Intro

Testudo3D (T3D) adds an interactive mode to help place tiling modules.
T3D simply helps create and position dupli-groups.
Expert users can use **auto-tiling** to procedurally generate tilemaps.

This tool was initially designed for game developers, but it could have some interesting uses beyond game development.

Using Blender as a level editor is a radical choice, but there are compelling reasons to try it:

- do modeling and level design simultaneously
- use the program you are already comfortable with
- game engine independent (no platform lock-in!)
- blender is free and open source
- blender is easy to script and extend with addons

Note if you intend to use for game development, you will have to figure out the **export** part yourself.
If you can't figure out how to import a scene made with blender into your game engine, this tool will not be much use to you as a level editor.

Tested with 2.78 and 2.79

Limitations

- Blender doesn't perform as well as you might expect with 100's of objects in the scene.
- auto-tiling doesn't do **diagonals** or **terrain**
- tiles must be 1x1x1
- can only undo once exit modal operator (but it does work)
- controls not configurable (todo)

Contact

If you find a bug or something to be improved, please contact **alcornwill@gmail.com**

Turtle Graphics?

wikipedia
(This is where the name comes from, Testudo = Tortoise in latin)
to see an example of turtle graphics, run the unit tests in 'test/run_tests.py'
NOTE you must have **blender in your environment variables** to run the tests
Python also has some turtle graphics samples you can run **(Python36\Lib\turtledemo)**

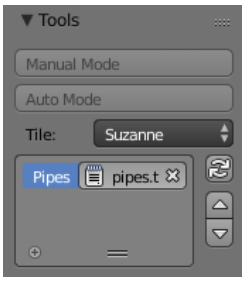
Install

The folder you need to install is **testudo3d/testudo3d/**
i.e.

- **unzip** testudo3d.zip
- find testudo3d/testudo3d/
- **zip it**
- in Blender, install **testudo3d.zip** (User Prefs > Addons > Install From File)

You will find all the tool's operators in the **T3D tab**. (3DView > T3D)

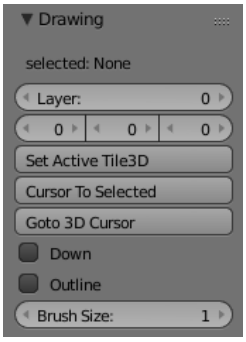
T3D Tab



tools panel

Tools Panel

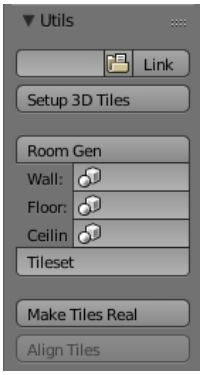
- **Manual Mode** position tiles by hand, choose specific tile to paint with
- **Auto Mode** automatically choose tile and rotation from **rules** (aka **auto-tiling**)
- **Active Tile** the active tile used in **Manual Mode**
- **Tileset List** the list of tilesets in the blend. A Tileset can have **rules**.
- **Refresh Tilesets** refresh the tileset list



drawing panel

Drawing Panel

- **Active Layer** tiles will be created/deleted in the active layer
- **Cursor Pos** cursor position
- **Set Active Tile3D** set the selected **tile** as the active tile
- **Cursor To Selected** move the cursor to the selected tile
- **Goto 3D Cursor** move the cursor to the 3D cursor (draws a line if 'down')
- **Down** is the 'paintbrush' down
- **Outline** draw an outline instead of a filled circle
- **Brush Size** the size of the circle

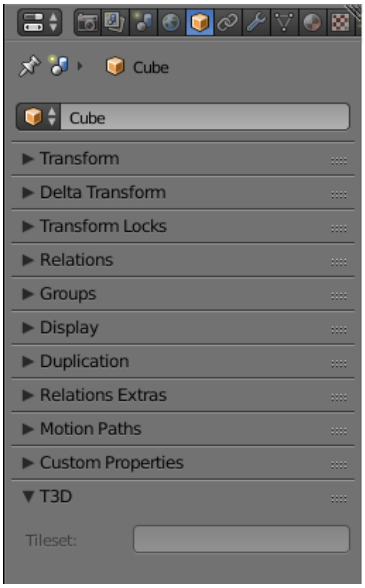


utils panel

Utils Panel

- **Link Tile Library** convenience tool to link a tile library
- **Set/Clear Tileset** set/clear the tileset that selected tiles belong to
- **Setup 3D Tiles** essential utility for creating tiles
- **Room Gen** powerful tool for automatically generating a tileset + rules
- **Make Tiles Real** like 'Make Duplicates Real' but only top-level

- **Align Tiles** align objects to the grid (if you have been moving by hand)



object properties panel

Object Properties Panel

- **Tileset** The name of the tileset this object belongs to (greyed out if not pressed **Setup 3D Tiles**)

Workflow

overview

- Create a set of tiles
- use **Setup 3D Tiles** to automatically create groups for every object in the scene and arrange them nicely
- add the tiles to a **Tileset** (select object, Properties > Object > T3D > Tileset)
- refresh the tileset list

additionally, if using auto-tiling

- Create the rules
- link the rules to the tileset

Once setup, you can hit **Manual Mode** or **Auto Mode** to start painting.
The **testudo3d/samples/** folder contains lots of blends with tiles and rules already setup, if you just want to get started.

