

How Different Tile Types are Handled

Painter expects to encounter one of two basic tile varieties: Tile and TileBase. There's actually a big difference between the two.

A Tile has a sprite, a transform, and a Color. All tiles derived (subclassed) from Tile have these characteristics. However, a TileBase has none of those. Any tile that isn't derived from Tile needs special handling for accessing these properties so that icons and previews of the Sprite can be created.

For example, a Rule tile has a `m_DefaultSprite` field rather than a `Sprite` property. 2D Tilemap Extras' `AnimatedTile` is also derived from `TileBase` and has only a list of sprites but no `Sprite` property.

Plugins for TileBase types

Plugins for `TileBase` types are easy to create, and examples can be found in the `Plugins/TilePlus/Editor/Painter/TpPainterPlugins` Assets folder.

Basically, a plugin is a `Scriptable Object` asset which you place in an `Editor` folder (outside of the `TilePlus` plugin folder). It provides access to the missing properties (sprite, transform, Color).

Please create only ONE asset for each Type (C# Type). It does not have to be in an `Editor/Resources` folder, just in an `Editor` folder (so it doesn't become part of your build).

Plugins for basic Rule tiles and for the Unity `AnimatedTile` are already installed. The Rule Tile plugin works with the basic Rule tiles from Unity's 2D Tilemap Extras package.

If there's no plugin for a Type, then that Type of tile will not have any paint previews nor thumbnails in lists and inspectors. Hence, the plugins are a workflow convenience only.

If you add a plugin yourself, you might need to perform a scripting reload for the plugin to be recognized. This can be done in Painter via the Setup panel's Reload button (at the bottom of the panel).

If done correctly, the plugin's name should appear in the System Info window.

- Ensure that `Show Painter/Selection Info` is checked; plugins are listed near the end of that group's information.

Picking, the Favorites List, and TPT tile Cloning

TPT tiles require special handling when picking a tile from the scene and painting it, or if a TPT tile is picked from the scene, poked into the Favorites List, and subsequently painted on a Tilemap.

Why is this? TPT tiles are cloned from the original TPT tile asset located in a Project folder somewhere. This allows each TPT tile to be its own instance in the Scene. When picked, the clone tile is what's picked.

If that clone tile were to be repainted, any tile positions with that clone would have the same instance as the original clone. Changes to any of the clones' fields would affect all of them. Hence, we re-clone the TPT tile, creating a fresh copy. This is all handled transparently, but you will see messages such as 'Picked tile is a clone' or similar in Painter's Edit-mode inspector.

None of what's mentioned in this paragraph affects any Tile or TileBase tiles; they're never cloned.

Revision #7

Created 2025-06-23 16:40:48 UTC by Vonchor

Updated 2025-07-09 21:05:07 UTC by Vonchor