New Tiles!

New Tile Class?

The key component of TPT is a new Tile base class cleverly dubbed “TilePlusBase” (TPB). This tile clones itself when placed on a Tilemap in a scene.

Why would anyone care?

The One With No Instance Data

One of the issues that developers run into with Unity Tilemaps is that there’s no way to add fields (variables) to tiles and have the data serialized and saved in the scene just like the serialized fields of scripts used for components. This is because the Tilemap’s serialization is hard-wired to save data from the basic fields present in a Tile class.

For many (including your TPT developer) this is annoying, to say the least. Perhaps you want a configurable waypoint. Maybe you want to be able to paint a tile and set it up as a spawn zone. And you want to be able to edit fields as you usually do.

Using TilePlusBase and the supporting libraries you can have any sort of code and/or data in a tile. Since the tile is cloned, it is no longer connected to the asset in the project folder: it exists in the Scene, and its data is saved with the scene.

If you’ve used the Unity Tilemap Editor (UTE) you’ve used its Selection Inspector. That inspector is very different from the normal Unity inspector panel: the UTE Selection Inspector is hard-wired to support the fields of the Tile-class tile and that’s it.

The support libraries for TPT have an alternative Selection Inspector that’s available in the UTE as the “Tile+Brush” or by using the TPT’s Tilemap painting and editing tool: Tile+Painter (T+P).

Decorate your TPB-derived tiles with TPT’s custom attributes and this alternative Selection Inspector lets you view and edit those fields or display property values. See the Online Documentation for more information.

Revision #1
Created 2025-07-15 13:41:22 UTC by Vonchor
Updated 2025-07-15 13:42:16 UTC by Vonchor