Onward Custom Content Guide

Feb 2024 - Patch 1.14 - Guide/Package Version: 14 - SDK Increment 14

Disclaimer

- All information in this guide is up to date as of **Onward 1.14.0**, February 1st, 2024 and is subject to change. Changes and updates will be noted in the 'Change log'' section below.
- Updates to the custom content package files may cause inconsistencies with this guide
- If you encounter any issues with this guide or the custom content package files, please contact us using the <u>Onward Discord channel</u>.

Changelog

- 1/18/2023 Version 9.3 released. Changed upload method to upload more successfully more often. Fixed incorrect custom content link on page 7 and 10. Added Table of Contents for easier browsing, especially of new sections.
- 6/27/2023 Version 11.0 released. Fixed the ammo boxes Upgraded unity to 2019.4.40f1. Guide rewritten to improve readability and clarity.
- 8/15/2023 Version 12.0 released. Upgraded unity to 2021.3.23f1. Fixed the box gizmo matrices. Added Fire Fight support. Updated Lightbake guide. Updated set up for the new Unity version. Updated Quest guidelines now that Quest 1 is not supported. Added custom tent visuals section. Removed Max LOD on Quest (PC and Quest are equal now). Reduced the far clip plane from 2000 down to 1000. Usage of the built in skybox is now recommended for Quest maps. Added troubleshooting section (it will grow over time).
- 8/31/2023 version 12.1 released. Added far clip plane settings. Added new skybox shaders.
- 10/25/2023 version 13 released. New AI workflows. Hunt and Evac are no longer tied together. Night vision LUTs. Cleaned up old/out of date error checks. Shader clean up pass: replaced the old HLSL PreProcessing with the SG one. Replaced the skybox shaders with one, deleted the others. Updated all other shaders, added per texture tiling and offset for multiple maps.
- 11/16/2023 version 13.1 released. Improved saving and loading of AI entities.
- 1/31/2024 Version 14 released. Updated shaders.

Requirements

- Unity 2021.3.23f1 & 60-100GB free storage
- Onward Custom Content Package
- An Oculus Account (<u>Publishing</u>)

Table of Contents

Onward Custom Content Guide	1
Disclaimer	1
Changelog	1
Requirements	1
Table of Contents	2
Content submission rules	4
Technical Guidelines	4
Content Guidelines	4
Setting up Unity	6
Starting your first project	9
Updating an existing project	9
Creating a custom map	9
Setting up game modes	13
Set objectives & spawn points	14
Setting up the Escort game mode	17
Setting up the Assault game mode	
Setting up Fire Fight	19
Setting up the Co-op game modes (Hunt and Evac)	22
Additional navigation node types	26
AI Spawners	27
Cover Manager	28
Navigation Mesh	29
Setting up the Social game modes (OITC, Gun Game, Spec Ops)	32
Gun Game & One In the Chamber	32
Spec Ops	
Additional gameplay components	34
Set Physic Material	34
Elevators	35
Ammo Boxes	
PostProcessing Profile	37
OnwardAudioSource	
AreaReverb	39
Check For Errors	
Restricted Components	40
Testing your maps	42

PC Testing	
Quest testing	43
Performance Considerations	45
PC performance considerations	45
Quest performance considerations	45
Best practices in development	47
Publishing	
Conclusion	
Appendix A: Updating your project to work with Onward Workshop (update 1.9 onwards)	52
Appendix B: Claiming map ownership (update 1.9 onwards)	53
Appendix C: Optimization tools & tricks for Quest	56
OVRMetricsTool	56
Optimize Window	56

Content submission rules

Below are technical and content guidelines your maps must adhere to. Not following these guidelines will result in your upload being denied. In addition to these rules, please review the <u>Community</u> <u>Guidelines</u>. You are responsible for the testing and verification of your own map content submissions as well as curating an acceptable user experience.

Submission of custom content for play in Onward is a free service we offer to enrich the experience for players, it is not a right. Please follow these rules and you will be on your way to share your creations with thousands of Onward players. Your content/maps may be removed from our servers at any time post-approval if Downpour Interactive deems it has violated the rules for content submission.

By submitting your map(s) to the Onward Workshop, you agree to the <u>Custom Content End User</u> <u>Agreement</u> terms.

Technical Guidelines

- Your map materials must use only the PreProcessing (prefixed; PP_xyz..) shaders included with the Onward Custom Content SDK v1.12.0+. If your map currently does not contain only these shaders, that's alright. You can use our Shader Conversion steps and tools to assist you in converting your shaders. It is a simple process that can be carried out in a few minutes to an hour, depending on the complexity of your custom content scene.
- You must lightmap your map. Use <u>this lightmapping document</u> as a guide when you are baking lightmaps in your map.
- Your map must support the FreeRoam game mode and at least one (1) other game mode.
- Far clip plane now has a max of 1000m. (down from 2000)
- PC and Quest share LODs now. Before Quest would strip LOD0 and use LOD1. Now Quest uses LOD0.
- The built in Unity sky box is recommended for both PC and Quest. You no longer need to use the old sky box mesh on Quest.

Content Guidelines

• You may not use any copyrighted / unlicensed assets in your map. For example, you may not directly port de_dust2, or Blood Gulch etc.). Inspired homages of maps from other games and universes must be carried out without infringing source material.

- You must refrain from using terminology in your map descriptions that likens or compares it to a 1:1 copy of an existing map in another game or universe.
 - "A recreation of..." is not allowed.
 - "Inspired by..." is allowed.
- You may not place vulgar, inappropriate, racist, obscene or otherwise unpalatable or illicit content in your map.
- You may not use the full name, likeness or representation of any person (alive or dead) in your scenes.
- Do not put any content in your map that could be deemed hazardous or otherwise induce bodily harm to a player. This includes but is not limited to: extremely loud sounds, flashing/strobing visual aspects, nausea-inducing mechanics or otherwise hazardous elements.

Setting up Unity

Ensure that you have enough disk space to store the custom project and associated data generated during map creation (>60GB)

Download and install Unity 2021.3.23f1

Go to <u>the Unity website</u>, Select Unity 2021.x and locate and download Unity 2021.3.23f1 with Unity Hub, install Android Build Support when installing.

Add modules for Unity 2021.3.23f1 LTS		×
Add modules	Required: 0 byte	es Available: 159.18 GB
- DEV TOOLS	DOWNLOAD SIZE	SIZE ON DISK
Microsoft Visual Studio Community 2019	Installed	1.24 GB
▼ PLATFORMS	DOWNLOAD SIZE	SIZE ON DISK
Android Build Support	Installed	1.88 GB
└─ OpenJDK	Installed	67.2 MB
└─ Android SDK & NDK Tools	Installed	165.94 MB
iOS Build Support	413.3 MB	1.81 GB
tvOS Build Support	409.01 MB	1.79 GB
Linux Build Support (IL2CPP)	54.49 MB	225.69 MB
Linux Build Support (Mono)	54.5 MB	225.22 MB
Linux Dedicated Server Build Support	102.6 MB	417.14 MB
Mac Build Support (Mono)	328.11 MB	1.79 GB
Mac Dedicated Server Build Support	326.36 MB	1.78 GB
Universal Windows Platform Build Support	280.48 MB	1.97 GB
WebGL Build Support	330.03 MB	1.62 GB
Windows Build Support (IL2CPP)	Installed	1.76 GB

Note: You can install these tools later if you installed Unity with the Hub option. If you did not install Unity with Hub, you will need to do so in order to continue. To add these packages Go to Installs, click the cog icon next to 2021.3.23f1, and then click Add modules. Add the modules shown above.

2021.3.23f1 LTS		3
C:\Program Files\Unity\Hub\Editor\2021.3.23f1\Editor\Unity.exe	Add modules	
Android Windows	Show in Explorer	

Make a new project. Select the **3D Sample Scene (URP)**. This will take care of installing URP to our project for us, which is the pipeline used by Onward.



Check that it installed URP Version 12.1.11. You can check the version and change it if need through the package manager.

Package Manager		: 🗆 🛪
+ ▼ Packages: Unity Registry ▼ Sort: Name ↓ ▼ Mathematics	1.2.6 🗸 👗	
ML Agents		Universal RP Refease
		Unity Technologies
Mobile Notifications		Unity recurriculates View Constraints and Cons
▶ Multiplay		Registry Unity
▶ Multiplayer Tools		
Netcode for GameObjects		
▷ Oculus XR Plugin		The Universal Render Pipeline (URP) is a prebuilt Scriptable Render Pipeline, made by Unity. URP provides artist-friendly workflows that let you quickly and easily create optimized graphics across a range of
▶ OpenXR Plugin		
Polybrush		▶ Samples
Post Processing		
▶ ProBuilder		
Profile Analyzer		
▶ Recorder		
▶ Relay		
▶ Remote Config		
Scriptable Build Pipeline		
▶ Sequences		
▶ Shader Graph		
Sysroot Base		
Sysroot Linux x64		
► Terrain Tools		
Test Framework		
▶ TextMeshPro		
▶ Timeline	1.6.4 🔂	
▶ Toolchain Linux x64		
Foolchain MacOS Linux x64		
▶ Toolchain Win Linux x64		
Tutorial Authoring Tools		
Tutorial Framework		
Unity Distribution Portal		
▶ Unity Http	2.2.1 Git 🗸	
▶ Unity Profiling Core API		
▶ Unity Transport		
► Unity UI		
Universal RP	12.1.11 🗸	
▶ User Generated Content Bridge		
► Version Control	2.0.3 🕤	
▷ Visual Effect Graph		
▶ Visual Scripting		
Visual Studio Code Editor	1.2.5 🗸	

Download and add the custom content package to your Assets folder (drag and drop the package into your Unity window to import it)

Now, Open your **Project Settings** window under **Edit > Project Settings** and navigate to the **Graphics** tab. Once there, locate the **UniversalRenderPipelineAsset** file located in the root folder of **Assets > OnwardCustomContent** and drag it into the **Scriptable Render Pipeline Settings** field of the Graphics tab in the project settings window.

Assets > OwwardCustomContent Audo/a Graphics Editor Scriptable Render Pipeline Asset Scriptable Render Pipeline Asset (UniversalRender/Pipeline Asset (UniversalR	Assets > OnwardCustomContent Audio Editor Elitional Graphics Professor Professor Changelog ChineseRenderPipelineAsset City UniversalRenderPipelineAsset_Renderer City UniversalRenderPipelineAsset_Renderer City UniversalRenderPipelineAsset_Renderer City UniversalRenderPipelineAsset_Renderer	Craphics Scriptable Render Pipeline Settings Of UniversalRenderPipelineAsset
Initianal Editor Oraphics Oraphics In UTs Oraphics In Prefabs Physics 2D Physics 2D Physics 2D Player Preset Manager OG, UniversalRenderPipelineAsset_Renderer Scriptable Render Pipeline in use, some settings will not be used and are hidden Of, UniversalRenderPipelineAsset_Renderer Pareet Manager Og, UniversalRenderPipelineAsset_Renderer Scriptable Render Pipeline Asset Og, UniversalRenderPipelineAsset_Renderer Scriptable Render Pipeline Asset Og, UniversalRenderPipelineAsset_Renderer Pareet Manager Data Nures Scriptable Render Striptable Render Striptable Scriptable Render PipelineAsset_Renderer Scriptable Render Pipeline Asset Og UniversalRenderPipelineAsset_Renderer Scriptable Render Striptable Render Striptable Scriptable Render PipelineAsset_Renderer Scriptable Render Striptable Render Striptable Render Striptable Scriptable Render PipelineAsset_Renderer Scriptable Render Striptable Render Striptable Render Striptable VEX Scriptable Render Striptable Render Striptable	Internal Editor Graphies InvertManager Prefabs UT3 Prefabs Prefabs	Scriptable Render Pipeline Settings O UniversalRenderPipelineAsset(UniversalRenderPipelineAsset)
Element 3 Hidden/JubeBlend Element 4 Bryntes/Default Element 5 BUI/Default Element 6 BUI/Default/EIC1 Element 7 Biidden/VideoComposite		Built-in Shader Settings V Always Included Shaders Size 9 Element 0 B Legacy Shaders/Diffuse Element 1 B Hidden/CubeBlur Element 2 B Hidden/CubeCopy Element 3 B Hidden/CubeClend

Go to **Edit > Project Settings > Graphics tab** scroll down to the **"Shader Stripping"** section. Ensure all checkboxes are checked and both fields for **Lightmap Modes** and **Fog Mode**s are set to **'Custom'**. The screenshot below shows the correct settings.

OProject Settings			• <u>*</u>
Audio	Graphics		🛐 🕂 🗘,
Editor Graphics Input Physics D Physes 2D Player Preset Manager Quality Script Execution Order Tags and Layers TextMesh Pro Time	Graphics Reflection Probes Ble Detail Normal Map Enable Semitranspar Enable Light Probe Pr Cascaded Shadows Prefer 32 bit shadow Use HDR HDR Mode Rendering Path Readime Global Illum		÷
VFX			
	Built-in Shader Settin Deferrad Deferrad Screen Space Shadows Depth Normals Motion Vectors Light Halo Lens Flare P Always Included Shade Shader Stripping Lightmap Modes Baked Nor-Directional Baked Directional Baked Directional Baked Subtractive Fog Modes Linear Exponential Squared	Builterin abader Builterin abader Builte	
		(Keep All	÷
	Shader Preloading		

Then, navigate to the **XR Plugin Management** tab. Click the button to **Install the XR PLUGIN Management** Expand.

🌣 Project Settings	:=:
	<u>(a</u>
Adaptive Performance Audio	XR Plugin Management
Editor Graphics	1 order to use the new XR Plugin system you need to install the XR Plugin Management package. Clicking the button below will install the latest XR Plugin Management package and allow you to
Input Manager	Install XR Plugin Management
Memory Settings	
Package Manager Physics	
Physics 2D	
Player	
Preset Manager	
Quality Scene Template	
Script Execution Order	
Tags and Layers TextMesh Pro	
Time	
UI Builder	
Version Control	
XR Plugin Management	

Once the installation is complete, select PC and select **Oculus**. It will go through another small installation step.

