MicroSplat F.A.Q.

These are the most common questions I get, just about all of which are covered by reading the included documentation.

Pre-purchase questions:

Q: **“Does the terrain collection include module X”**
A: The terrain collection description on the asset store lists exactly which modules it includes. If it is not in that list, it is not included.

Q: **“Will module X be added to the terrain collection?”**
A: No

Q: **“Does module X do Y?”**
A: The documentation for every module is included in the free Core module, so you can look through how everything works before purchase

Q: **Do I need to purchase the HDRP module if I want to run in HDRP? Or the URP module if I want to run in URP?**
A: Yes

Q: **When will feature X be available in render pipeline Y**
A: When Unity stops constantly changing the pipelines or provides a reasonable abstraction layer, or whenever I decide I can add it without having to re-code it every few weeks.

First use Questions:

Q: **“How do I start?”**
A: Preferably by reading the documentation I spent so much time writing, or watching one of the many videos I provide on YouTube.

Q: **“I installed MicroSplat and the URP/HDRP module on Unity 2026 alpha and the shader is pink!”**

A: You need to set the render loop on your shader to the appropriate render pipeline; this is covered in the documentation for the render loop adapters. Also, read the supported versions for the SRP modules, as Unity often breaks compatibility in SRP pipelines and only some versions can be reasonably supported.

Q: **“The textures are too large/small”**
A: UV scaling is available as both a global under the “Splats” section of the material, or in the per-texture properties section at the bottom of the shader. When you convert a terrain, if it has existing textures with very different scales, it will turn on per-texture UV scale, set the global scale to 1, and compute the scale for each texture as part of the conversion.

Q: “The textures are blurry!”
A: See above

Q: “I added a new texture and it’s blurry!”
A: See above

Q: “I added a new texture and it shows up black”
A: “Make sure your Max Texture count on the material is higher than the number of textures you have in your Texture Array Config”

Q: **“I keep editing the terrain layers but they just get ignored”**
A: As stated in the documentation, once you convert to MicroSplat you manage all textures from the Texture Array Config. This is because Unity’s layer system is not extensible, and does not have a way to store all the types of textures MicroSplat supports.

Q: **“How do I add a texture?”**
A: See above

Q: **“I added a texture but it won’t let me paint it”**
A: See above

Q: **My texture is shiny**
A: This is controlled by the smoothness texture provided in the Texture Array config, if you did not provide one, smoothness values are automatically generated. Other 3d applications often use roughness maps, which can be automatically inverted by clicking on invert in the texture array config on the smoothness texture. There is also a per-texture property to adjust the smoothness of each texture. Finally, read the core documentation on the Unity lighting model and alternate lighting models provided if you are running the standard renderer.

Q: **“The blending looks different than the Unity Terrain”**

A: By default, MicroSplat uses a height map based blend for terrain blends. If you prefer blurry Unity style blends, you can disable height map based blending in the core features section, or adjust the amount of height blending with the interpolation contrast slider.

Q: **“I want multiple terrains to share the same texture arrays and shader and such”**
A: Simply select all the terrains when doing the conversion, and one MicroSplatData directory will be created for all of them. If one terrain has been converted already, simply add the MicroSplatTerrain component to the new terrain, set the template material to the one in your MicroSplatData directory, and press sync.

Q: **“Can I rename things willy nilly in the MicroSplatData directory?”**
A: Sure, but you might break things. Best to not rename this directory or anything in it- that said, you can move the entire folder to where ever you want it.

Q: **“The interface looks every so slightly different in this YouTube video, will you go re-record all 26 hours of YouTube videos so it matches exactly?”**
A: No

Q: **“I removed the MicroSplatTerrain component and it destroyed my terrain and totally erased all my work”**
A: No it didn’t - MicroSplat does not edit terrain data. Most likely, the problem is that the basemap distance on your terrain is set to 0.

Q: **“I added a new texture and now it’s drawing black”**
A: Raise your max texture count on the material, it’s likely set lower than the total number of textures you are using.

Q: **“I don’t have height maps for the textures and it says MicroSplat will create them, but I don’t see them being created”**
A: They don’t get written onto disk or show up in the interface. Rather, they are generated from the best data you provide and stored directly in the resulting texture arrays. For instance, if you only provide a diffuse map, it will generate the height map from the luminosity of the diffuse map. However, if you have a normal map, it will generate the height from the normal map. It’s better to provide these textures yourself.