Tiles

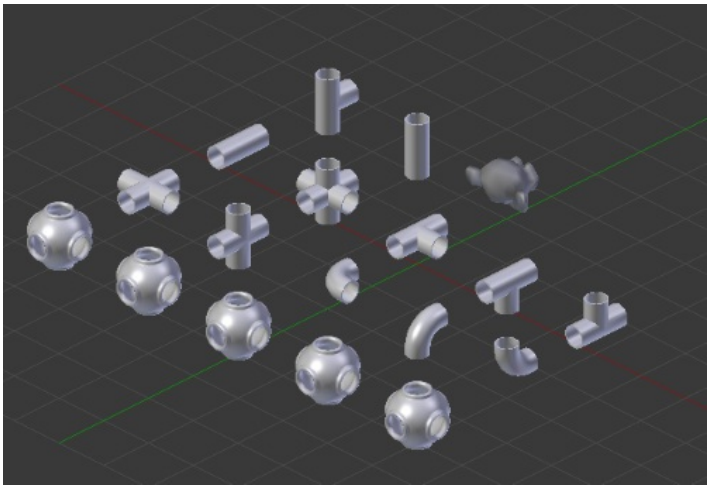
a **Tile** is just a group instance.
You should not create groups yourself as **Setup 3D Tiles** will do this automatically.

To create a tile:

- Create a top-level object in the scene. The object can have any number of children.
  The **object origin** should be at the bottom of the tile (see sample blends for examples)
- press **Setup 3D Tiles**.
  This will add the tile and it's children to a **group** of the same name, so it can be duplicated.
  It will also layout your tiles nicely
- set the **Tileset** (select object, Properties > Object > T3D > Tileset)
  A tile must belong to a Tileset.
  To add multiple tiles to a tileset at once, use **Copy To Selected**

Tiles can be any size, though they must be **1x1x1** to tile.
(actually, there is a hack for changing tile z. on a root empty, change the custom property 't3d_tile_size_z')
Since they're just objects, you can add, delete, move them around and rotate them by hand if you want

Tip: **Setup 3D Tiles** will not operate on hidden objects, if you want to exclude certain objects
Tip: use **Align Tiles** to align objects to grid. (3DView > T3D > Utils > Align Tiles)
Note: some object types cannot be duplicated with dupli-groups



Tiles (samples/pipes.blend)

Tilesets

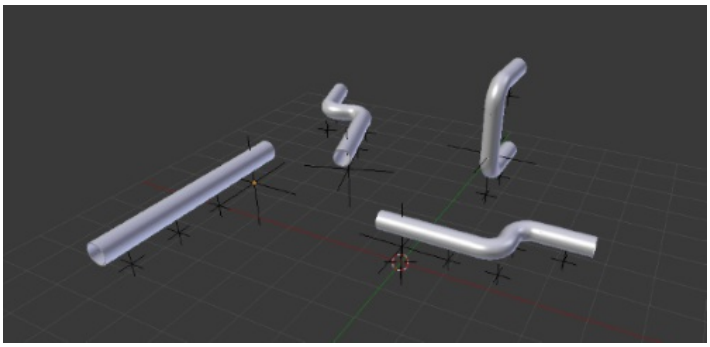
A **Tileset** is just a collection of tiles.
Once you have created some tiles and set the tileset name, press **Refresh** next to the **tileset list** to auto-detect tilesets in the scene.
You can have tiles from multiple scenes in the same tileset; as long as the tileset name is the same, they are considered to be in the same tileset.
If you are using **auto-tiling**, you can set the rules in the tileset list

Roots

When you run Manual/Auto Mode, an empty called 'Root' will pop up in your scene
Using multiple roots can help organize your scene.
Roots can have any transformation, they do not have to stay at 0,0,0
This helps you create levels that feel more organic, as you can break right-angles by rotating the root

Any object under a root will be treated as a tile, however if the object is not an actual tile you will not be able to duplicate it.
Hence, you can actually have nested roots (a root treated as a tile).
Though, perhaps a better way to make a tree of roots is to use a ChildOf constraint.