🌣 Project Settings					: 🖬 ×
Project Sattings Adaptive Performance Audio Gator Gator	XR Plug-in Man			•	•
• Jorden Cloud Diagnostics Cloud Diagnostics Collaborate Tassanci Layvers Tasshean Pro Tassanci Layvers Tasshean Pro Time Tassanci Layvers Tasshean Pro Tassanci Layvers Tasshean Pro Tassanci Layvers Tasshean Pro Tassanci Layvers Tasshean Pro Tassanci Layvers Visual Scripting Visual Scripting Xeal Ugan Management Oculus	Information about config	tracking and migration c	n be found below.		

Now select the Android tab and again select Oculus.

C Project Settings				٩	:
Adaptive Performance Audio	XR Plug-in Man	agement			
Editor Graphics			æ		-
Input Manager Memory Settings		2			
Package Manager Physics	Plug-in Providers 🛛				
Physics 2D Player	ARCore Coulus				
Preset Manager Quality	Unity Mock HMD				
Scene Template Script Execution Order					
v Services					
Ads Cloud Build					
Cloud Diagnostics Collaborate					
In-App Purchasing Tags and Layers					
TextMesh Pro Time					
Timeline					
UI Builder Version Control					
Visual Scripting v XR Plug-in Management					
Oculus					

Next select Oculus on the left and ensure PC is set to Single Pass Instanced.

	🗘 Project Settings						: 🗆 ×
1							
	Adaptive Performance Audio	Oculus					
	Editor				王	#	
	Graphics Input Manager		Single Pass Inst				
	Memory Settings	Shared Depth Buffer		anceu			
	Package Manager	Dash Support	~				
	Physics Physics 2D						
	Physics 2D Player						
	Preset Manager						
	Quality Scene Template						
	Script Execution Order						
	v Services						
	Ads Cloud Build						
	Cloud Diagnostics						
	Collaborate						
	In-App Purchasing Tags and Layers						
	TextMesh Pro						
	Time						
	Timeline Ul Builder						
	Version Control						
	Visual Scripting						
	V XR Plug-in Management Oculus						
	Oculta						

Finally select the Android tab and make sure that the render mode is set to Multiview.

•••••					
Adaptive Performance	Oculus				
Audio Editor				æ	
Graphics Input Manager Memory Settings Package Manager Physics 20 Player Physics 20 Player Science Physics 20 Player Physics 20 Player Physics 20 Player Science Physics 20 Player Physics 20 Player Physics 20 Player Science Physics 20 Player Science Physics 20 Player Science Physics 20 Player Science Physics 20 Player Science Physics 20 Player Physics 20 Player Science Physics 20 Science	Steree Rendering Mode Low Overhead Mode (GLES) Optimize Buffer Discards (Vulkan) Phase Sync Symmetric Projecton (Vulkan) Subsamptied Layout (Vulkan) Foveated Rendering Method Enable TrackingOrigh Stage Mode Depth Submission (Vulkan)	Multiview	d Rendering		
Ads Cicul Build Cicul Diagnostics Collaboration In App Purchasing Taga and Layves Taga and Layves Taga the Pro Timeline	System Splash Screen Target Devices Quest Quest 2 Quest Pro ▶ Experimental	None (Texture	2D)		
u Tanana U Buuder U Buuder Visuul Scripting Visuul Script					

Ξ×

Starting your first project

Locate the unpacked **Custom Content Template Project** in your Assets and open **ExampleScene**. You can use this scene as an experimentation area and working example for custom content *game mode* setups in Onward.

► IMPORTANT

Rename your Example Scene and save it as a separate file. This will avoid any issues if the ExampleScene file is updated in the future.

► IMPORTANT

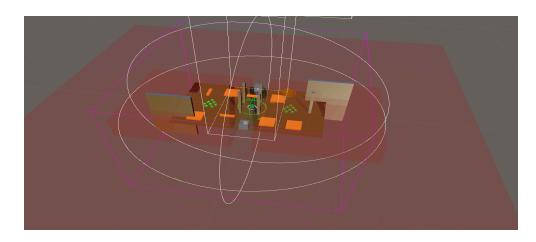
It is recommended to build your own map set-up by following the steps in this document so you have a full understanding of each component and how it is used. You can still use the Example Scene as a reference of what a valid set-up looks like.

Updating an existing project

- 1. Make sure to backup your project before updating.
- 2. Confirm that you have renamed any test scene and that it is no longer named **ExampleScene**.
- 3. Drag the new custom content package into your Assets folder inside the editor and accept the import of new and changed files.

Creating a custom map

After loading the ExampleScene, you will see a simple map, as shown in the screenshot below.

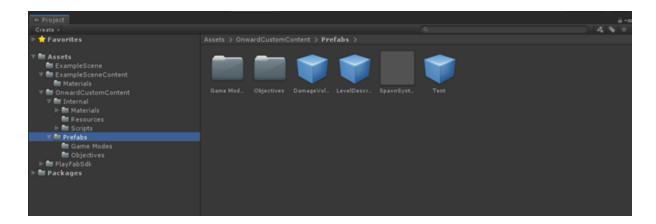


► IMPORTANT

Make sure you have already renamed this scene and saved it as a separate file. If you have not, do so now. This will avoid any issues if the **ExampleScene** file is updated in the future.

This scene is a valid map for all game modes, which can be used as a starting point to experiment with your own custom map.

All Unity prefabs for this project are located in Assets > OnwardCustomContent > Prefabs



► IMPORTANT

Do not modify the existing prefabs. **Changing these prefabs will cause errors** in your build. You may make modifications to the prefabs within the scene hierarchy instead.

Now, we will begin to set up these game objects in a blank scene (or your own pre-existing map) to show you how they work and to allow you to build scenes from scratch. You can also follow along and inspect the game objects in the Example Scene to gain an understanding of how they work.

You can always look to the original Example Scene for a reference on a working setup of a given game mode or feature.

To begin, every scene needs the following prefabs, you can find them in **OnwardCustomContent/Prefabs**:

- Tent
- MapQuad
- LevelDescriptor

Place these three Prefabs into your scene.

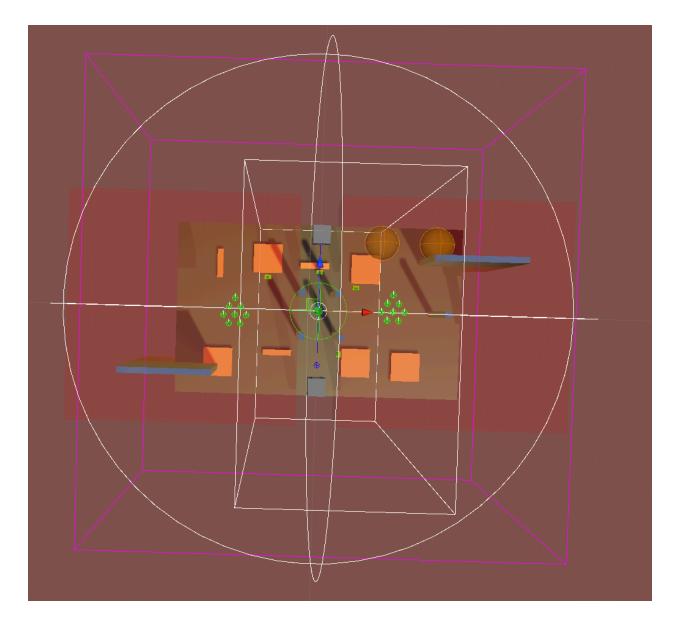
► IMPORTANT

Ensure the **Tent Prefab** is moved to be situated well below the play area of the map, or players may see

your level geometry through the tent's windows.

► IMPORTANT

Ensure the **MapQuad Prefab** is appropriately-sized over the map, encapsulating its playable-area within the magenta-pink wireframe box that is MapQuad's bounds. Use the size values in the inspector to shape the MapQuad to fit your map. Center the MapQuad over your map as shown in the picture below.

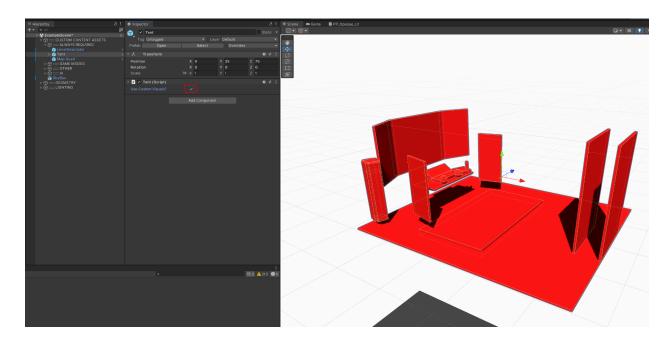


The LevelDescriptor's position does not matter, it will contain information about your map for later use.

Now that you have essential gameplay objects in your map and have set them up properly, you can move onto adding Game Modes to your map.

Custom Tent Visuals

By default the tent will be created in game for you at the tent position in your map. You can set it to not load visuals and build your own tent if you want. To do so check the **Use Custom Visuals** box. This will display a red framework of the tent to make it easier to place assets.



Setting up game modes

In order to support a game mode, the Prefab for that game mode must be added to your custom map scene. We will be using the Uplink game mode for this first example, but you can add more/other game modes by following the same steps. Note that some game modes have additional steps beyond Uplink, those will be detailed below this initial setup.

► IMPORTANT

Remember, every map is required to support the Free Roam game mode and at least one other mode.

► IMPORTANT

If you are starting from scratch with a blank scene, create some geometry to serve as the 'ground' in your scene.

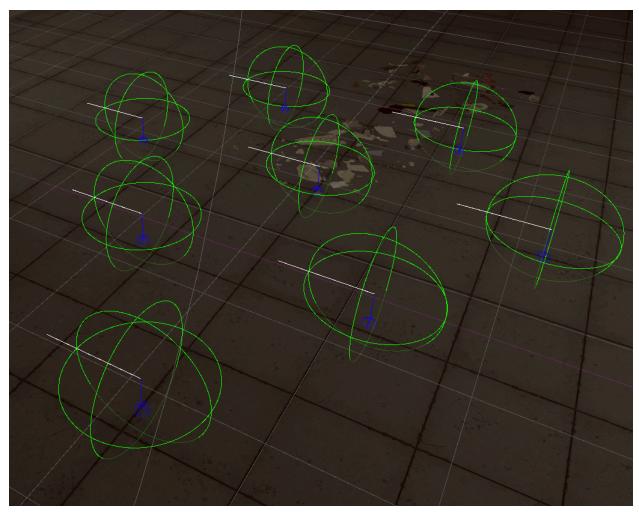
The **Example Scene** is designed to support all game modes and thus has pre-existing setups, but we will be replicating a new **Uplink** game mode in your own scene to start off.

To replicate its setup, begin by opening *Assets > OnwardCustomContent > Prefabs > Game Modes* and add **GM_Uplink** and the **GM_FreeRoam** prefabs to your scene. Game Mode prefabs will always start with '**GM_'**.

Once the game modes have been added, you will need to create *objectives* and *spawn points*. You can find the objective prefabs in *Assets > OnwardCustomContent > Prefabs > Objectives*. In this case, drag the **Obj_Uplink** prefab into your scene. (*Objective prefabs will always start with 'Obj_'*) Select the **Obj_Uplink** prefab in your scene hierarchy so you can see the information below in the inspector window.

		Obj_	Uplir	۱k				Sta	atic	•	
-	Tag	Unta	gged		+ Layer Default						
	Prefab (Ope		Se	lec	t (Ove	rride		•	
	L Tr	ansf	orm					2	규!	\$	
P	osition	ı X	0		Y	0	Z	Z 0			
R	otatio	ıХ	0		Y	0	Z	Z 0			
s	cale						Z	Z 1			
▼ .	ob	ject	ive	Upli	nk	(Scri	pi	2	규	\$	
					0					0	
)efendi)efendi				aw	'ns					
	Size		cun	q د 0							
Attacking Team ▼ Attacking Team Spawns Size 0											
	Add Component										

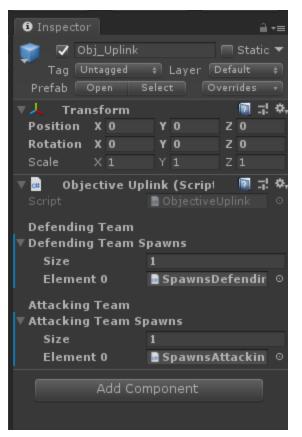
Now it's time to add spawn points. *Open Assets > OnwardCustomContent > Prefabs* and drag two SpawnSystem prefabs into your scene. Rename each prefab (F2, or through the inspector window) to SpawnsAttacking and SpawnsDefending. These are example names, you can rename them to whatever you wish.



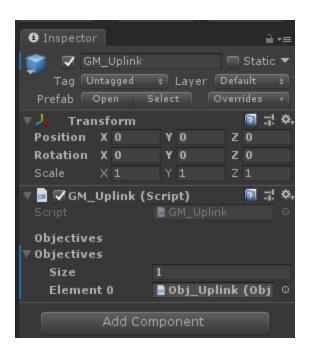
SpawnSystem prefab in scene.

