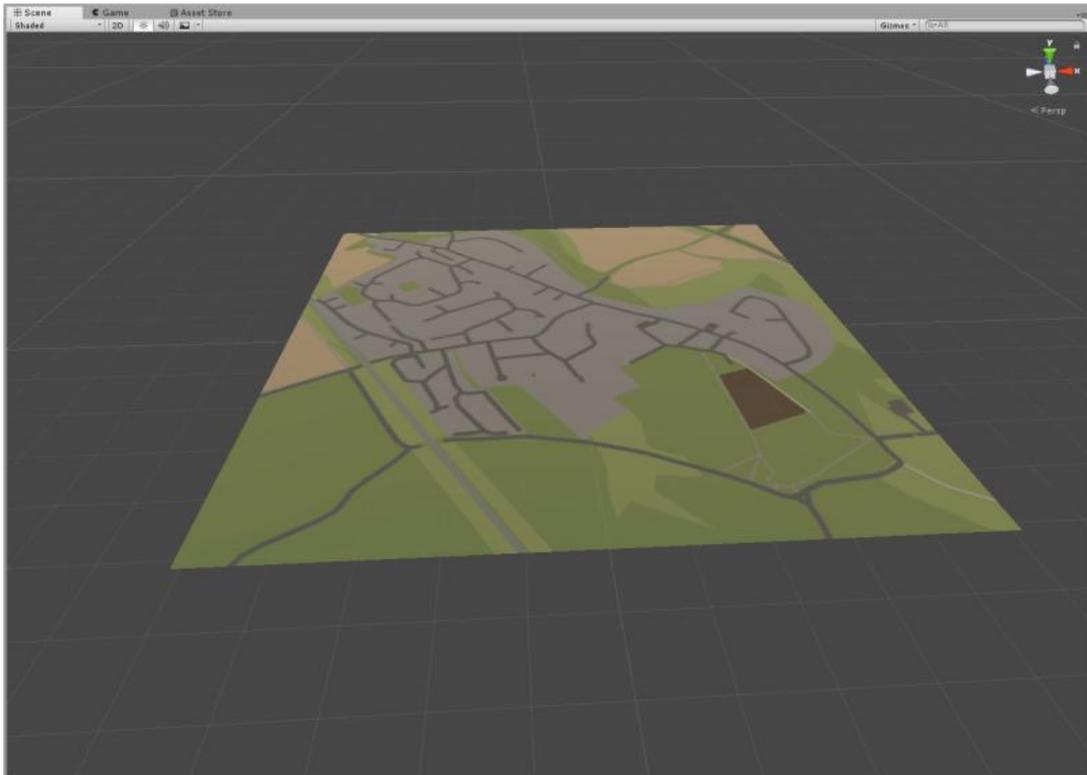
Click this button to apply flattening on selected terrains, under all colliders associated with objects in specified layers.

7. SplatMap

Terrains in CityGen3D can be automatically textured based on your material library and surface rules that you define.

Texture Base Surfaces

To texture your terrains using your chosen materials and rules for each surface type, click this button while one or more Landscapes are selected in the scene. Your terrains will then appear textured appropriately, similar to the image below.



Texture Custom Surfaces

It is also possible to detect surface types from spawned prefabs and texture them appropriately. Take a look at the House 05 prefab in the CityGen3D Prefabs folder. That has a Lawn plane with a Map Surface component attached. The surface type is set to Grass to ensure that when the prefab is spawned you can click this button to texture the area the lawn covers as defined by your texturing rules for Grass. You could have hundreds of these houses and all their lawns would be painted appropriately with one button click. So when designing your prefabs consider using this terrain technique sometimes instead of adding additional geometry.

8. Highways

The unique way in which roads are created in CityGen3D allows for complex road networks to be generated automatically. You can make adjustments to the look of your roads by interacting with the Highways panel.

