# Tilesheet Converter

## Motivation:

When using the Unity 2D tilemap system, gaps or incorrectly colored pixels can be shown between tiles. This can happen in the scene, game view or released product.

Sometimes the effect can be alleviated with settings changes, but the fix can be temporary or can come with a cost in terms of graphics quality.

The underlying problem appears to be caused by rounding errors, due to the nature of floating point operations. When a gap occurs, Unity tries to fill it in using the pixels *surrounding* the tile in the original tilesheet. This usually means transparent pixels, pixels outside the tilesheet (which count as transparent) or pixels from adjacent tiles. In most cases this causes unwanted artefacts in the final product.

This tool reformats tile sheets such that each tile is surrounded by a rectangle of pixels, each taking their color from the pixel at the edge of the original tile.

In most cases the tile sheet can be automatically overwritten with the new sheet, meaning the scene is updated immediately. In some cases, there isn't enough padding or offset in the original tilesheet to allow each tile to be surrounded. When this occurs, the tilesheet is expanded by the offset and/or padding amount and the user is prompted to update settings in the sprite editor.

## Installation

Make sure the "TilesheetConverter.cs" file is located in a directory called "Editor" somewhere in your asset folder.

## Instructions

PLEASE ENSURE YOUR PROJECT IS BACKED UP BEFORE USE. Either use the Unity collaboration tool to save your project to the cloud, use a version control system like Git to save your current project as a commit, or simply copy your project directory to a different location on your hard drive. This plugin will save files to your project, and may overwrite files that are in use. If user settings are incorrect or an error occurs, work may be lost.

From the "Window" menu, select "2D" then "Tilesheet Converter". You may want to dock the window, or resize it to allow a large vertical space and enough width for the controls.

Drag your tilesheet image file into the “Drop image asset” field in the Tilesheet Converter editor window. Only .png files are supported, due to texture import limitations in the Unity engine.

Enter the correct values for “tile width”, “tile offset” and “tile padding” values, matching those used in the sprite sheet editor.

- Tile Width and Height are the dimensions of the tile, without any offset or padding.
- Offset X and Offset Y are the number of empty or spare pixels around the edges of the tilesheet.
- Padding X and Padding Y are the gaps in between tiles on the tilesheet. A padding of 2 means there are two pixels between tiles, rather than there being a gap of two pixels around each tile (which would make it 4 pixels gap between tiles.)

Click the “Slice” button. This is non-destructive and will not save any files.

If all goes well, you will see a collection of tiles appear under the slice button. Pay attention to any text between the “Slice” button and the new “Source Image” control. This will inform you of errors, input mistakes, or any instructions you may need to follow after saving your image.

Check the tiles in the “Tile preview” area to make sure they are sliced correctly. Use the “Tile thumb size” slider to get an overall view or zoom in to see individual tile detail.

If your tiles look misaligned (edges appear inside tiles for example) you will need to adjust your offset or padding settings. Do so, then tap “Slice” again.

Once your tiles are properly aligned, check or uncheck the “Make backup” and “Overwrite original” checkboxes for your desired outcome.

If “Make backup” is checked, a copy of the original file will be created in the same directory as the original before conversion.

If “Overwrite original” is checked, the original image file will be overwritten with the fixed tilesheet.

If “Overwrite original” is not checked, a new file will be created containing the converted tilesheet, in the same directory as the original.

Click the “Overwrite original” or “Save updated file” button (the text will change depending on your checkbox choice) to perform the conversion.

Right-click within the “Project” window in the image file’s directory and select “Refresh” in order to see the file changes. This may update the Scene window at the same time, if you have chosen to overwrite your image file.

If you decided not to overwrite the original file, or if the image had not yet been sliced, or if the tilesheet converter window displayed messages telling you to change your sprite slice settings, you will need to slice or re-slice your tiles in the usual way using the Unity sprite editor.

If you decided to overwrite the original file and the offset/padding values were compatible with gapless tiles, your tiles should now be fixed and no further action is required.

## Example scene

The included example scene contains examples of tilesheets that exhibit gaps. It can be used to practice using the utility before using it on a working project.

## Contact

In case of queries, issues, errors or feature requests, please contact ben@chemica.co.uk or via http://www.chonkypixel.com

## Disclaimer

Chonky Pixel do not take any responsibility and are not liable for any damage caused through the use of this product, be it indirect, special, incidental or consequential damages (including but not limited to damages for loss of money, loss of time, loss of business, loss of profits, interruption or the like.)

Please back up your work.