Feature Questions:

Q: **“I turned on Triplanar and now my textures are gone”**
A: No, they are just in a very different scale. When not set to triplanar, the default is to scale them based on the UVs, which are 0-1 across the whole terrain. So a UV scale of 500 makes th

texture scale across the terrain 500 times. But when it triplanar, they are based on meters. A value of 500 would tile the texture 500 times over a 10 meter area.

Q: "**When I turn on tessellation, my characters feet go through the terrain."**
A: This is because the physics engine cannot collide with the tessellated surface, as that is happening on the GPU. The best workaround is to use the displacement offset and center each of the textures- this is covered in the tessellation documentation. Keeping displacement distances reasonable, offset from center, and possibly combining with parallax, can greatly enhance the depth effect without offsetting the surface too far from the original positions.

Q: **"How do I set a per-texture property via scripting?"**
A: There is an API on the propData object which will allow you to set any value you want.

Q: **"How do I set value X on the material?"**
A: You can grab the materialTemplate from the MicroSplat* component and set it through the standard material.SetFloat style interface. Afterwards, call Sync on the component, or MicroSplatObject.SyncAll if more than one thing is using the same template.

Other Questions:

Q: **"I turned on a per texture property and it's not working"**
A: 99% of the time it's because something in the MicroSplatData directory has been renamed. The material editor, for instance, looks up the object which holds the per-texture properties based on the material name - it has no way to know which objects may be using it, so it can't find it from the terrain or mesh. However, if you rename the material, then it will create a new PropData object to hold these properties, but the terrain or mesh will still be using the old one. Unless you really know what you are doing, don't rename things in the MicroSplatData directory.

Q: **"Something isn't working!?!"**
A: The basic process I will put you through will look something like this:

- Are you in an SRP?
- Which Unity version are you running?
- Windows or OSX?
- Do you have compile errors in your project?
- Did you try the included example? Does it work there?
- Did you try in a clean install, instead of your project with 7000 other assets stuffed into it?

If you have not done these things, or included this information, you can save us both a lot of trouble by doing these things first.

Q: **“I get an error about too many samplers”**
A: Most platforms have a maximum number of samplers, 16 or 32. MicroSplat tries to share samplers whenever possible, but sometimes a sampler is needed and cannot be shared- if these get over 16, Unity will not compile the shader, and you will have to disable an option to reduce sampler count. Note that Unity internally uses some of its own samplers for lighting and other effects.

Q: **“When I turn on feature X, the shader does not compile, I get an error in the console”**
A: Please send me this error, along with a screenshot of your material (the top section, but preferably the whole thing). Unity version and which SRP and version you are using as well.

Q: **“How do I read what texture the player is on?”**
A: Same as you would with Unity’s shader, unless you are using custom splat maps or procedural texturing. For custom splat maps you’ll have to read the data yourself. For PT, there are utility functions that allow you to do this.

Common Unity Warnings

Q: **“I get binary to Yaml conversion errors when I install”**
A: These are a Unity bug, supposedly fixed in an upcoming release. They cause no issues.

Q: “I get an error about duplicate _MainTex”
A: This has recently been fixed by Unity and is being back ported into latest patch releases. It causes no issue.

Q: **“I see a warning on the terrain component about the shader needing tangents”**
A: This is an erroneous warning and can be ignored.

Advanced Topics

Q: “I want to change individual textures at runtime, as this is a user generated content game”
A: The easiest way to do this is to put each texture the user can select into its own texture array to pack it into the correct format, then use Graphics.CopyTexture to copy the texture into the target array. However, if you are downloading the textures as png’s or something like that, then

you will have to pack them into the correct format yourself. The code in TextureArrayConfigEditor.cs is what you want to look through to understand this.

Q: “I removed MicroSplat and it destroyed all my work!!!”
A: No, it didn’t. After removing MicroSplat from a terrain you may have to (depending on Unity version)

- Reset the material on the terrain back to the Unity default terrain material or any other you want to use
- Reset base map distance to something larger (say 200 or more, MicroSplat often sets this to 0)
- If you delete your MicroSplatData directory, any terrain layers created in there will be deleted with it, causing missing layers on the terrain. When you add new texture via the MicroSplatConfig, it will automatically generate a terrain layers for this texture in the MicroSplatData directory. Preferably don’t delete these files.