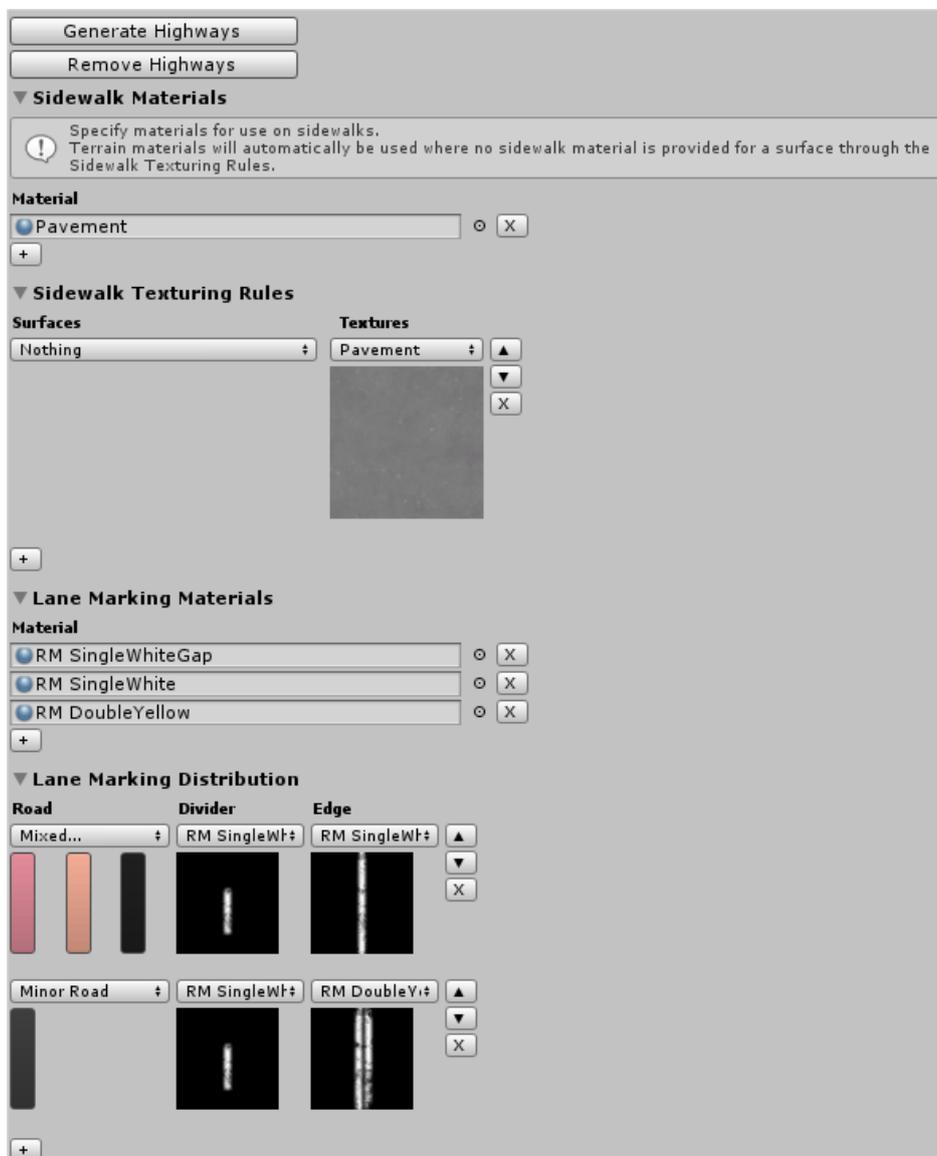
Each road type can be associated with different road markings using the Markings Distribution controls. Allowing you to customise it quite easily for different locales.

By default, sidewalks will be textured with the same material as specified on the Splatmap panel for the same surface type at that location.

You can override this with specific sidewalk materials if required.

Generate Highways

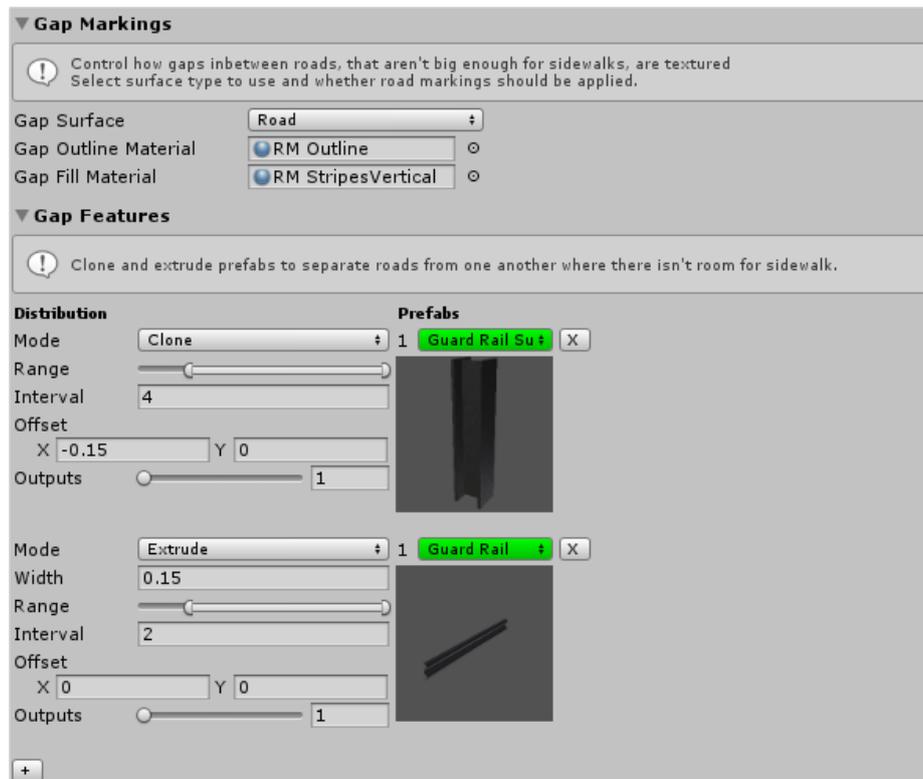
Click this button to create your roads, which will automatically generate suitable batched geometry and offset the terrain heightmap as appropriate.



