Don't Use Prefabs for Tilemaps

You Probably Don’t Want Tilemap Prefabs

They’re not really that useful except in limited circumstances. Each time you drag in a Tilemap prefab, it instantiates an entirely new Grid with Tilemap children. This is true for any Tilemap prefab, even one created normally by dragging a Grid GameObject from a scene to a project folder.

What’s more useful is being able to load tiles in groups to existing Tilemaps. You may already do this with Tilemap block move methods and so on.

TilePlus’ TileFabs and Bundles handle all this for you and are easy to create.

TileFabs and Bundle assets are not Prefabs!

Please note that the TileFab and Bundle assets are NOT prefabs: you can’t drag either of these into a scene.

However, a TileFab archives the original source Tilemap names and/or tags and the TilePlus library functions for placing them in a scene expects to find these same names and/or tags to place the multiple archived Tilemaps correctly. That is, which Bundle assets referenced by a TileFab should be ‘painted’ on to which Tilemap. It can’t guess.

Other Uses for Bundled Tilemaps

When using Tile+Painter, Bundle assets and TileFabs appear in the Painting Source (center) column and behave as if they were a single tile: you can paint them onto Tilemaps.

This means that you can create chunks of tiles and paint them as if they were a single tile.

Bundles: you can paint the entire Bundle or view a list of all the tiles in the Bundle and paint them individually.

TileFabs When created from a Grid Selection a TileFab is like a layer cake, with the Tilemaps being the layers. Basically, you can paint N Bundles with one click.

TileFabs are also used in the Layout system.

For simpler uses, you can add the TpBundleLoader component to any Tilemap's GameObject. This component loads a single Bundle. Place the component on any compatible Tilemap (same layout, etc.) and drag in the Bundle asset reference. Switch to Play mode and loading will happen automatically if the component's LoadOnRun toggle is checked. Or you can click the Load button to make a quick test.

The Load button is hidden if the Tilemap is part of a Prefab.

Similarly, you can add the TpFabLoader component to any Grid's GameObject. It works in the same fashion as TpBundleLoader. However, it can't work correctly if the Grid's child Tilemaps do not match the names and/or tags of the Tilemaps embedded in the TileFab asset. That's up to you.

Finally, you can use a TpBundleTile. This simple tile takes a Bundle reference as a parameter. You can copy it to a Unity Palette or to Painter's Favorites list and paint it on a Tilemap. When you do, the tile loads the bundle and deletes itself.

Bundles and TileFab assets can be used in a running game to dynamically load tiles into Tilemaps using the TpBundleLoader or TpFabLoader components, a TpBundleTile, TpAnimZoneLoader tile, the TpZoneManager, or at the lowest level, the TileFabLib and TpZoneManager libraries.

One other note: When you make an archive, all Scene references are lost as usual. TPT tiles can have Scene references but if your tile doesn't have any then this doesn't matter.

Revision #4
Created 2025-07-15 17:21:06 UTC by Vonchor
Updated 2026-09-23 16:53:59 UTC by Vonchor