

[MCEdit Filters](#) >

Conway's 'Life' in Minecraft via MCEdit, Raspberry Juice API, and Bukkit

posted Jun 23, 2013, 3:16 PM by Adrian Brightmoore [updated Jun 23, 2013, 8:16 PM]

I was compelled to take a look at interactive world manipulation in Minecraft using the Raspberry PI API (or rather, the port called Raspberry Juice that Martin O'Hanlon has been using recently for his programs). I used MCEdit as a harness for this because the seed image of the Life game is basically a selection of blocks that need to be updated, making MCEdit appropriate to create those blocks.

'Life' simulates a cellular automation that follows simple rules on each iteration of the world - if a block is alive and is surrounded by 2 or 3 alive blocks, it will stay alive next turn. If there are more or less blocks around it, it will die next turn due to loneliness or overcrowding. If there is a dead cell surrounded by three other alive cells, it will be re-born on the next turn. In this implementation I have wrapped the map so blocks moving off one edge rejoin at the other.

There have been some impressive Life mods for Minecraft in the past, including 3D variants. This implementation is a proof of concept to flush out issues with the Raspberry API as a tool for more interesting things.



- ▶ **Adrian Brightmoore**
- ▶ **Builds**
- ▶ **MCEdit 2 Plugins**
- ▶ **MCEdit Filters**
- ▶ **Minecraft Mods**
- ▶ **What's Happening Now?**
- Sitemap**
- Recent site activity**

How to use it

1. Start a Bukkit server with the Raspberry Juice API plug in.
2. Copy the script attached below into the MCEdit filters directory
3. In MCEdit you create a new flat scratchpad world
4. Make a 1 block high selection area, width and depth are your choice (don't go crazy though, as chunk redraw can be an issue)
5. Populate this area with non-air blocks.
6. Select the filter and run it
7. Go in-game to see the Life simulation play out.
8. At the end of the filter run, the MCEdit selection box is updated with the last frame that was generated, so you can pick up where you left off, or make changes if needed.



How it works

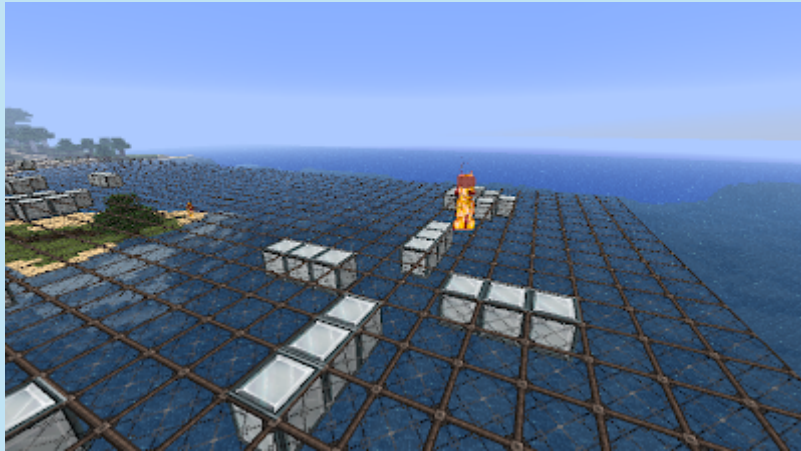
1. MCEdit handles the SEED for the canvas. You set up the initial conditions, and can save / load the Life universes to suit. With other filters, like my PNGtoBlocks, you can also bring in bit maps from historical runs of Life.
2. MCEdit calls the filter script, which has two buffers for the current Life universe, and the next one. I swap the buffers on each iteration.
3. After running the rules, I draw the blocks to Bukkit/Minecraft through the Raspberry Juice API `setBlock()` function.
4. At the end of the cycle, the final universe state is drawn back into MCEdit for you to discard, save, edit, or launch the filter again from that point.

The Pi API is dreadfully slow for reading the world blocks. As a result the filter writes to the world instead, with all the game calculation in the MCEdit memory. The Minecraft world is being used as a canvas. If you break blocks they will be re-created by the MCEdit filter on the next iteration.

Learnings

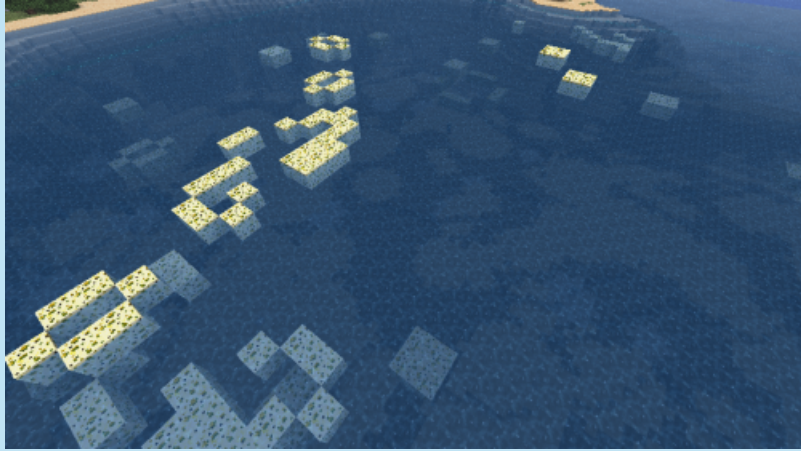
This was a proof of concept. What I take away is:

1. Using the Pi API for interactive world editing is possible.
2. More than one player in the world causes the getPlayerPosition call to hang
3. Reading blocks from the world is too slow to be reliably done in large volumes. Any implementation should take this into account.



Other Resources

- [Learn more about Conway's Game of Life on Wikipedia](#)
- [Get the CraftBukkit Minecraft server here](#)
- [Get the Raspberry Juice API for Bukkit here](#)
- Get the MCEdit filter I wrote... er... here.



Life

Conway's Life in Minecraft using MCEdit, Raspbe...



Ocean Life

Ocean Life in Minecraft



 ALIFE_GosperGliderGun.schematic (0k)	Adrian Brightmoore, Jun 23, 2013, 9:03 PM	v.1	↓
 ALIFE_LightweightSpaceshipLWSS.schematic (0k)	Adrian Brightmoore, Jun 23, 2013, 8:46 PM	v.2	↓
 ALIFE_PiCraft_v1.py (8k)	Adrian Brightmoore, Jun 23, 2013, 3:17 PM	v.1	↓
 ALIFE_PiCraft_v2.py (9k)	Adrian Brightmoore, Jun 23, 2013, 4:28 PM	v.1	↓
 ALIFE_PiCraft_v3.py (9k)	Adrian Brightmoore, Jun 23, 2013, 8:45 PM	v.1	↓
 ALIFE_Pulsar.schematic (0k)	Adrian Brightmoore, Jun 23, 2013, 8:45 PM	v.1	↓
 Turing.schematic (10k)	Adrian Brightmoore, Jun 23, 2013, 9:32 PM	v.1	↓

Comments

You do not have permission to add comments.