Go back to the inspector window for the Obj_Uplink prefab and set the Defending and Attacking Team sizes fields to 1, and then drag and drop the SpawnSystem prefabs that you have named SpawnsAttacking and SpawnsDefending in their respective fields for attackers and defenders. Your settings on the Obj_Uplink should now look like this

Correct spawns (you can have more than 1 spawn for Attackers to vary the gameplay)



The last item in this step is to make sure that the **GM_ prefab** is referencing the matching **Obj_ prefab.** Select **GM_Uplink** from the hierarchy to open it in the inspector. Increase the Size of the Objectives field to **one (1)**. Drag the **Obj_Uplink** prefab into the field of the **Objectives** dropdown. Your settings should match the image to the right



If you want to create an Uplink game with a single objective, that's all you need to do! Your Uplink game mode will now work. Refer to **Section 7: Testing for testing information**. You can add more objectives and game modes using the steps above, just add more Obj_Uplink prefabs (that are set up with different spawns) to the GM_Uplink prefab.

Be aware that game modes can share spawn points, and you can also modify individual spawn points within the SpawnSystem object in your hierarchy.

We recommend that you place a **DamageVolume** prefab below the map and scale it to cover the entire footprint of your map. If a player somehow falls through the map, this will prevent them from falling endlessly. This has already been done in the ExampleScene, and you can use this as a reference for your own maps. You can find **DamageVolume** in *Assets > OnwardCustomContent > Prefabs*.

Setting up the Escort game mode

Escort is a game mode in which MARSOC has to escort 1 member of their team (The VIP) to one of two extraction points- marked with red smoke grenades. For this mode use **GM_Escort** and **Obj_Escort** prefabs. This mode allows you to define 2 different objective placements as dictated in the screenshot below:

Objective Escort (Script) Objective Escort Objective Escort Defending Team Objective Escort Objective Escort Defending Team Spawns Size Element 0 SpawnsDefending Team - SpecOpsDefender (SpawnSystem) Attacking Team SpawnsDefending Team - Anchor End - Down 1 (SpawnSystem) Size Element 0 SpawnsAttacking Team - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective A Spawn position for Objective B Objective Spawn Pos B EscortObj1 (Transform) Add Component 		
Defending Team > Defending Team Spawns Size Element 0 Attacking Team Spawns Size Element 0 Attacking Team Spawns Size Element 0 SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem) Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Objective Spawn Pos B		ObjectiveEscort
Defending Team Spawns Size Element 0 Attacking Team Attacking Team Spawns Size I Size Size I SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem) Attacking Team Attacking Team Spawns Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Spawn position for Objective B JescortObj2 (Transform)	Script	
Defending Team Spawns Size Element 0 Attacking Team Attacking Team Spawns Size I Size Size I SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem) Attacking Team Attacking Team Spawns Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Spawn position for Objective B JescortObj2 (Transform)		
Size 1 Element 0 SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem) Attacking Team * Attacking Team Spawns 1 Size 1 Element 0 SpawnsAttackingTeam - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective A * Objective Spawn Pos A * Spawn position for Objective B * Objective Spawn Pos B *		
Element 0 © SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem) Attacking Team * * Attacking Team Spawns 1 size 1 Element 0 © SpawnsAttackingTeam - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective A * Objective Spawn Pos A * Spawn position for Objective B * EscortObj1 (Transform) * Spawn position for Objective B * EscortObj2 (Transform) *	▼ Defending Team Spawns	
Attacking Team Attacking Team Spawns Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Objective Spawn Pos B Attacking Team - Anchor End - Down 1 (SpawnSystem) Attacking Team - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective B Objective Spawn Pos B	Size	
Attacking Team Spawns Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Objective Spawn Pos B	Element 0	🗟 SpawnsDefendingTeam - SpecOpsDefender (SpawnSystem)
Attacking Team Spawns Size Element 0 Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Objective Spawn Pos B	Attacking Team	
Size 1 Element 0 SpawnsAttackingTeam - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective A J.EscortObj1 (Transform) Spawn position for Objective B J.EscortObj2 (Transform)		
Element 0 © SpawnsAttackingTeam - Anchor End - Down 1 (SpawnSystem) Spawn position for Objective A J.EscortObj1 (Transform) Spawn position for Objective B J.EscortObj2 (Transform)	▼ Attacking Team Spawns	
Spawn position for Objective A Objective Spawn Pos A Spawn position for Objective B Objective Spawn Pos B	Size	
Objective Spawn Pos A <pre> LEscortObj1 (Transform) LescortObj2 (Transform) Dijective B Ubjective Spawn Pos B LescortObj2 (Transform) LescortD</pre>	Element 0	🗟 SpawnsAttackingTeam - Anchor End - Down 1 (SpawnSystem)
Objective Spawn Pos A <pre> LEscortObj1 (Transform) LescortObj2 (Transform) Dijective B Ubjective Spawn Pos B LescortObj2 (Transform) LescortD</pre>		
Spawn position for Objective B Objective Spawn Pos B JEscortObj2 (Transform)	Spawn position for Objective A	
Objective Spawn Pos B AEscortObj2 (Transform)	Objective Spawn Pos A	🙏 EscortObj1 (Transform)
Objective Spawn Pos B AEscortObj2 (Transform)		
	Spawn position for Objective B	
	Objective Spawn Pos B	↓Escort0bj2 (Transform)
		Add Component

When you place the **Obj_Escort** prefab you will have two objective children that are created that you must place in unique locations.

Take time to consider the implications of two objectives to reach and how that might influence your gameplay.

Setting up the Assault game mode

Assault is a game mode which has MARSOC and Volk fighting over an objective with respawns for both sides. For this mode use the **GM_Assault** and **Obj_Assault** prefabs. To prevent spawn camping, additional no-go gameplay zones can be defined using the **AssaultMapBoundary** prefab. Place and scale these prefabs to fit around zones the opposing teams should not be able to encroach on (such as player spawns), then associate them with the **Obj_Assault** prefab. You can associate just a single zone or multiple to each team's limitations (as is done in this screenshot).

🟮 Inspector 🖪 Occlusion 📑 Lighting 🔀 Navigation				≟ •≡
👕 🗸 Obj_Assault				
Tag Untagged	+ Laye	r Default		÷
Prefab Open		Overrides		
▼ 人 Transform				□ :
Position	X 76.9	Y -3.9	Z -15.2	
Rotation	X O	ΥO	Z 0	
Scale				
🔻 📾 Objective Assault (Script)				💽 🗐 🎲
Script				
Defending Team				
▼ Defending Team Spawns				
Size				
Element 0	📄 SpawnsDefendin	gTeam - SpecOpsDefende <mark>r (</mark> §	SpawnSystem)	0
Element 1	🗟 SpawnsDefendin	gTeam - SpecOpsDefender (S	SpawnSystem)	0
Attacking Team				
▼ Attacking Team Spawns				
Size				
Element 0	📓 SpawnsAttacking	Team - Anchor End - Down 1	(SpawnSystem)	0
Element 1	📄 SpawnsAttacking	Team - Anchor End - Down 1	(SpawnSystem)	0
Boundaries for defending team				
▼ Defending Team Boundaries				
Size				
Element 0	🗟 KillBoundary (As	saultMapBoundary)		0
Element 1	🗟 KillBoundary (As	saultMapBoundary)		0
Element 2	🖪 KillBoundary (As	saultMapBoundary)		0
Element 3	🖻 KillBoundary (As	saultMapBoundary)		0
Element 4	🗟 KillBoundary (As	saultMapBoundary)		0
Element 5	🗟 KillBoundary (As	saultMapBoundary)		0
Boundaries for attacking team				
▼ Attacking Team Boundaries				
Size				
Element 0	🗟 KillBoundary (As	saultMapBoundary)		
Element 1	🖻 KillBoundary (As	saultMapBoundary)		
Element 2	🗟 KillBoundary (As	saultMapBoundary)		

Place 2 SpawnSystem for each team, to allow them to choose between A and B spawn points in-game.

Boundaries for **ATTACKERS** will penalize **ATTACKERS** from entering the zone. Boundaries for **DEFENDERS** will penalize **DEFENDERS** from entering the zone.

You can also rename your **AssaultMapBoundary** prefabs to help distinguish them from each other.

Setting up Fire Fight

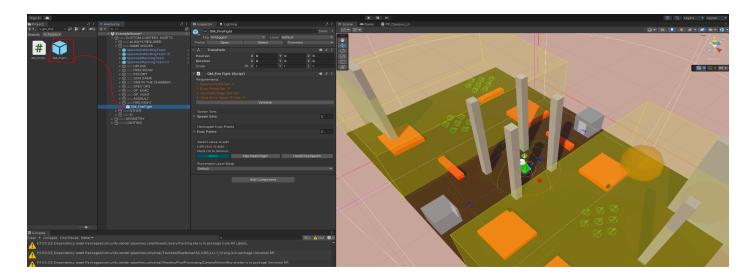
Fire Fight has each team split into two smaller fire teams. You can have the teams spawn anywhere you want. Best practices are to spawn the teams in positions where they won't see the other teams fire team for 15 or more seconds. The mode will randomly select from a spawn set, randomly select an evac point, and randomly select a hard drive spawn location. Try to add a lot to provide new experiences for the players.

Most official maps have: 10-14 or more spawn sets 6-8 evac points 50-150 hard drive spawns

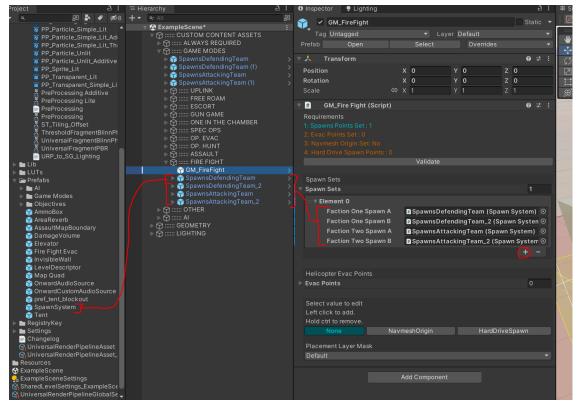
These are just what we found to work well, but feel free to play around and see what works for your map!

Step by step guide:

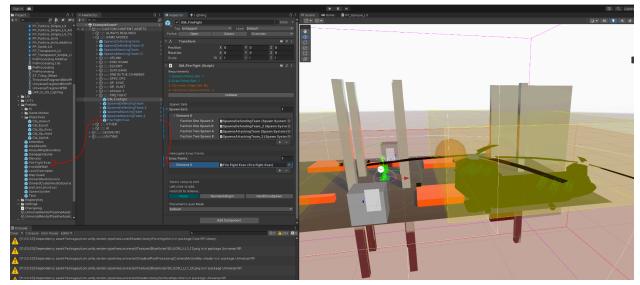
1. Add the GM_FireFight object to the scene.



2. Add a spawn set to the Spawn Sets list on the GM_FireFight game object. Add four SpawnSystem prefabs to the scene. Each faction needs two sets of spawns. Assign the spawn systems to the Faction spawns inside the spawn set. You can have as many spawn sets as you want.



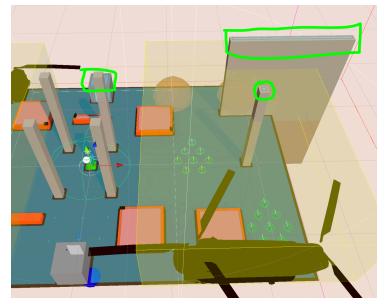
3. Add Evac points. Use the Fire Fight Evac prefab. Add the prefabs to the Evac Points list on the GM_FireFight prefab in the scene. Ensure the ramp is on the ground. You can add as many evac points as you want.



4. Fire fight requires a navigation mesh. If your map is already set up for the AI modes you should be good. If not, open the Navigation window (Window<AI<Navigation). Check the settings match these in the image, and click bake.

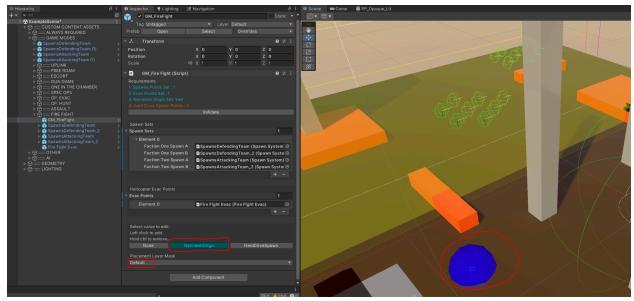
🛭 Inspector 🛛 🌻 Ligh	hting 🛛 🚼 Navigation	:
	Agents Areas Bake Object	
Baked Agent Size		
-	_R = 0.25	
		/
	H = 2	/
0.4	45	•
Agent Radius	0.25	
Agent Height		
Max Slope		45
Step Height	0.4	
Generated Off Mesh L		
Drop Height Jump Distance	0	
Advanced		
, navanosa		lear Bake
		2 2
To try an experim	ental, component-based workflow for configur	ing NavMeshes install the Al
Navigation packag		
Į		details, visit the Unity manual

NOTE: You do not need to worry about separated navmesh areas (areas in green circles). The Navmesh Origin in the next step takes care of them for you.

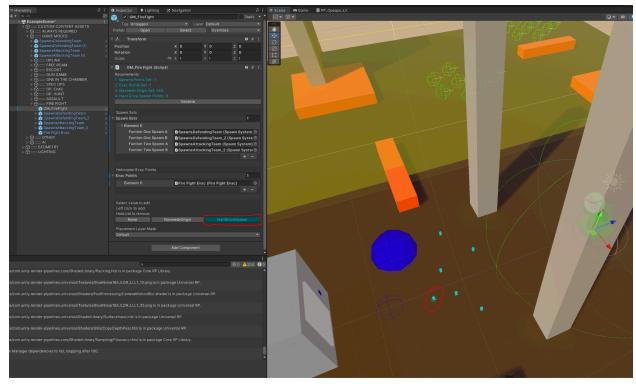


5. Next add a Navmesh Origin to the scene. Click NavmeshOrigin, check that the Placement Layer Masks are set up to something (Default is a good one to start with). Then just left click in the

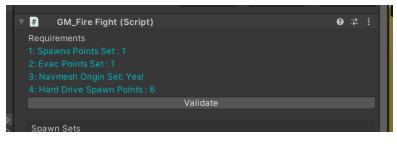
scene. You'll see a blue sphere in the scene.