In CityGen3D, roads are generated by creating geometry to form blocks where no road exists. The road itself is simply the terrain mesh with road marking decals on it.

These blocks are created by extruding sidewalks, but in some cases it's possible to get gaps in between roads that are not large enough for a block, because it's not big enough to extrude a sidewalk all the way around it.

These are referred to as Gaps and you can control what happens to these separately for some really nice effects.



Gaps can have their own markings and you can even extrude meshes or clone prefabs around them, such as guard rails or traffic cones.

They are textured according to the **Gap Surface** type you assign to them and an outline is drawn around them using the **Gap Outline Material**, effectively indicating the edge of the road. In addition the **Gap Fill Material** allows you to put hatched markings within the gap itself to better indicate a “no go area”.



Note that CityGen3D is not currently able to determine a suitable orientation for the Gap Fill texturing automatically based on road direction. So you may want to substitute RM StripesVertical with RM StripesHorizontal on a case by case basis by dragging it onto the mesh manually for now, where needed.

Gap Features are one of two types.

A **Clone** is just an instantiated prefab spawned at specified **Interval** around the gap outline.

Extrude can be used to repeat a mesh around the gap outline.

Both are used in the demo setup to produce a guard rail.



The **Range** setting provides a simple way of including or excluding gaps from additional geometry based on their size.

The default settings are setup so that smaller gaps do not have guard rails, but longer ones will.

In the above screenshot we can see how the gap between the slip road and the freeway is not deemed to be large enough to extrude a guard rail, so only the Gap Markings are applied to it.

But the long central reservation between the two motorways in the following screenshot falls within the range to trigger the extrusion of the guard rail mesh and cloning of the support beams.





This manual is work-in-progress, so not all panels are covered yet and there's lots more to explore. But you should now be able to use CityGen3D to download map data and use it to auto generate an environment in Unity, applying what you've learnt so far to other areas of the Generator.

Please experiment with the other Generator panels to create roadside objects (including houses) and trees, remembering to select a Landscape before clicking action buttons.

Once you have got to grips with the basics, you can go on to read more about advanced features such as Runtime scene streaming and origin shifting.



9. Runtime

In a multi-terrain environment you'll want to consider scene streaming so that nearby terrains are loaded at run-time when they are close to the camera, and unloaded when the camera moves far away from them.

If you have saved your landscapes as individual scenes (see [Heightmap - Terrains](#)) then CityGen3D is able to manage this process for you using the Landscape Manager at runtime.

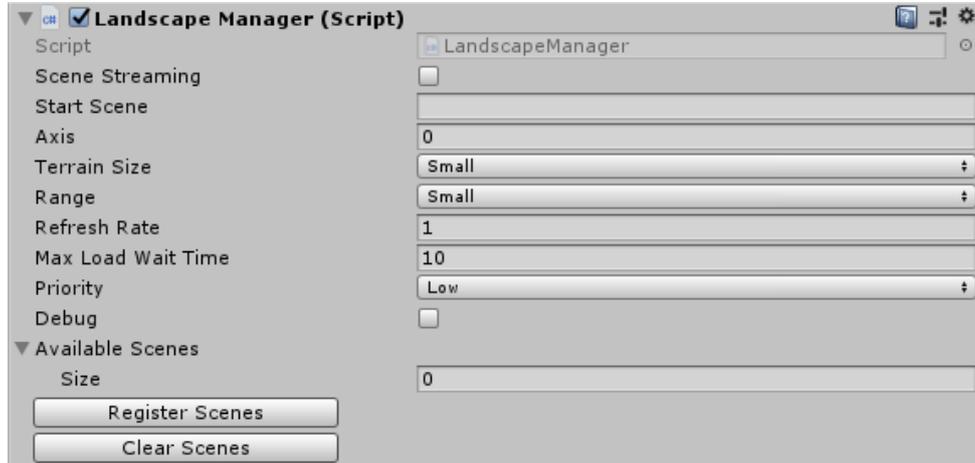


The performance of asynchronous loading and unloading of scenes by Unity is vastly improved in final builds when compared to running in Play mode in the Unity Editor.

