Properties

Why so many properties?

Fewer serialized fields for things that are just basically boolean switches used by various parts of the system. For subclasses that don’t implement a particular functionality, specific fields aren’t needed, just a constant value. For those that do, serialized and non-serialized fields allow data to be provided via the properties, or a return value is computed for the property.

The TilePlusBase class implements all the items in the ITilePlus interface except those having to do with simulation: that has default values provided by the properties in the interface (C# 8 feature).

Most of what’s in ITilePlus are used internally and it’s unlikely that you’ll use them at all. But there are three which are especially useful: ParentTilemap, TileGridPosition, and TileWorldPosition.

- ParentTilemap always has a reference to the tilemap for the tile. This is useful for a lot of things, but beware: if a tile tries to erase itself by using the ParentTilemap reference to place a null tile at the TileGridPosition, Unity will crash.
- TileGridPosition always has the location of the tile in Grid coordinates.
- TileWorldPosition always has the location of the tile in World coordinates.

Revision #2
Created 2025-07-14 11:36:45 UTC by Vonchor
Updated 2025-07-14 11:37:29 UTC by Vonchor