6. Now add hard drive spawns to the scene. Click HardDriveSpawn, then click to place. Ctrl+click will remove the spawn. You need at least 1 but you can place as many as you want. Remember to place the spawns where a player could reach it.



7. Check that all of the requirements at the top are finished. They will be orange if you still need to complete a step.



Setting up the Co-op game modes (Hunt and Evac)

Onward has two co-op modes; Hunt and Evac.

Hunt spawns a player-determined number of bots into the level and the players goal is to eliminate them. Players can select between 4 and 32 bots. You can set up spawn points to decide where the bots spawn and waypoints for them to follow. Bots will spawn randomly from the spawn points you set up.

Evac requires the players to survive until a helicopter arrives to rescue them. Players spawn at the start of a round and then spawns bots over time. Lower difficulties spawn fewer bots, higher difficulty spawns more bots. Bots will seek out and attack the player. If the helicopter has arrived then bots have a chance to defend the evac point.

Co-op modes are set up using a few prefabs:

Al_Navigation - contains important navigation information

AI_WaypointSet - defines patrol paths for the bots

Al_Spawnpoint - determines where and what types of bots spawn

GM_Op_Hunt - contains all of the hunt objectives

Obj_Op_Hunt - defines one hunt objective including player spawns, bot spawns, and waypoint sets

GM_Op_Evac - contains all of the evac objectives

Obj_Op_Evac - defines one evac object including player spawns, bot spawns, and evac locations

Operations Evac Point - determines where and how and evac helicopter will spawn

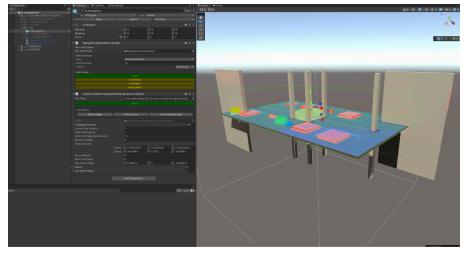
AI_Navigation Set Up

Navigation Mesh Baker

The navigation mesh baker is just a wrapper for Unity's normal nav mesh with a couple additional things like the ability to specify an area to bake. This must be done before moving to the Navigation Map Generator section.

- 1. Drag the AI_Navigation prefab into the scene.

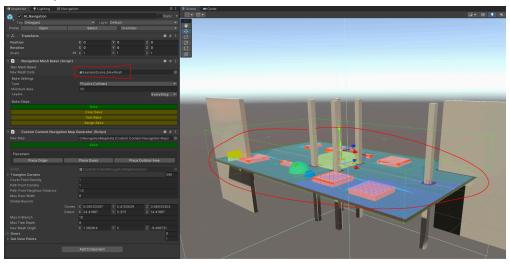
2. First use the bounds gizmos to cover the playable area of your level. The bounds are the green wire box and you can adjust the size using the small green squares at the center of each face.



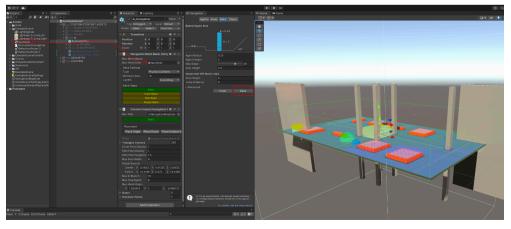
3. Click Bake

# Navigation Mesh Baker (Script)		0 ≓ :
Nav Mesh Baker		
Nav Mesh Data	2 ExampleScene_NavMesh	
Bake Settings		
Туре	Physics Colliders	
Minimum Area	10	
Layers	Į.	Everything 🔫
Bake Steps		
	Clear Bake	
	Null Bake	
	Assign Bake	
🛯 🗯 🔹 Custom Content Navigation Map G	enerator (Script)	0 ≓ :
Nav Map	କ୍ତNavigationMapData (Custom Content Naviga	ation Map) 💿

4. Make sure you can see a Nav Mesh Data object and the nav mesh in the scene (the blue stuff on the floor).



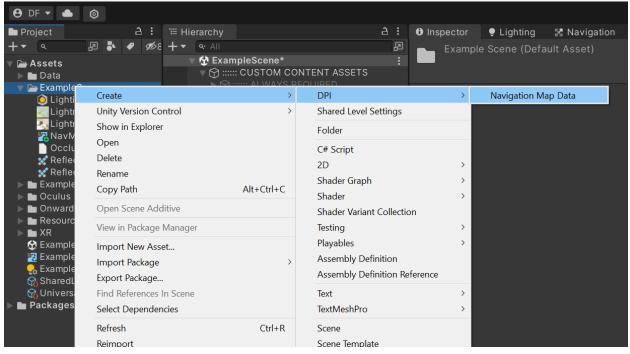
Alternatively if you need to customize the setting of the nav mesh, you can just use the built in baking system. You must then manually assign the Nav Mesh Data to the Navigation Mesh Baker.



Navigation Map Generator

The navigation map generator helps the bots make decisions during runtime. It's an improved version of the old nav node system.

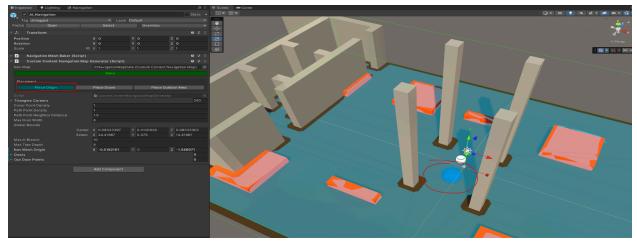
 Create a Navigation Map data object. Right click a folder, go to Create < DPI < Navigation Map Data.



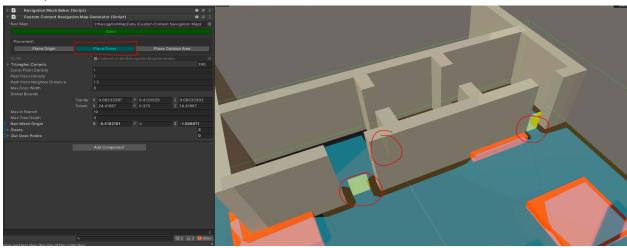
2. Name the data whatever you want and then drag it into the Nav Map Field in the Navigation Map Generator.

Project A : 🗃 Hierarchy		Inspector 🌻 Lighting 🛛 🔀 N	avigation		a :
+ ▼	P 😭	AI_Navigation			Static 🔻
► Assets C	JSTOM CONTENT ASSETS	Tag Untagged	- Layer Det	ault	
▼ ExampleScene		Prefab Open	Select	Overrides	•
lightingData Lightmap-0_comp_dir ▷ □ ······		👃 Transform			0 ‡ :
Lightmap-0_comp_light ▼ 🗘 ::::::		Position	x o y	0 Z 0	1
😪 Navigation MapData	I_Navigation >	Rotation		0 Z 0	
		Scale	48 X 1 Y		
	perations Evac Point >				
😴 ReflectionProbe-1 👘 🔍		# Navigation Mesh Baker (S			0 ∓ :
► ► ExampleSceneContent ▼ ♥ :::::: GE		# Custom Content Navigation	on Map Generator (Script)		0 ∓ :
			🗕 🔿 Navigation Map Data	(Custom Content Navigation	Map) 💿
► Resources					
KR Pilla Kr G FyampleScene G Wall					
		Placement			
ChampleSceneSettings		Place Origin	Place Doors	Place Outdoor Area	
	undPlane vard_Logo		CustomContentNavio		
ChiversalRenderPipelineGI₁ ConversalRenderPipelineGI₁ Packages Cov		Triangles Corners			240
		Cover Point Density	1		240
Q Cov		Path Point Density			
Cov M Cov		Path Point Density	1.5		
Cov		Max Door Width	8		
Č Cov		Global Bounds	0		

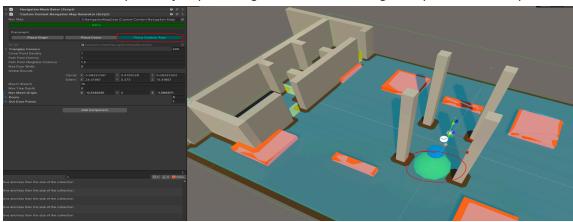
3. Next Select "Place Origin" and click a point on the navigation map. Try to place the point outside and as close to the center of the map as possible. You will see a blue sphere when it's placed.



4. Next select "Place Doors". Then left click to place a doorway. It's best to place doors in actual doorways but they can be used to divide up larger spaces. (if needed increase the "Max Door Width")



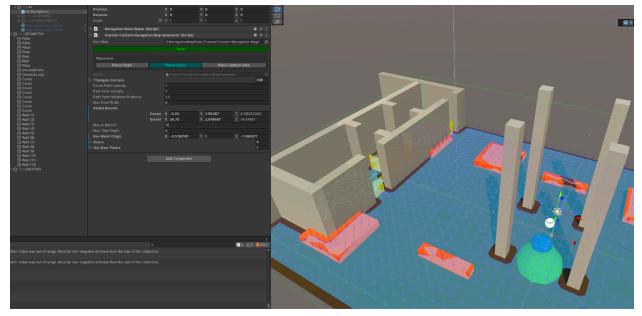
5. Next select "Place Outdoor Area". Left click to place an outdoor space. Even indoor only maps need an "Outdoor" space so just pick a large room. You'll see a green sphere when it's placed.



6. Finally click "Bake". Large maps can take a few minutes. Small maps a few seconds.

₹ #	🛛 # Custom Content Navigation Map Generator (Script) 🛛 🛛 😨 🥫								9 ∓±	:
Nav Мар				NavigationMapDa	ta I	Custom Content N	av	igation Ma	ap)	0
	Placement									
	Place Origin		Place Doors Place Outd							
				CustomContentNa						
	riangles Corners								240	
C	over Point Density		1							
Pa	ath Point Density									
	ath Point Neighbor Distance		1.5							
	ax Door Width		8							
GI	lobal Bounds									
		Center		0.08333397		0.4120029		0.08333		
м	ax In Branch	Extent	× 10	24.41667		0.375		14.41667		
	ax Tree Depth		8							
	av Mesh Origin		x	-0.5182161		0	z	-1.58697	71	
	oors								3	
► 0	ut Door Points								1	
			A	dd Component						

When the Bake is finished you will see a bunch of little lines on the map. Rooms will be different colors.

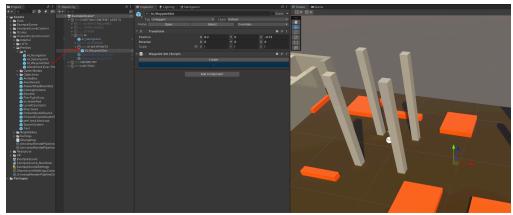


That's it for the AI navigation set up. Make sure you save your scene. If you are upgrading a map from a previous version of the SDK you can delete all the old prefabs.

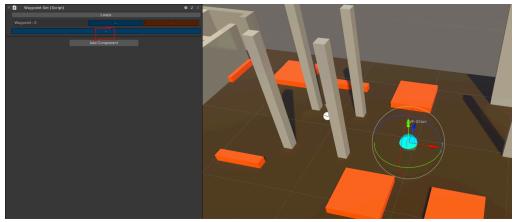
Waypoints

You can now define paths or waypoint sets for the AI to patrol along. Waypoints only affect Hunt. Waypoints can be used for just one objective or shared between objectives.

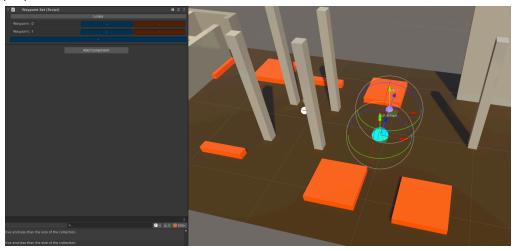
1. Drag the waypoint set prefab into the scene.



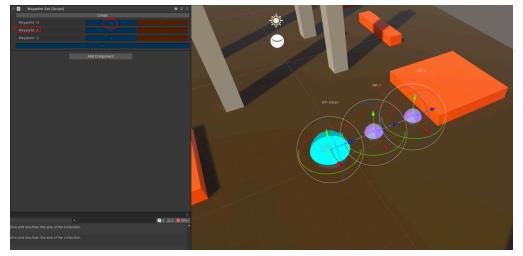
2. Click the "+" button to add a point. This first point is the starting point.



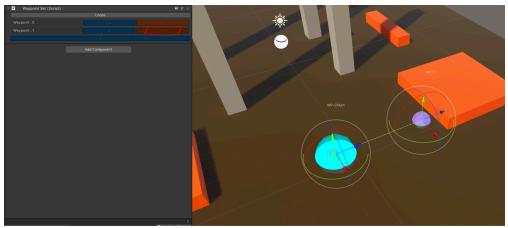
3. Press the "+" button again to add a second point. The start is light blue, the waypoints are purple.



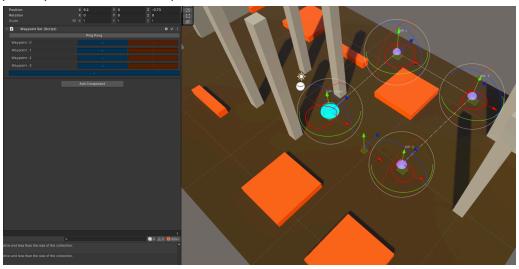
4. Clicking the "+" next to each way point will add a point in between that point and the next.