9.1 Runtime - Landscape Manager

When you create a new environment in CityGen3D you have a main scene that has your primary objects in such as the camera, the map data, the Generator, and the Landscape Manager attached to the Environment prefab.

Let's take a look at the Landscape Manager in more detail by clicking the Environment prefab in your main scene and looking at the Inspector:

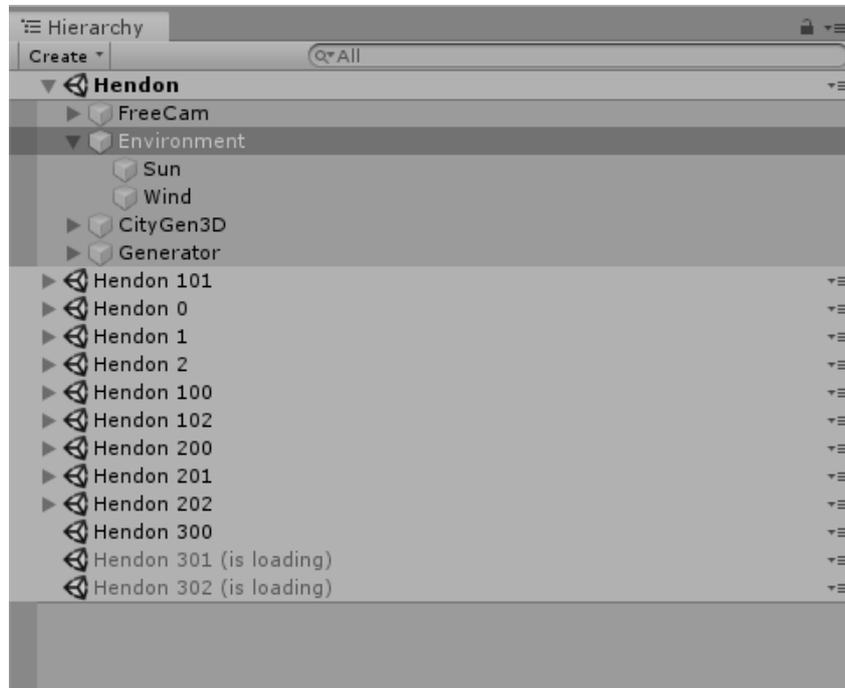


You can think of a Landscape as a terrain tile. Each one has a terrain component and acts as the parent object of all other objects above it. This makes it very easy for Unity to enable and disable different tiles as the user moves around the environment.

To tell the Landscape Manager to manage the loading and unloading of these terrain tiles at runtime, tick the **Scene Streaming** checkbox.

Enter the name of your main scene in the **Start Scene** field. CityGen3D will use this to identify which scenes to load.

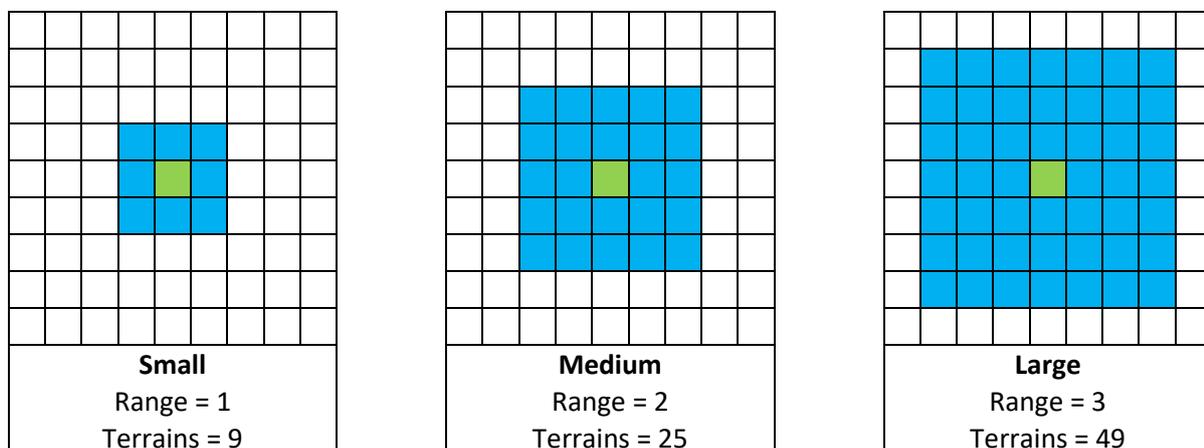
In this example you can see that we downloaded data for Hendon, which became the name of our start scene. All Landscapes created from this data were saved as separate scenes and share this name, but also have a numeric identifier to indicate which terrain tile the Landscape is for.



