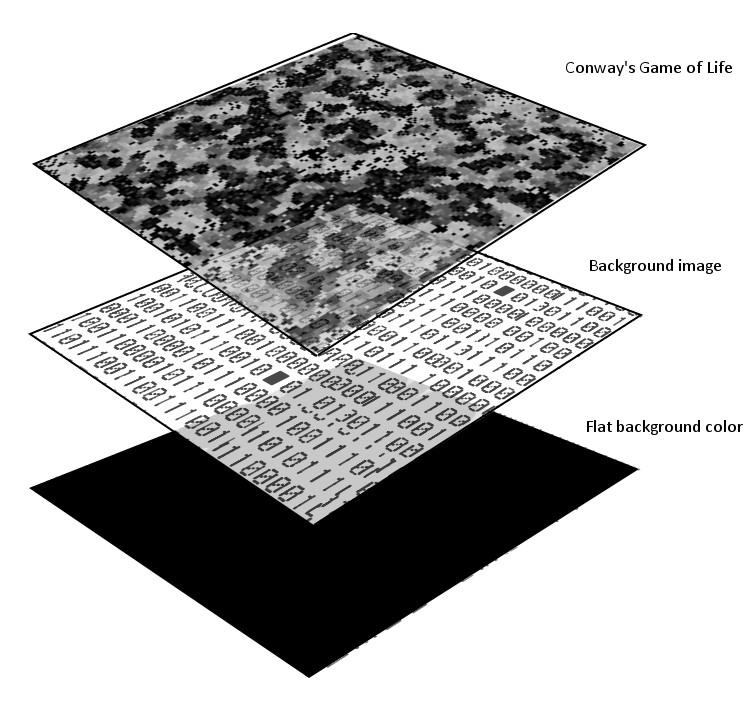
Conway’s Game of Life (hereinafter CGOL) was used in producing the background effects in Binary Wars. CGOL is a notable cellular automaton which has a simple set of rules where the state of a cell is governed entirely by the states of its neighbor cells. The rules are as follows:

* A cell in the off state that has exactly three (3) of its neighbors in the on state will turn on during the next iteration, otherwise the cell remains off.
* A cell in the on state that has either two (2) or three (3) of its neighbors also in the on state will remain on during the next iteration, other the cell turns off.

We made a fundamental change to the CGOL to allow its cells to control the opacity of its output image; we changed the contents of the cells from storing a purely binary value (on or off) to storing a full byte (0-255) that controls the alpha layer of a background image. The alpha layer of an image controls the opacity of the RGB channels of the image. Binary Wars’ CGOL understands a cell with a value of 255 as on and a cell with a value of 0 through 254 as off. When a cell live cell dies instead of pushing its value straight to zero we actually multiply its current value by a constant multiplier less than one (1) but greater than zero (0)—this is done for every cell that is off for each iteration thus giving the appearance of the dead cells fading away over time. Each of the CGOL cells are next related to a corresponding X and Y coordinate on the background texture and are then upon rendering the graphics card samples the cell’s value which is then used for the texture’s alpha layer value at that coordinate. The graphics card sampling is achieved through a custom effect file (.FX extension) we wrote; the advantage of using a custom effect is that it offloads most of the calculations and memory accesses to the graphics card rather than hogging CPU time.



All the cells of our CGOL are initially populated with the value zero (off) thus creating a black background upon starting the game. As actors are created and they move about the screen the Actor Manager tells CGOL where all of the actors are; once CGOL has this information it then turns a cell on (value 255) at every actor location. Due to the gradual fading of the off cells we are guaranteed to see a trail form behind the actors (including bullets and explosion particles) which gives the player the feeling that the background graphics are reacting to his/her actions.

Finally, we noticed that due to the particular rules of the CGOL that if left on its own certain cell patterns will rapidly become stuck in the on state which is aesthetically displeasing and can be distracting much like a stuck pixel on an LCD monitor. To solve this problem with minimal CPU and memory overhead we wrote a method that turns on a certain number of cells at random locations every frame. The random cells being turned on is not noticeable except during the start of the game where it produces a pleasing twinkling effect in the background.

SampleTexture Output