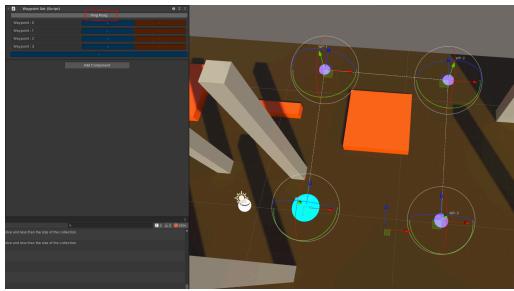
5. Clicking the "-" button deletes that point.



6. Use the transform gizmos to position the waypoints. Waypoints must be placed near a navmesh. There's a validation step during the build process that will snap them to nearby navmeshes for you so you don't need to be perfect, within a 5 meters is fine.

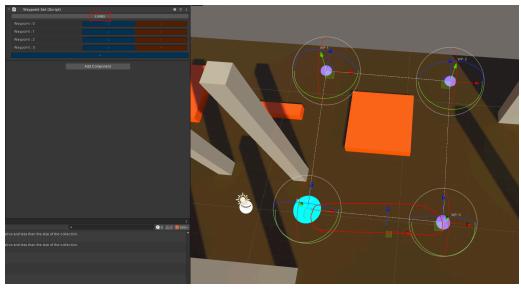


You can swap between a Looping or Ping Ponging path by clicking the button at the top. This will cause the bots to either; Ping Pong - reverse direction when reaching the start and end, Looping - go to the start point when reaching the end. You can see this in the path too.



Ping Pong

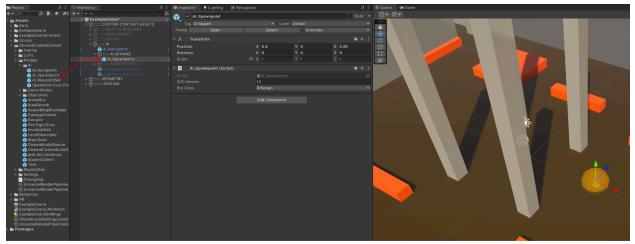
Looping



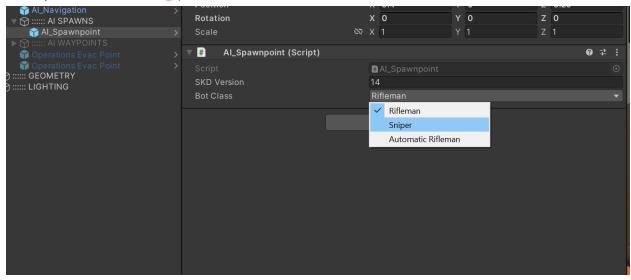
AI Spawners

Next, you will add some AI Spawners to the level.

 Drag the AI_Spawner prefab into the scene. Spawn Points must be placed near a navmesh. There's a validation step during the build process that will snap them to nearby navmeshes for you so you don't need to be perfect, within a 5 meters is fine.



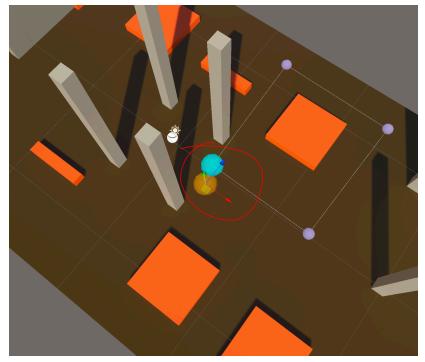
2. Here you can select from 3 options; Rifleman, Sniper, and Automatic Rifleman. (NOTE: Only the rifleman will spawn in the game for now. Other classes are coming in the future and we want to future proof the SDK a bit)



You need at least 32 spawn points in the level. Spawn points can be shared between Hunt and Evac, or you can define spawn points per objective. If you are upgrading from an older SDK the important thing to note is that the spawn points are individual now, the squad spawning is gone to provide more flexibility in bot placement.

To have a bot use a Waypoint set you just need to place the spawn point near the start point of the waypoint, within 5 meters should work. Note, the spawn points are randomly chosen so there may not always be a bot on a patrol path.

Bots not placed near a Waypoint set will use some default idle behaviors. They will mostly just wander around their spawn area. More behaviors will come in future updates.



Project 🗅 🗄 🗄		a: 0	Inspector 🌻 Lighting	🔀 Navigation			
+ - 🔍 🔊 🦆 🛷 🕫		2 4	> GM_Op_Hunt (Prefab A				0:
🔻 🚘 Assets	🔻 🔂 ExampleScene*	:					
🕞 🖿 Data							
ExampleScene	► 🛱 :::::: ALWAYS REQUIRE						heckout
ExampleSceneContent	V 🖓 :::::: GAME MODES						
▶ 🖿 Oculus	Figure 1 - SpawnsDefendingT Figure 1 - SpawnsDefendingT						
OnwardCustomContent Internal	► Spawnsberending T			III editing support)			
▶ ■ Internal ▶ ■ LUTs	► M SpawnsAttackingTe	eam (1)					
V Prefabs			GM_Op_Hunt				
▶ ■ AI			Tag Untagged		ayer Default		
🔻 🗁 Game Modes			3				heckout
😭 GM_Assault							neckout
😭 GM_Escort			🙏 Transform				0 ‡ ;
GM_FireFight			Position	x 0	Y O	z o	
☆ GM_FreeRoam ☆ GM_GunGame	V 🕅 :::::: OP. HUNT		Rotation	x o	YO	zo	
M_GunGame GM_OneInTheChamber	GM_Op_Hunt						
GM_Op_Evac	🖌 🕤 Obj_Op_Hunt						
GM_Op_Hunt	► 💬 :::::: ASSAULT		# GM_Operations_Hunt	(Script)			0 ‡ :
GM_SpecOps			Script				
GM_Uplink				GM_Operat			
V Dijectives	Al_Navigation			14			
☆ Obj_Assault ☆ Obj_Escort	► 🕅 :::::: AI SPAWNS		Objectives				
Pobj_Escont	► 💮 :::::: AI WAYPOINTS		Objectives				0
Obj_Op_Hunt							
Obj_Uplink							
😭 AmmoBox	► 💬 :::::: GEOMETRY						
😭 AreaReverb							
脊 AssaultMapBoundary 🈭 DamageVolume				Add Compon			
Pamagevolume							
😭 Fire Fight Evac							
invisibleWall							
LevelDescriptor							
ү Map Quad							
OnwardAudioSource							
OnwardCustomAudioSource							
😭 pref_tent_blockout 🈭 SpawnSystem							
Tent							
Registry Key							
Settings							
📄 Changelog							
G UniversalRenderPipelineAsset							
G UniversalRenderPipelineAsset							
Resources XR							
ExampleScene							
2 ExampleScene_NavMesh							
ExampleSceneSettings							
SharedLevelSettings ExampleSc							

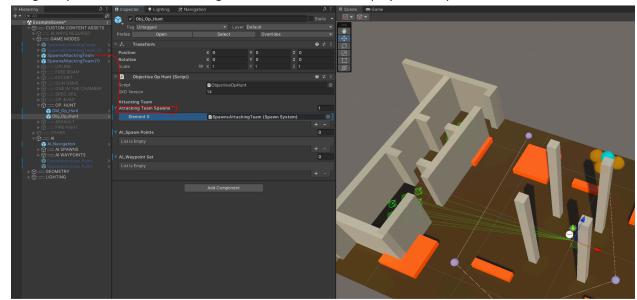
1. Add a GM_Op_Hunt prefab and a Obj_Op_Hunt prefab to the scene.

- 👌 : 🚺 Inspector 🌻 Lighting 🛛 🔀 Navigation # 5 😭 🗹 GM_Op_Hunt ✿ ExampleScene*
 ♥ ⊕ :::::: CUSTOM CONTENT ASSETS Tag Untagged * Open Select Overrides 0 1 :
 Image: SpawnsDefendingTeam

 Image: SpawnsAttackingTeam

 Image: # GM_Operations_Hunt (Script) Objectives Objectives GM_Op_Hunt Obj_Op_Hunt (Objective Op Hunt)
 [™] AI_Navigation
 P ⊕ :::::: GEOMETRY
 P ⊕ :::::: LIGHTING
- 2. Add the Obj_Op_Hunt prefab to the Objectives list on the GM_Op_Hunt object.

3. Assign a spawn set for the Attacking team. This is where the players will spawn.



4. Assign at least 32 AI spawn points to the AI_Spawn Points list.

	🗄 🛛 Inspector 🌻 Lighting 🛛 😤 Naviga		🖴 : 🗰 Scene 🛛 😎 Game
			9 ≠ :
▶ 👸 SpawnsAttackingTeam (1)			
		ObjectiveOpHunt	
	Attacking Team		Ø
	V Attacking Team Spawns		1
	Element 0	SpawnsAttackingTeam (Spawn System)	
▷ ③ :::::: OP. EVAC ▼ ④ :::::: OP. HUNT		a opanno (tradinig roam (opanni opatom)	+ -
GM_Op_Hunt	AL_Spawn Points		34
👔 Obj_Op_Hunt			
		Al_Spawnpoint (Al_Spawnpoint)	
	Element 1	AI_Spawnpoint (1) (AI_Spawnpoint)	
		Al_Spawnpoint (2) (Al_Spawnpoint)	
Al_Navigation	Element 3	AI_Spawnpoint (3) (AI_Spawnpoint)	
ALSPAWNS	Element 4	■AI_Spawnpoint (4) (AI_Spawnpoint)	
ALSpawnpoint (1)	Element 5	Al_Spawnpoint (5) (Al_Spawnpoint)	•
AL_Spawnpoint (2)	Element 6	AI_Spawnpoint (6) (AI_Spawnpoint)	
Al_Spawnpoint (3)		Al_Spawnpoint (7) (Al_Spawnpoint)	
AL_Spawnpoint (4)	Element 8	AI_Spawnpoint (8) (AI_Spawnpoint)	
ML_Spawnpoint (5)	Element 9	■AI_Spawnpoint (9) (AI_Spawnpoint)	
Al_Spawnpoint (7)	Element 10	Al_Spawnpoint (10) (Al_Spawnpoint)	
😚 Al_Spawnpoint (8)	Element 11	AI_Spawnpoint (11) (AI_Spawnpoint)	
Al_Spawnpoint (9)		Al_Spawnpoint (12) (Al_Spawnpoint)	
AL_Spawnpoint (10)	Element 13	AI_Spawnpoint (13) (AI_Spawnpoint)	
Al_Spawnpoint (12)	Element 14	AI_Spawnpoint (14) (AI_Spawnpoint)	
😭 Al_Spawnpoint (13)		Al_Spawnpoint (15) (Al_Spawnpoint)	
AL_Spawnpoint (14)	Element 16	AI_Spawnpoint (16) (AI_Spawnpoint)	
AL_Spawnpoint (15)	Element 17	Al_Spawnpoint (17) (Al_Spawnpoint)	
Al_Spawnpoint (17)	Element 18	AI_Spawnpoint (18) (AI_Spawnpoint)	
📸 Al_Spawnpoint (18)	Element 19	AI_Spawnpoint (19) (AI_Spawnpoint)	
AL_Spawnpoint (19)		Al_Spawnpoint (20) (Al_Spawnpoint)	
AL_Spawnpoint (20) AL_Spawnpoint (21)	Element 21	AI_Spawnpoint (21) (AI_Spawnpoint)	
AL_Spawnpoint (22)	Element 22	Al_Spawnpoint (22) (Al_Spawnpoint)	
Al_Spawnpoint (23)	Element 23	AI_Spawnpoint (23) (AI_Spawnpoint)	
AL_Spawnpoint (24)	Element 24	AI_Spawnpoint (24) (AI_Spawnpoint)	
ALSpawnpoint (25)	Element 25	Al_Spawnpoint (25) (Al_Spawnpoint)	
ALSpawnpoint (27)	Element 26	AI_Spawnpoint (26) (AI_Spawnpoint)	
Al_Spawnpoint (28)		AI_Spawnpoint (27) (AI_Spawnpoint)	
AL_Spawnpoint (29)	Element 28	AI_Spawnpoint (28) (AI_Spawnpoint)	
AL_Spawnpoint (30)	Element 29	AI_Spawnpoint (29) (AI_Spawnpoint)	
AL_Spawnpoint (32)	Element 30	Al_Spawnpoint (30) (Al_Spawnpoint)	
Al_Spawnpoint (33)	Element 31	AI_Spawnpoint (31) (AI_Spawnpoint)	
	Element 32	Al_Spawnpoint (32) (Al_Spawnpoint)	
Operations Evac Point Operations Evac Point	Element 33	AI_Spawnpoint (33) (AI_Spawnpoint)	
►			+ -
	▼ ▼ AI_Waypoint Set		0 -
		Q 4	🛦 0 🏮 999+
sception: Index was out of range. Must be non-r	enative and less than the size of the collection		·

5. Finally, assign the waypoints for the objective.

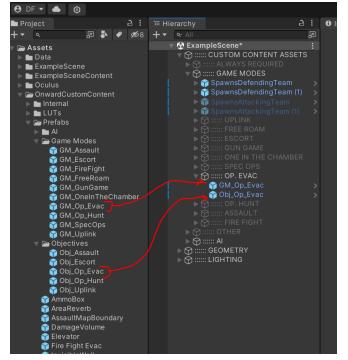
i≡ Hi		🛛 Inspector 🛛 🜻 Lighting 🛛 😤 Nav	/igation	≙ :	# Scene 🖙 Game
+-	🗣 All 🖉	☆ Obj_Op_Hunt			
	ExampleScene* ExampleScene* CONTENT ASSETS	Tag Untagged	 Layer Defaul 		
		Prefab Open		Overrides 🗸	
	V 💮 ::::: GAME MODES	V 🙏 Transform		0 # 1	- •
	 Image: SpawnsDefendingTeam Image: SpawnsDefendingTeam Image: SpawnsDefendingTeam 	Position	x o y o		
	SpawnsAttackingTeam	Position Rotation	X 0 Y 0 X 0 Y 0		
© 70	▶ m SpawnsAttackingTeam (1) >	Scale	δ X 1 Y 1	z 1	
		Bijective Op Hunt (Script)			
		Script SKD Version	ObjectiveOpHunt 14		
			14		
	▼ 😭 :::::: OP. HUNT	Attacking Team			
	GM_Op_Hunt >	Attacking Team Spawns AL_Spawn Points		1	
	Obj_Op_Hunt SSAULT SSAULT	▼ Al_Waypoint Set		2	
		Element 0	AI_WaypointSet (Wayp		
		Element 1	AL_WaypointSet (1) (W		
	Al_Navigation >				
	▷ ① :::::: AI SPAWNS □ ① :::::: AI WAYPOINTS				
	Al_WaypointSet >				
	Al_WaypointSet (1)				
	Operations Evac Point Operations Evac Point Operations Evac Point				
	F :::::: GEOMETRY				
e					
t t					

That's it. You now have one Hunt objective set up. You can have as many hunt objectives as you want. Just remember to add them to the GM_Op_Hunt objective list.

