Masking

By default, decals will project onto everything within their bounds. Masking allows us to choose which objects the decals project onto, and which they ignore. The system has two approaches to this problem; layer based and material based. You can use either or both of these approaches as appropriate.

Layers

This approach uses Unity's inbuilt layers. It would be criminally inefficient to render every layer into mask buffers, so instead, we have intermediary layers called "Mask Layers". These allow us to pick the Unity layers relevant to the decal system, as well as group, name and organize them.

We can then tell our decals to either draw only on these layers or draw on everything but these layers. These options can be found among the instanced properties of the projection renderer and can be set per instance without breaking instancing. This allows us to render thousands of decals in a single draw call with different masking preferences set per decal.

Materials

Instead of specifying which layers you want to accept or ignore decals, you simply specify which materials you want to ignore decals. These materials will then be set to render after all decals have rendered. This approach is less flexible but significantly cheaper if used exclusively when paired with the mobile shader method. If used in conjunction with the layer method, this method will override the layer method (ie. If an object within a layer set to be projected on, has a material set to ignore decals, it will still ignore decals).

Example scenarios

We're drawing footprints. These are generated via a printer attached to the player's foot. Without masking the footprints would project onto the player's foot, grass, rocks and all manner of other debris. Really, we only need the footprints to project onto the ground. In this scenario, we would create a Mask Layer containing only the ground/terrain and title it something like "Terrain". Then we would have our footprints set to "Only Draw On" and then tick the Terrain layer.

We're drawing blood splatters. These are generated from a particle effect that collides with everything but the character layers and a few dynamic object layers. We want the blood to project onto all of the static layers, but ignore these dynamic layers as well. In this scenario, we could either create a Mask Layer containing all of the static layers, or a Mask Layer containing the character/dynamic layers. The preferred method being the one which contains the least layers. We would name this layer accordingly and based on the approach chosen, either "Only Draw On" the static Mask Layer, or "Draw On Everything Except" the dynamic layers.