NOTE when you enter manual/auto mode, the currently selected object will be used as the root
be careful to always select the root, or you may accidently create a nested root.
In fact the root object can be anything. It doesn't have to be an Empty.
If no object is selected, a new 'Root' will be created



multiple roots

Layers

Notice the **Layer** number in the T3D Drawing panel

Tiles will be created/deleted in the active layer
Since a cell can only be occupied by one tile at a time, layers are needed to build up complex scenes
e.g. create a road on layer 0, houses on layer 1, street lamps on layer 2, ...

Rules

Auto-tiling rules are contained in a rules file (a text data-block)
Writing a rules file by hand is tedious, but it's simple once you understand it.
Sometimes a rules file can be auto-generated (see RoomGen operator)
There are plenty of examples of rules files in the **testudo3d/samples/** folder
The template rules file **testudo3d/samples/rules.txt** contains all possible bitmask permutations (64) should you ever want to define them all

Format:

```
[bitmask] [tile]
```

e.g.

```
001001 TileA
001010 TileB
110001 TileC
```

each bit of the bitmask corresponds to a direction:

```
DUWSEN
Down Up West South East North
(North is +y and East is +x)
```

```
0 = unoccipied
1 = occupied
```

e.g. in the bitmask '000001' only North is occupied
NOTE: tiles will be rotated in Z if applicable

Limitations: note that the bitmask doesn't contain information for **diagonals** or **terrain**
The 6-bit design was chosen in favor of simplicity, but it could be extended

```
default junction_white junction_red junction_yellow
000000 pipe_H
000001 pipe_H
000011 pipe_elbow_H
000101 pipe_H
000111 pipe_T_H
001111 pipe_cross_H
010000 pipe_V
010001 pipe_elbow_V_up
010101 pipe_T_H_up
100000 pipe_V
100001 pipe_elbow_V_down
100101 pipe_T_H_down
110000 pipe_V
110001 pipe_T_V
110101 pipe_cross_V
```

rules (samples/pipes.blend)

If there are multiple tiles per bitmask, a **random** tile will be chosen.
This also supports wildcards (e.g. wall_* will be expanded to wall_a, wall_b, wall_c, ...)
The **default** tile will be used when no rule is defined for a specific bitmask

Link

(3DView > T3D > Utils > Link)
Blender's linking system is very useful for keeping your work organized.
You may want to save each tileset in separate .blend files and link them into another .blend where you create your scene.

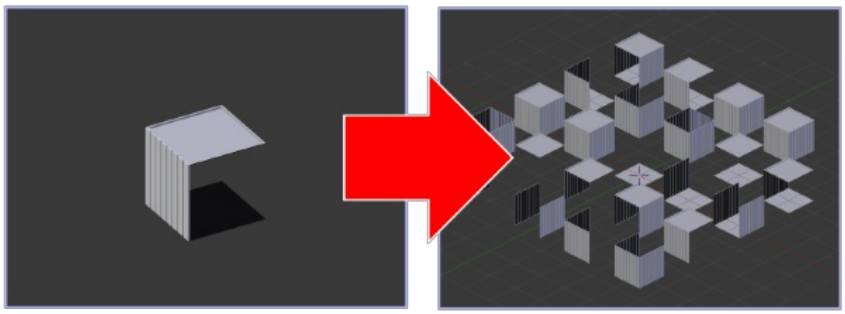
The **Link** utility will help you do this by quickly linking just the groups and text blocks from any .blend file.

Room Gen

(3DView > T3D > Utils > Room Gen)
If you want to create a building with **Walls**, **Floors** and **Ceilings**, you may be able to use **Room Gen** to generate your **tiles** and **rules**

- create 3 groups **Wall**, **Floor** and **Ceiling**
- choose the new tileset name
- press **Room Gen**

This will generate your tiles and rules. **Setup 3D Tiles** is automatically called.
The Wall should be facing north (for an example see **testudo3d/samples/house.blend**)



room gen

Make Tiles Real

(3DView > T3D > Utils > Make Tiles Real)
Since tiles are just groups, it is easy to change the tile by hand. (Properties > Object > Duplication > Group)
If your tile is itself composed of objects or groups, you may want to change it without changing all the other tiles of the same type
Blender does have the **Make Duplicates Real** operator, but unfortunately it only works recursively
Make Tiles Real will only make the first level of group instances real

Controls

Key	Action
TAB	toggle mouse paint
CURSOR KEYS	move
CTRL UP/DOWN	move up and down
CTRL LEFT/RIGHT	strafe left/right
SHIFT CURSOR KEYS	move faster
ENTER	paint
X	delete
CTRL C/V	copy/paste
G	grab
B	select region
S	sample
[	increment brush size
]	decrement brush size
SHIFT [	increment layer
SHIFT]	decrement layer
ESCAPE	escape/cancel

NOTE: controls are not configurable yet (sorry)