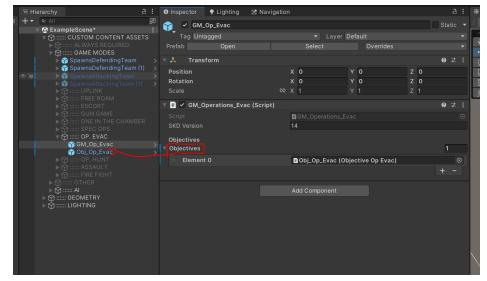
Now that Spawnpoints are not shared across all hunt spawns and you have the ability to place waypoints for each objective, feel free to mix and match AI spawn points and waypoints to come up with some basic scenarios for the players.

Evac

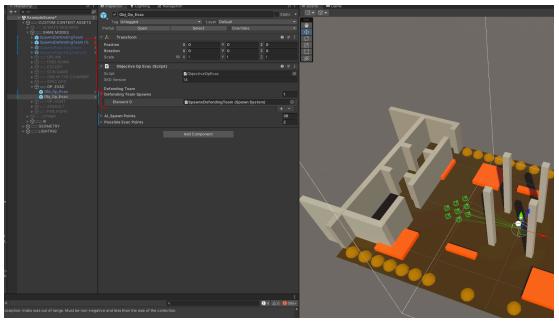
1. Add a GM_Op_Evac prefab and a Obj_Op_Evac prefab into the scene.



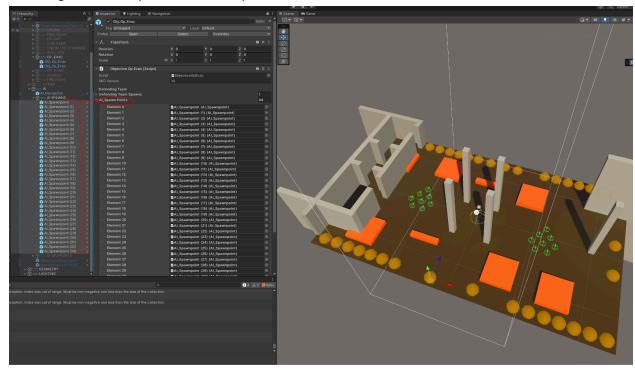
2. Add the Obj_Op_Evac item to the Objectives list in the GM_Op_Evac.



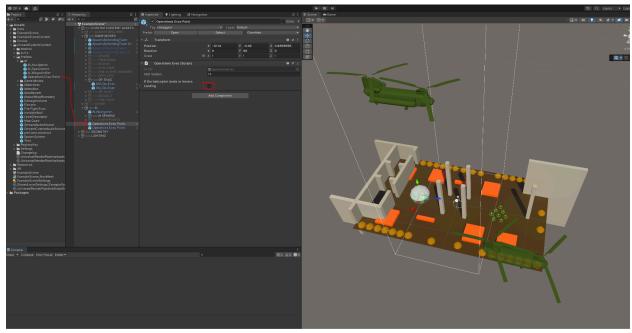
3. Add a Spawn System to the Defending Team Spawns on Obj_Op_Evac. This is where the players will spawn.



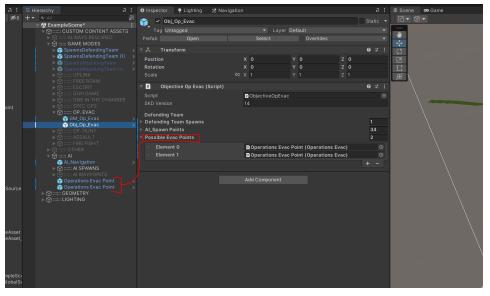
4. Add AI spawn points to the AI_SpawnPoints list. Best practice is to have them spawns on the outer edges of the map since the Bots spawn over time.



5. Add one or more Operations Evac Prefabs to the scene. You can have the helicopters land or hover.



6. Add the Operations Evac Prefabs to the Possible Evac Points list on the Objective. The Evac will be randomly selected so you can add as many as you want. You must have at least one.



That's it for Evac. Evac doesn't use waypoints, the bots will assault the players position, or defend the helicopter landing point. You can add additional Evac Objectives to provide different player spawns, bot spawns, and evac points. Objectives will be chosen randomly.

Setting up the Social game modes (OITC, Gun Game, Spec Ops)

In Gun Game, players must fight to be the first to achieve a kill with a list of weapons, receiving a new one each time they score a kill.

In One in the Chamber, players must fight to be the last one standing, their only weapon being a bolt-action rifle that contains a single round. You can find additional rounds throughout the game mode, but if you fail to kill on the first shot, you'll need to resort to melee tactics.

In Spec Ops, a *MARSOC* squad with flashlight-equipped pistols and limited ammo tries to survive a team of *Volk* assassins armed with combat knives and night vision until the MARSOC can escape after a set time period. A dense fog covers the battlefield, limiting MARSOC visibility. The Volk assassins have ditched all unnecessary equipment, greatly increasing their mobility and stealth. Ammo Boxes are scattered throughout the map- enticing each side with the tools to survive, or assassinate. The MARSOC will find extra magazines for their sidearms and the Volk will have a chance to find a flashbang that can give them the edge to close the gap. *Who will survive*?

► IMPORTANT

Social Modes are different from other **GM_** Prefabs because they do not require an **Obj_** Prefab to link into the **GM_** Prefab. They work just by adding a **GM_OneIntheChamber**, **GM_GunGame**, or **GM_SpecOps** Prefab into the scene and associating **SpawnSystem** Prefabs with them.

Gun Game & One In the Chamber

Both of these game modes use a single **SpawnSystem** Prefab that is associated into the respective **GM_GunGame** and **GM_OneInTheChamber** Prefabs. You can use the same **SpawnSystem** for both Game Modes, or individualize them.

Initially, the **SpawnSystem** will have eight spawn locations arranged next to each-other when you place this prefab. For these two *Free For All* modes, you will be moving each individual spawn to a unique location. If you desire more than eight spawn points, you can duplicate any of the **SpawnPoint** children under the **SpawnSystem** parent and continue to place them in new locations until you are satisfied with the amount of spawns.

Once your **SpawnSystem** is linked to the **GM_GunGame** and **GM_OneInTheChamber** prefabs and you have moved the individual *SpawnPoints* in the **SpawnSystem** to unique locations (and possibly added additional ones through duplication), your setup is complete!

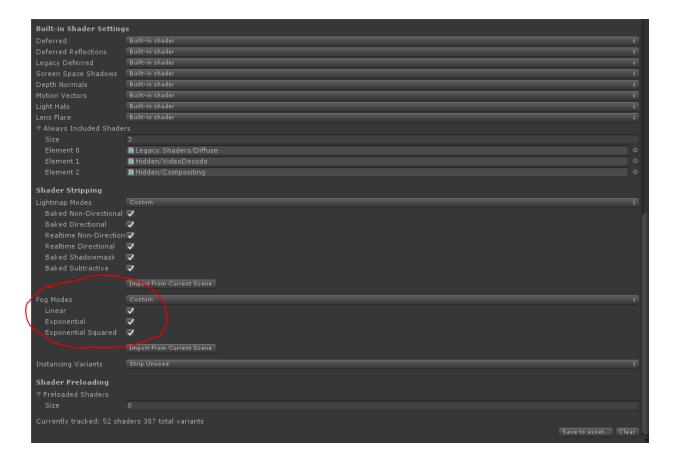
Spec Ops

This mode makes use of **SpawnSystems** for attackers (Volk) and defenders (MARSOC) as the regular non-FFA game modes do.

Place **Spawn System** prefabs for both Attackers and Defenders in the fields of the **GM_SpecOps** prefab. Once you've moved your spawns into unique locations, your setup is complete. Note the section below on fog settings if you have not already verified your settings before.

► IMPORTANT

If your map supports Spec Ops please double-check that your **Shader Stripping / Fog modes are set to Custom and all checkboxes are checked**. These checkboxes can be found by going to **Edit > Project Settings > Graphics tab** scroll down to **"Shader Stripping"** section. Ensure all checkboxes are checked and both fields for Lightmap Modes and Fog Modes are set to 'Custom'



Additional gameplay components

There are additional scripts, prefabs and settings available in the Onward Custom Content bundle that will allow you to further upgrade the quality and gameplay of your maps. These components and their use are covered in this section

Note: You can find scripts by searching for them by name in the "Add Component" dropdown of the inspector

Set Physic Material

Adding the **Set Physic Material script** to any GameObject with a **collider** will allow you to choose a **physical material** to add to that collider. This will decide what type of **impact effect and penetration** a given surface has.

V 🖬 🗸 Set Physic Material (Script) Script	SetPhysicMaterial	l 국 수 , ○
Material Type		\$
Can AI opponents see through this material? AI See Through		
Apply same settings to children, too? Include Children		

There are a few different physical materials available but the general overview is as follows:

- Concrete Impenetrable
 Glass Penetrable, reduces damage slightly
- Glass Unpenetrable Impenetrable
 GlassVfx Penetrable, does not reduce damage
- Metal Penetrable, reduces damage Metal Unpenetrable - Impenetrable Wood - Penetrable, reduces damage Wood Unpenetrable - Impenetrable
- Dirt Impenetrable
- Grass Impenetrable
- Cloth Penetrable, reduces damage slightly
- Cloth Unpenetrable Impenetrable
- ClothVfx Penetrable, does not reduce damage
- Water Penetrable, reduces damage
- Snow Penetrable, reduces damage
- SnowUnpenetrable Impenetrable
- Ice Penetrable, reduces damage
- IceUnpenetrable Impenetrable

There is a checkbox called **'AI See Through'** for allowing AI to see through a material as well on a given object using Set Physic Material, use this for Glass or other see-through objects such as chain-link fences.

► IMPORTANT

If you have no SetPhysicMaterial, the materials of all objects with colliders will default to Concrete - Impenetrable.

Elevators

The **Elevator Prefab** is a functional elevator that players can use to send themselves or equipment/items to a paired elevator. Two Elevator Prefabs will be able to be linked together by dropping each respective Elevator Prefab instance into the field of the linked elevator instance. (Elevator A reference goes in Elevator B's field, and vice-versa.) The screenshot below shows a proper elevator set-up. Elevators will start named as 'Elevator' but you should rename them to suit your organizational needs.

Inspector '≅ Hierarchy			≙ •≡
ElevatorTesterUpstairs1 (1)			🗹 Static 👻
Tag Untagged	‡ La	iyer Default	
🔻 🙏 🛛 Transform			💽 🕂 🐥
Position	X 53.58001	Y 9.542	Z -20.518
Rotation	X 0	Y -90.00001	Z 0
Scale			Ζ1
🔻 📾 Elevator (Script)			🛛 🕂 🐥
Script	🖻 Elevator		
Paired Elevator	🗟 ElevatorTest	terDownstairs1 (1) (Eleva	tor) ⊙
	Add Compone	ent	

Note: *ElevatorTesterUpstairs1* has the connection to *ElevatorTesterDownstairs1* (its destination elevator), ElevatorTesterDownstairs1 has a setup that is a mirrored version (reverse) of this.

Ammo Boxes

The **'Ammobox'** Prefab is a cache of ammunition and other supplies that can be found across the map in these game modes: **Hunt, Evac, One in the Chamber, Spec Ops**. Its contents will change depending on the game mode that utilizes them.



► IMPORTANT

You are required to place at least four (4) Ammobox throughout your map if your map has a Hunt/Evac, One in the Chamber or SpecOps game mode. You may add more AmmoBoxes as desired.

The ammo boxes contain the following items in these specific game modes:

• Hunt/Evac

The Ammo box spawns equipment such as magazines for weapons that a player has equipped, or grenades and syringes to support them in battle.

- One in the Chamber The Ammo box spawns a magazine containing one round for a bolt-action rifle.
- Spec Ops

The Ammo box contains different tools for MARSOC and Volk. They are as follows: Pistol magazines for MARSOC. Combat knives and a rare chance to spawn a single flashbang for VOLK.

PostProcessing Profile

Onward uses post processing to further boost the quality of the image delivered to a player during gameplay and stylistically distinguish each map. There are several different profiles available to choose from for day and night maps of varying locales. Experiment with the different choices for your use-case and determine which you like the best for your map. Alternatively, you can have no PostProcessing.

litle:	
Outlet	
Descriptions	
Description:	
Radio chatter indicates th	is shuttered mall in rural America has been occupied by a Volk cell. MARSOC has just entered it's dilapidated halls on a search and destroy mission.]
Map enviroment:	
Map size:	(Medium ÷
Time of day:	Day ÷
Post processing profile:	

Note: you will only be able to see a PostProcessing setting change when you load into your map.

