

[MCEdit Filters](#) >

ABrush

posted Nov 5, 2016, 10:16 PM by Adrian Brightmoore [updated Nov 19, 2016, 9:44 AM]

A framework for 3D Voxel modelling using 'brush strokes'.

The basic idea is that each brush is a procedure for placing or modifying blocks along a path.

The framework includes support for creating paths of various smoothness, adjusting the brush effect along the length of the path, and for working with a block palette.

To use:

1. A **path** is an array of (x,y,z) points. You define your path either programmatically using Python code or by placing blocks of increasing ID in the selection area.

***Programmatically:** use `path = []` and then repeated calls to `path.append((x,y,z))` to add a number of key inflection points to the path, then you can call the `brush(level,box,options,RAND,path,theBrush)` method. **level**, **box**, **options** are the same variables passed to MCEdit's `perform()` method. **RAND** is derived in `perform()` and **path** is your array of key points that will be used to find the shape of the brush stroke. **theBrush** is a string which is the method name of your brush procedure.*

2. A **brush** is a procedure that implements the following method signature:

`def yourBRUSH(level,box,options,materials,overwrite,RAND,radius,path,arclength):`

*The name '**yourBRUSH**' can be anything, but it must be used as the parameter when `brush()` is called, with the correct upper and lowercase and spelling.*

These are the fields you can work with in your brush implementation:

level - this is an MCSchematic working area.

box - this is a BoundingBox defining the region of the working area

options - these are all the filter options from the user interface

materials - this is an **array of (blockID,blockData)** tuples, one for each material on the user interface BLOCKS tab.

► **Adrian Brightmoore**

► **Builds**

► **MCEdit 2 Plugins**

► **MCEdit Filters**

► **Minecraft Mods**

► **What's Happening Now?**

Sitemap

Recent site activity

overwrite - this is a boolean **True** or **False**. If True then any blocks in the target should be overwritten, otherwise if False then this brush should not replace blocks already in the workspace MCSchematic.

RAND - this is an instance of Random created with the user-specified seed. Use **Random.randint(1,100)** to generate a random integer, for example.

radius - this is the widest radius that the brush stroke should use. In the example **line** brush you can see that at either end of the brush stroke the line is 1 block only, and in the middle it is the size of this radius. If set less than 1 the framework will choose a radius based on the selection box height.

path - this is an ordered array of points that define where the brush will execute.

arclength - this is the step-size of each block along the path and can be used to determine how many degrees along the path the brush is at each point in the path. It can be used with trigonometric functions to smoothly scale the brush stroke size along the length of the path.



| | | | |
|--|--|----------------------|-------------------|
|  ABrush_v1.py (50k) | Adrian Brightmoore, Nov 28, 2016, 11:41 PM | v.16 | ↓ |
|  ABrush_v2.py (51k) | Adrian Brightmoore, Nov 30, 2016, 12:04 AM | v.1 | ↓ |
|  ABrush_v3.py (59k) | Adrian Brightmoore, Dec 27, 2016, 9:30 PM | v.1 | ↓ |
|  ABrush_v4.py (60k) | Adrian Brightmoore, Dec 29, 2016, 7:38 PM | v.1 | ↓ |
|  ABrush_v5.py (62k) | Adrian Brightmoore, Mar 22, 2018, 7:16 AM | v.1 | ↓ |

Comments

You do not have permission to add comments.