For this assignment my main goal was to try and make an effect/tool that takes a buffer and splices it randomly to make a wall of sound. The biggest challenge was playing poked samples over each other in a poly with individualised data to reduce clicking as best as I could. Musically to create a wall of sound I wanted my patch to be able to take a buffer an splice it randomly spacing out the each fragment of sound differently. The piece I used in my example and video was one I previously wrote but wanted to a texture of fragmented sound to create a sense of tranquil disorder.

Dry signal

Output

Reverb

Gate

Peek~ (buffer in fft)

Sends amplitude data for each bandwidth 0-1

Multislider (512)

fft

Set domain

\*~ random amplitude

Metro

Decides how many and which buffers to poke into poly buffer

Combine poly(number of voices)

Voices dial

R Poly buffer number

R grain duration

Poke~ poly.1

Counter: 1 –-> (voices dial value)

\*~ function envelope

Play poly.(polybuffernumber)

Tells poly which poly buffer to playback

Set domain for envelope in voice

Poly~ voice 70

Counter 1-70

Playback bang

Grain duration

Count~ 0 2000

Bang

Replace ‘load buffer’

Start/finish/duration

random

Pack i i

Play~ new [start/finish]