OnwardCustomAudioSource

OnwardCustomAudioSource is a prefab that allows you to play your own audio files in the game. It is recommended to not use too many as there are potential memory usage issues since the audio files are fully uncompressed during runtime, and the cpu usage will be higher than the game's built in audio.

- Audio Source Type refers to how the audio is heard. Local Ambience will play from a 3D spatialized position. Global Ambience will play throughout the entire scene as if you are in the center of it.
- Audio Clip to Play is your own audio file you wish to play. Any audio type that is supported by Unity is allowed (.ogg, .wav, .mp3, .aiff/.aif, .mod, .it, .s3m, .xm). It is recommended to use .ogg since the compressed format will reduce the download size of the map.
- **Normalize Clip Volume** normalizes the volume of the audio clip. This will essentially make the 0 to 1 volume setting consistent for all audio files.
- Volume controls how loud the audio provided is.
- Loop continues to loop the audio clip after it reaches the end.
- Attenuation Type (Local) is the falloff distance for the audio source in meters.
- When Audio is Played (Global) determines if the audio source ducks when inside or outside

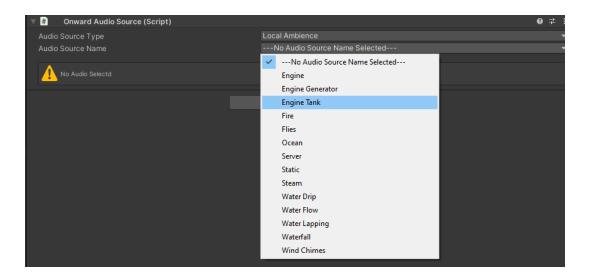
🔻 井 🔹 Onward Custom Audio Source (Scrip	ot)	Ø	ᅶ	
Script	OnwardCustomAudioSource			
Audio Source Type	Local Ambience			•
Audio Clip To Play	None (Audio Clip)			
Normalize Clip Volume	 Image: A set of the set of the			
Volume	0.5			
Loop	 Image: A set of the set of the			
	D020			-
Attenuation Type	0020			
Attenuation Type	5020			
Attenuation Type		0	ᅷ	:
		0	7	:
▼ # Onward Custom Audio Source (Sci	ript)	Ø		: •
▼ # Onward Custom Audio Source (Scr Audio Source Type	ript) Global Ambience	0		: •
 # Onward Custom Audio Source (Sci Audio Source Type Audio Clip To Play 	ript) Global Ambience None (Audio Clip)	0		÷ ▼ ⊙
 Onward Custom Audio Source (Scr Audio Source Type Audio Clip To Play Nomalize Clip Volume 	ript) Global Ambience None (Audio Clip) ✓	0		•

OnwardAudioSource

OnwardAudioSource is a prefab that allows you to select audio used in the game and play it. Some good uses for it are to break up the silence in your level by adding ambience to it.

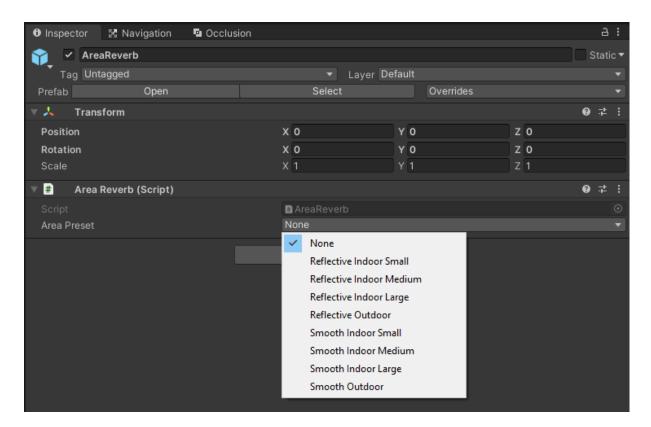
- Audio Source Type refers to how the audio is heard. Local Ambience will play from a 3D spatialized position. Global Ambience will play throughout the entire scene as if you are in the center of it.
- Audio Source Name is the type of effect you wish to play.

Use Onward Audio Source to give some character and ambience to your custom scene.



AreaReverb

AreaReverb is a prefab that allows you to add different spatial effects to areas allowing for more realistic sound reverberations. Some examples are the Hallway preset for tight hallways, or the Auditorium preset for large open interior areas. Experiment with your local map versions to try different sound reverberations out and see what fits for you.



Place the AreaReverb prefab into your scene, then scale it to enclose the area you wish to have reverberations within and set the Area Preset to your desired effect.



Check For Errors

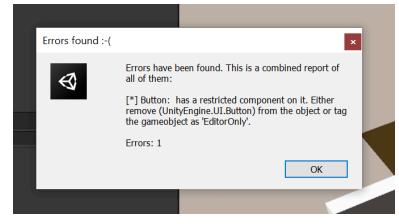
Click **ONWARD > Check scene for errors**. If there are any errors they will be listed in a dialogue box.

e-F	PC, Mac & Linux Standalone - Unity 2019.4.36f1 [PREVIEW PACKAGES	IN USE]*
ent	ONWARD Services PlayFab DownpourInt Window Help	
	Check scene for errors	
	Fix Lit materials in scene	
	Scan for URP migration incompatible materials in scene	
	SDK Version	
	Onward Dashboard	NT ASSE
	Onward Launcher	
	Optimize Window	
	Shader replace	d)
	Find materials incompatible with Onward	
Sce	Strip Unity AudioSources	
	🔂 EventSystem	

Restricted Components

Restricted components are things that may not be compatible with Onward. If you see an error message mentioning a restricted component there are a couple ways to fix it.

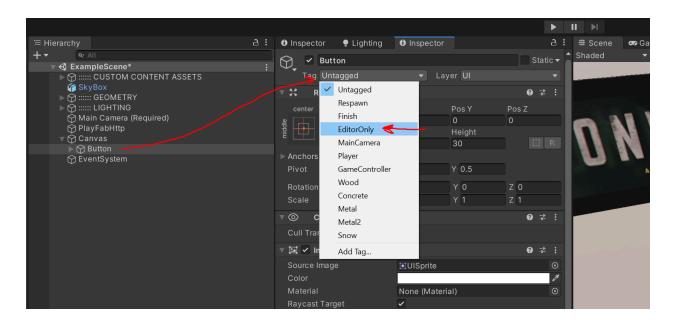
In this example you can see that the Button is a restricted component.



The first way to fix it would be to simply go to the game object holding the component and remove it.

 O PlayFabHttp Y ⊕ Canvas D Button O EventSystem 	₩ Anchors Pivot Rotation Scale	Width 160 X 0.5 X 0 X 1	Height 30 Y 0.5 Y 0 Y 1	Z 0 Z 1	UN W	VR V
	O Canvas Senderer Cull Transparent Mesh Cull Transparent Mesh			07:		
		■UISprite None (Materi > Sliced 1	al)	● 7 ² : ○ ✓		
	Sutton Interactable	2		0 .	Reset	
	Transition Target Graphic Normal Color Highlighted Color Pressed Color Selected Color Disabled Color Color Multiplier Fade Duration	Color Tint EButton (Ima 0.1	nge) 		Remove Component Move Up Move Down Copy Component Paste Component As New Paste Component Values Find References In Scene Edit Script	
	Navigation On Click ()	Automatic	Visualize	-		
	On Click () List is Empty Default UI Material			+ - 0¢		

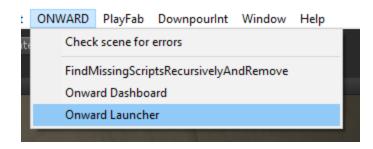
If it is something required at edit time, a terrain tool you are using, or an automated light probe placement script for example. You can tag the object as "Editor Only" This removes that object from the build but lets it stay in the scene file.



Testing your maps

You can test your maps by using a local version and your installed Onward game client to test without uploading. For publishing instructions, refer to the **Publishing** section. For local testing, refer to the instructions immediately below.

You will be able to test your maps without uploading them and waiting for approval. You can use the **Onward** dropdown for this. Select **"Onward Launcher"** and a new window will appear:



PC Testing

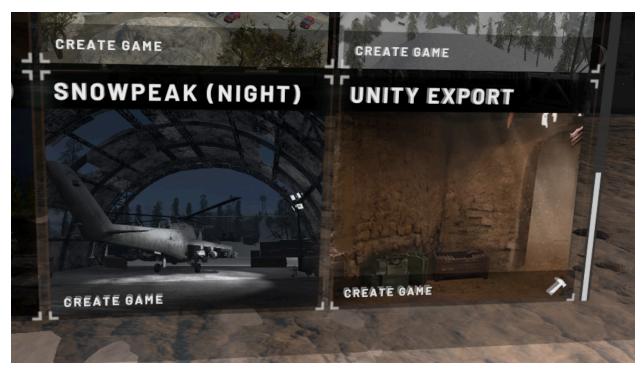
Direct this window to the prompted file path (as shown in the screenshot) and press "Build"

Onward Launcher	× _≠≡
Path to Custom Content folder:	
C:/Users/Storm.DESKTOP-E7CK20G/AppData/LocalLow/Downpour Interactive/Onward/CustomContent	
The path should look like: C:\Users\YourUserName\AppData\LocalLow\Downpour Interactive\Onward\CustomContent	
Build	

A progress bar will appear indicating the packaging of your map. Once it is complete, you will receive a success message.

Map should be locally available as 'Unity Export' now.	Success!	×
	\triangleleft	Map should be locally available as 'Unity Export' now.

After getting your success message, you can launch a locally-installed version of Onward and browse for your Custom Map using the server creator. It will be labeled as "**Unity Export**".



Locally-hosted Custom Content in the map selection menu.

Quest testing

Testing for Quest - Set your Platform to Quest and press "Build". You will be prompted where to save the files. Select a memorable folder location.

Onward Launcher			: 🗆 ×
Platform	Quest		▼
		Build	

Upon Success you will see a message denoting where it is to be installed.

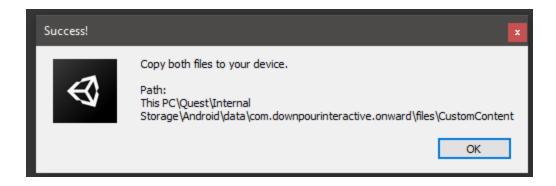
► IMPORTANT

To test on Quest you **MUST manually** add your map locally on the device and you must have Developer Mode enabled on your Quest device. Plug the device into your PC with developer mode active, then browse the file structure and add the files created by the Onward Launcher Builder you saved to a memorable file location earlier to the directory below.

> QuestMap			
Name	Date modified	Туре	Size
unity-export.local.content2	10/13/2020 11:19 AM	CONTENT2 File	36,499 KB
unity-export.local.info2	10/13/2020 11:19 AM	INFO2 File	1 KB

Add these files to

yourdirectory\Quest\InternalStorage\Android\data\com.downpour interactive.onward\files\CustomContent



From there, you can launch the map locally on your Quest device when the transfer is complete.

PC performance considerations

- Try to not exceed 750,000 vertices and 700 draw calls the lower, the better.
- Make use of Occlusion Culling, Lightmapping and other performance tools in Unity.
- Resources for these tools and more can be found at Learn | Unity (<u>https://unity.com/learn</u>), or on YouTube
- If your submitted map suffers from major performance issues it may be denied.
- You should investigate your textures and scale them to a lower resolution if possible. The default resolution of many textures is in some cases egregious for the level of detail required and will only result in a bloated level size and slow loading times. You can inspect a given texture by clicking on it and choosing its max size. Reduce this from 4k to 2k then "Apply" and observe your texture in the scene. If the quality decrease was acceptable leave it, or even lower it further until it's not acceptable- then raise it up one level from that.

Lightmap-1_comp_dir Import Settings		🗑 🕂 🐟
		Open
Texture Type	Default 2D	ŧ
Texture Shape		
sRGB (Color Texture)		
Alpha Source		
Alpha Is Transparency		
▼ Advanced		
Non Power of 2		
Read/Write Enabled		
Streaming Mip Maps		
Mip Map Priority		
Generate Mip Maps	\checkmark	
Border Mip Maps		
Mip Map Filtering		
Mip Maps Preserve Coverage		
Fadeout Mip Maps		
Wrap Mode		
Filter Mode		
Aniso Level		
\sim		
Max Size		
Resize Algorithm		
Format		
Compression	High Quality	
Use Crunch Compression		

Quest performance considerations

► IMPORTANT

Quest scenes must use extremely limited draw calls, on-screen triangle counts and texture memory to perform optimally. Creating a custom map for Quest is intended for advanced map creators. Your submission may be denied if it does not meet performance requirements and causes simulation sickness.

Rough guidelines (the best thing to do is test!):

- ~150 draw calls
- ~250,000 triangles
- ~1GB of texture usage

These limitations account for players, objectives and other gameplay objects for you- so you can develop right to the edge of these specs if you wish.

Quick tips for developing Custom Content for Quest crossplay:

- Usage of occlusion culling is recommended for Quest scenes.
- Utilize clever line of sight to avoid large overview areas that will break your frame limits.
- Share the same shaders as often as possible to let the SRP batcher combine draw calls for you.
- Use smaller resolution texture arrays vs high resolution atlases when possible. This avoids the tiling issues caused by atlasing while increasing the texture look up speeds on the pixel shader.
- Use the frame debugger to check out your draw calls and triangle counts at many given points throughout the scene by placing a camera in your scene and removing it before submission once you are satisfied with your results.

There are many tricks and techniques to make a lean, yet playable level and these are just a few of them. Respect the performance of the Quest 2 platform and you will be rewarded with smooth gameplay and many highly-rated reviews of your custom content map.

Best practices in development

Onward is a game with an established set of rules, based on realistic squad-level combat. You should keep this in mind while designing your maps and objectives to ensure an intuitive and fun gameplay experience for all players.