The **Axis** setting needs to be the same value as specified when you downloaded the data. The default is 100. Likewise, you should set your **Terrain Size** to match the size of your terrains. It is important these two values are set according to the specifics in your environment and are the same as what you chose when you downloaded your data.

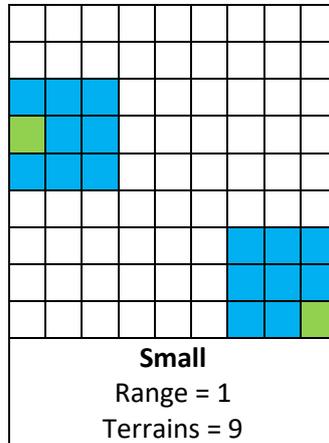
The Landscape Manager uses asynchronous additive scene loading and unloading to keep the landscapes around the camera loaded, and to unload those that are far away.

The bigger the selected **Range**, the more landscapes are retained in memory, in a grid layout around the landscape below the camera.



Note that this range respects world boundaries, which are determined by the Axis setting. This keeps the same amount of terrains loaded even when on the edge of the environment.

So with an Axis of 9 (81 terrains) and a Small range, the following terrains would be loaded if the camera was above the terrain marked in green in these two separate examples:



Because the Landscape Manager can identify which scenes to load based on location, it doesn't need to use colliders to trigger scene loading and unloading. Instead it will periodically check to see if the landscape below the camera has changed. How often this is done is controlled by the **Refresh Rate** setting, which is specified in seconds and defaults to one.

The **Max Load Wait Time** setting provides a failsafe, also specified in seconds, for the maximum length of time Unity should wait for an asynchronous scene load to finish.

By default, the Landscape Manager will attempt to load scenes in the background with as little impact on performance as possible at the expense of speed. The **Priority** setting allows you to have some control over this by allocating more time per frame to asynchronous tasks, which can result in faster loading at the expense of frame rate.

As is the case in any Unity project, you will need to add all your scenes to the Build Settings, which tells Unity which scenes to include in your build. This should include all Landscape scenes you create from your downloaded data that you want CityGen3D to be able to load at run-time.

You also need to tell the Landscape Manager which scenes are available to load by adding the name of each one to its **Available Scenes** list. Note that you don't need to add all your project's scenes to this list, just those created when you Generate Terrains. The order they are added is unimportant.

There are two buttons provided to make this process really easy, so you don't have to write the name of each one yourself:



Click to add all available Landscape scenes from the Build Settings into the Available Scenes list.



Removes all references in the Available Scenes list, making it empty and resetting it to zero.

9.2 Runtime - Origin Manager

Even with the aid of asynchronous loading of the environment, implementing very large 3D worlds poses certain challenges.

One of these challenges is to find a way of preventing spatial jittering due to limited floating-point precision.

This is where graphics appear to jump around slightly the further away from the world space origin (0,0,0) their transforms are.

This occurs because a computer can't represent numbers with infinite precision. There is a distinct set of numbers, which are not distributed evenly, and the gap between each number increases with its size and distance from the origin.

This lack of precision as an object moves away from the origin manifests itself in this jittering effect.

CityGen3D attempts to solve this using the Origin Manager, which periodically moves everything in the active scene so the camera is back at the origin.

This happens seamlessly because everything moves by the same amount and all objects retain their positions relative to each other, but it keeps your active environment close to the origin where there is high precision.

To setup the Origin Manager, simply assign your camera to the **Active Camera** field in the Inspector. If you don't do this, CityGen3D will attempt to find it for you at run-time.

The maximum distance the camera is allowed to move relative to the origin before this reset occurs is determined by the **Threshold** field. As soon as this is exceeded, the mass translation occurs.

The **Physics Threshold** setting specifies a distance at which physics is disabled. A value of zero prevents this and allows physics to be enabled for all objects.



Remember that the Origin Manager needs to be able to move all your objects around in the scene, so you need to ensure your objects are not marked as Static in the Inspector.

If you do not want CityGen3D to manage the origin in your project simply remove the Origin Manager component from your Environment object.