We highly recommend blocking out your maps with simple 3D shapes and playtesting them before finalizing your map design. This will give you a better understanding of how your map feels, and what sections need to be modified before you start polishing it visually.

Regular playtests are a great way to see how your map evolves. They help you locate areas where players might have difficulty moving through the map, lines of sight that may be too short/long, and spots where lighting issues may occur.

Lastly, getting in touch with other map creators gives you the opportunity to share experiences. You can pick up tips and tricks from more experienced creators, or share your own ideas with others looking for help. Constructive feedback is one of your most useful tools.

Publishing

This is what you've been working for - once you've set up your map, built it in Unity and tested it thoroughly, it's time to publish it!

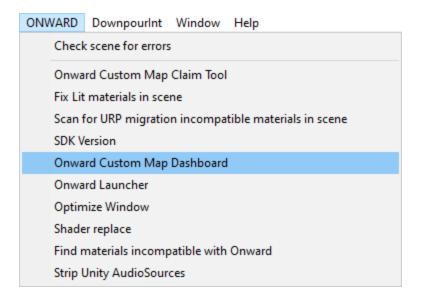
To upload, modify, or submit your custom map you must have an Oculus account and the oculus client open on your PC.

Install the Oculus app on your PC | Meta Store

Follow the guide to install the Oculus client and setup or login to your Oculus account. You do not need to set up a headset, but you MUST own Onward on your Oculus account.

Once the client is installed, you must restart your PC for Unity to register this change. Once your PC has restarted, open the Oculus client; it is required to be open to work with the custom map dashboard and claim tool.

First, in the Unity top menu, open the **Onward Custom Map Dashboard**.



Onward Custom Map Da	ONWARD		
		Find Level Descriptor	
	Publich/Update		Browse My Maps
Title: Al-Mad			
Description: Abando	ned by the local population two months ago, insurgents have taken over, intel suggests the insurgents have re		fore they can deels their communication logit
Map Environment: Map size:	Desert Medium		
Time of Day:			
Map Image Preview: Thumbnail (\$12x256 p			
Display Version:			<u>.</u>
 I agree to the follow Downpour Interactive Downpour Interactive 			N
 I agree to the follow Dowrpour interactive Dowrpour interactive Dowrpour interactive If you are running 			
 I agree to the follow Dowrpour interactive Dowrpour interactive Dowrpour interactive If you are running 			
 I agree to the follow Dowrpour interactive Dowrpour interactive Dowrpour interactive If you are running 			
Lagree to the follow Downpour Interactive Downpour Interactive Downpour Interactive If you are running M You can find the S Eat -> Project:		Uplans Mag	
Lagree to the follow Downpour Interactive Downpour Interactive Downpour Interactive If you are running M You can find the S Eat -> Project:		Union Mat	
Lagree to the follow Downpour Interactive Downpour Interactive Downpour Interactive If you are running M You can find the S Eat -> Project:		Upane Mad	
Lagree to the follow Downpour Interactive Downpour Interactive Downpour Interactive If you are running M You can find the S Eat -> Project:		Union May	
Lagree to the follow Downpour Interactive Downpour Interactive Downpour Interactive If you are running M You can find the S Eat -> Project:		union Nor	
✓ Lagree to the follow Dowrpour Interactive Dowrpour Interactive Dowrpour Interactive If you are consist at ~> Project 1		Union Ma	

Take a moment to read and accept the **Terms of Service**.

Make sure you include a clear title and description. We recommend following the suggestions below to ensure that your information displays correctly:

- Title: Try not to exceed 24 characters, however 6-18 is ideal
- Description: Try not to exceed 500 characters, however 250 is ideal
- Map Environment: Change this to match your map
- Map Size: Change this to match your map
- Time of Day: Change this to match your map.
- **Post Processing Profile:** optional setting, covered in the Additional Gameplay Components section of this document.
- Thumbnail: 512px wide by 256px tall.
- **Display Version:** for a finished map, use 1.0

Once you have filled out the information, a **Publish** button will appear. Click it to begin the submission process. Depending on the size of your map, this may take some time. Once it is complete, you will be notified with a **Success** message.

If you receive errors in your console log or any pop-up error message, take time to review the message and determine the issue with your scene, take steps as advised to fix the issue, then try again. The Onward Custom Content bundle will detect errors in your setup automatically and try to help you fix them.

► IMPORTANT

After your submission is sent, you will need to wait for approval. Approval may take up to 2 weeks. If you are denied, you will be notified. Updates must also be approved. These will be reviewed once per week, same as before. This means that you can submit your updates at any time and they will be reviewed within two weeks of your submission in most cases, be it updates or a new map submission.

Once you've published your submission and it has been approved, you can update it to submit map changes and improvements. The button at the bottom of the **Publish/Update** window will now read **Update**. Click it to submit any revisions. Once your map is available to the public, use local testing to test major changes before deploying them to the live servers. This will help prevent issues for users if you accidentally break some of your map's functionality. Reserve uploading and publishing updates to your map for builds which you are certain function correctly.

Conclusion

Thank you for reading through this guide, and for submitting your custom Onward map(s). We're excited to present them to the community, and we appreciate the time and effort you invested to create new experiences for your fellow players.

If you have any issues submitting your custom content, or if you find any errors in this guide, please let us know via the Onward Discord channel so we can address them. Your feedback will help us provide the best experience for creators and players alike.

Thanks again for your support.

"Onward!"

- Downpour Devs

Appendix A: Updating your project

SDK 9 Upgrade - Onward v1.9

There are a few more steps to update your Custom Map SDK compared to before, because the backend system has changed; but worry not, it is still straightforward. Let's get started!

Upgrading your map consists of 5 steps

- 1. Delete the PlayfabSDK folder in your project
- 2. Delete the OnwardCustomContent folder in your project. Note: your scene will appear pink. Ignore this for now.
- 3. Install the new CustomMapSDK into your project. Your scene should look normal again after it finishes.
- 4. Install and have the Oculus Client open whenever you plan to modify/submit your map
- 5. The Onward UGC package now contains a Registry Key Config that you must apply to your PC's Oculus Registry Settings. To do this, follow these steps:
 - Find the OnwardUGCRegConfig file at the path: Assets/RegistryKey in your project after importing the UGC SDK
 - Simply double click OnwardUGCRegConfig in the project view and you can automatically apply the developer key necessary to work with UGC.

Once done. Restart your PC.

SDK 13 Upgrade - Onward 1.13

- There is now only one sky box shader.
- Material map tiling & offset, the properties are new so any materials that previously used the old single tiling & offset will now use the default value.

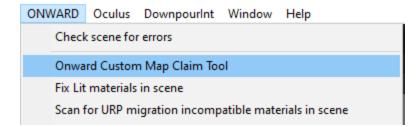
Test_Ma									
		ward/Shi	ader Graphi	I/PP_Opaque					▼ Edt
 Added 								Check	in Undo
▷ Surface Opt									
Base Map									
Tiling									100
Offset									States
Base Color					-	_	_		/
Specular Ma									
Tiling									
Offset									Select
Specular Co				_	-	~			
Normal Map					-				None
Tiling									
Offset									
Emission Ma									
Tiling									
Offset									
Emission Co									
Smoothness							•		
T Options									
Alpha Clippi									
Alpha Clippi							•		
Emission									
▷ Advanced G	lptic	ms							

Appendix B: Claiming map ownership (update 1.9 onwards)

Due to the backend change, all maps are now tied to your Oculus ID. This ID has no relation to your Downpour account which has been deprecated. As a one time process, older maps will need to be claimed by their original creators. By doing it this way, current maps can continue to be played while we assign creators to them.

Claiming your map(s) is a simple, one time process! Let's get started. First, make sure to update your custom map SDK to the latest version, as detailed in **Appendix A: Updating your project to work with Onward Workshop**.

Once this is completed, we will need to use the **Onward Custom Map Claim** tool; Open this now.



Once you have the tool open, you will need to fill in some information, including your playfab email address you used prior to 1.9 to submit your custom maps, your Discord username, and last but not least the maps you wish to claim.

NOTE: Discord account is optional but it is very easy for us to reach out to you with it.

Onward Map Claim Tool						: 🗆 ×
		Send You	ur Map Claim Re	quest		
Enter your original PlayFab Ac						
Enter your Discord username:	: (Name#1234)	Coloct the money	eu went te elein			
Map Filter:		Select the maps y	ou want to clain	has your own!		
Map Filter.						
Unov	wned Maps				Claimed Maps (2)	
Training Facility			> Pane	da Towers		
Old Town Day		>:	> Forg	otten		<<
Fallen Isle			>			
Terminal (CODMW2)		>	>			
Dustward			>			
		Sub	omit your Map C	laim!		

With your claimed maps selected, click '**Submit your Map Claim**'. You will receive a response from Downpour in a few weeks in Discord or in email once your request has been approved.

When your scene is approved, your map is almost ready! You will need to convert your old Playfab ID to the new ID system. Open your converted scene, then open the **Onward Custom Map Dashboard**. You will see an error at the bottom of the page, this is telling us that the ID is incorrectly set. Let's fix that!

Map Environment:	Woodland	
Map size:	Small	
Time of Day:	Day	
Map Image Preview:		
Thumbnail [512x256 px]		Select
Display Version:		
 I agree to the following Downpour Interactive Cus Downpour Interactive End Downpour Interactive Priv If you are running into 		
	razies and ous du along ingli memory consumption peake consider changing the onioder on pping options. er Stripping settings here: ings -> Graphics	

Click '**Browse My Maps**' at the top right of the dashboard and find your scene. If your scene has not been accepted as claimed yet it will not appear, so be patient until it has been reviewed.

Once it is available, you can finish setting the new ID. Make sure you are on the correct scene before continuing! Click the **Assign** button to the correct map. Once you have done this, your scene will be fully converted to the new system! You will then be able to update your scene whenever you want.

Onward Custom Ma	
	Find Level Descriptor
	Publish/Update Browse My Maps
	Refresh your Custom Maps Info
Part Meter	Last Stand[AI ONLY] Marsoc must defend the last bastion in the region. Using high ground and advantageous positions, defend against a superior sized force of VOLK. [AI ONLY] Rating: 0 Version: 0.7 IN REVIEW
	Your scene's man id is invalid or emoty. Do you you do not in scion it's ID to this man?
	Assign Last Stand(Al ONLY) ID to active scene's level settings.
	Al-Madinah Abandoned by the local population two months ago, insurgents have taken over. Intel suggests the insurgents have received weaponry and training from the Russian Military. Assault and Secure Madinah; in addition, secure intel Rating: 0 Version: 10 LIVE TO PLAYERS
	Your scene's map id is invalid or empty. Do you want to assign it's ID to this map?
	Assign Al-Madinah ID to active scene's level settings.

Appendix C: Optimization tools & tricks for Quest

Below are a few tools and features that will help you optimize your maps for Quest if you choose to publish for Quest.

OVRMetricsTool

This is a handy tool you can sideload on your device that allows you to monitor different performance metrics. Once it is installed, open it from your 'unknown sources' list and a new window will appear with settings for OVRMetrics. You will want to 'enable persistent overlay' and switch it to the 'advanced' view. From there, the main metrics you will want to check out other than FPS are CPU/GPU usage and App T (GPU time per frame in Microseconds).

The most important thing to look at is the overall frame time. Unity will process the game frame on the cpu and then pass it off to the GPU. You may see data like 70% CPU usage and 50% GPU usage and have performance issues. There are a few ways to fix this. These are recommendations so try mixing and matching solutions that fit your scenes requirements.

1. Reducing the draw call count through reducing shader count, mesh combination, or improved culling/LODs will all help on the render thread of the CPU and help the GPU render more efficiently.

2. Reduce texture resolution and use ASTC compression. The CPU must bundle up each draw call and the smaller that bundle is the faster it can send it to the GPU. Small textures and proper compression helps quite a bit. Lower resolution textures also increase the speed of texture lookups so it increases the render speed.

3. Reducing shader complexity will improve the GPU rendering speed. Using the PP shaders provided in the SDK will be the easiest way to achieve this. Vertex shaders are much faster than pixel shaders. If something can operate on a vertex try to do that. Try using look up tables if an intense operation needs to happen often.

4. Transparency is terrible on the tile based renderer used by the Quest. Keep transparency areas small, break it up into multiple meshes, or find an alternative. Sometimes having polygons for leaves instead of using transparency ends up being faster.

5. Use texture arrays when possible. This will require some shader work but it could be worth it. <u>https://docs.unity3d.com/Manual/class-Texture2DArray.html</u>

Onward uses Vulkan and the SRP batcher. The SRP batcher handles a lot of the cpu side optimization for you while Vulkan improves the CPU and GPU rendering efficiency. Find more information here: https://docs.unity3d.com/Manual/SRPBatcher.html

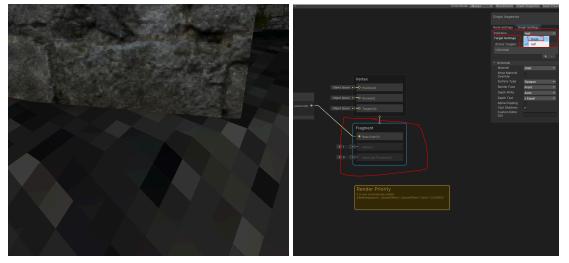
Optimize Window

This is a handy tool we have added to version 2.0 of the custom content package that allows you to see high-triangle count meshes in a list that will help with identifying assets for decimation or further level of detail usage. Please use it as you see fit to help optimize your Quest scenes.

Appendix D: Troubleshooting

Textures are Pixelated on the Quest

Change the shaders precision from Half to Single.



The skybox gets cut off

The far clip plane on the player's camera has been reduced from 2,000m down to 1,000m. If you are using the sky box prefab/mesh, swap it out for the built in skybox. Remember to check the "Use Built in Skybox" on the shared level settings object.