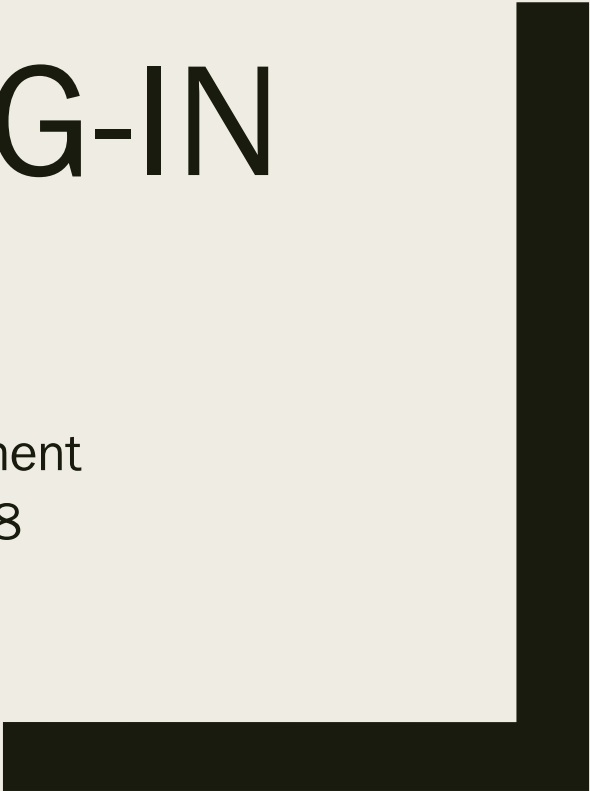# MY AUDIO PLUG-IN SUITE

6011MAPA – Audio Plug-In Development
Vladimiras Malyskinas - 10055638
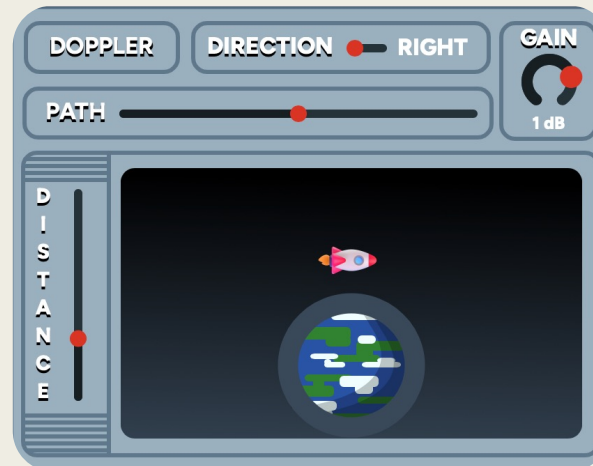
The Plug-Ins

Freeverb Reverb -
Schroeder-Moorer reverberator,
Developed by "Jezar at Dreampoint".

Doppler Effect -
Spatialization plug-in to create the
effect of space and motion.

Bit crusher -
Distortion plug-in, which reduces
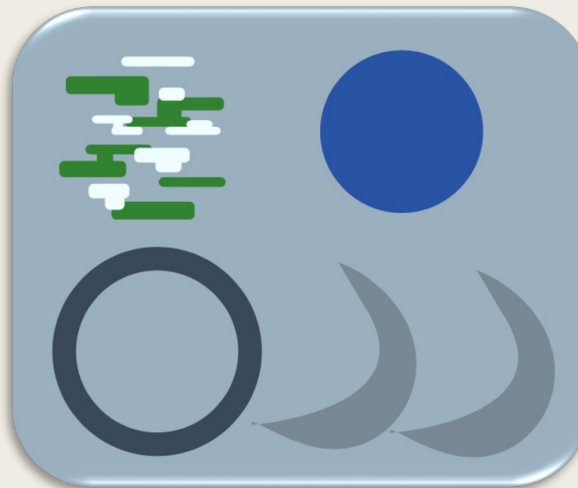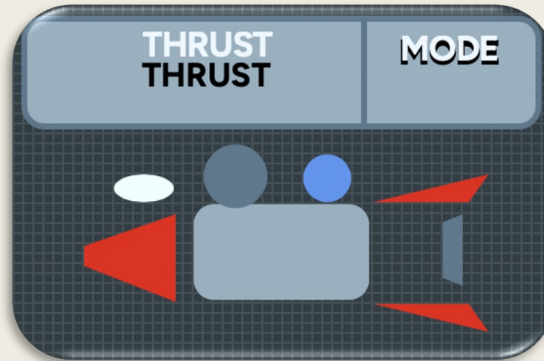the resolution of your audio signal.

A brief demonstration

The GUI

Main Design Aims:

- Cohesive
- Graphical
- Slick
- Minimalist

```
//========================================================
// Binary resources - be careful not to edit any of these sections!

// JUCER_RESOURCE: rocketL_png, 121289, "../../../../Downloads/rocketL.png"
static const unsigned char resource_NewComponent_rocketL_png[] = {
137,80,78,71,13,10,26,10,0,0,0,13,73,72,68,82,0,0,2,212,0,0,2,212,8,6,0,0,0,43,45,165,240,0
,0,0,1,115,82,71,66,0,174,206,28,233,0,0,32,
0,73,68,65,84,120,94,236,189,11,148,101,85,117,239,61,215,218,231,156,170,234,39,221,13,8,130
,47,162,128,160,224,3,163,1,141,104,76,52,38,196,68,3,49,198,107,162,73,52,137,122,243,126
,152,168,105,141,
94,99,94,55,35,15,51,134,95,238,248,242,93,191,152,68,98,110,188,38,74,212,8,2,242,80,184,62
,226,35,10,8,8,136,52,208,116,119,117,117,85,157,199,94,119,172,199,92,107,174,117,246,169
,170,102,87,87,87,
117,255,123,12,168,170,115,246,243,183,215,222,251,183,231,158,107,46,69,248,7,2,32,0,2,32,0,2
,32,0,2,32,0,2,32,240,176,9,168,135,61,39,102,4,1,16,0,1,16,0,1,16,0,1,16,0,1,16,32,8,53,26
,1,8,128,0,8,128,
0,8,128,0,8,128,0,8,180,32,0,161,110,1,15,179,130,0,8,128,0,8,128,0,8,128,0,8,128,0,132,26,109
,0,4,64,0,4,64,0,4,64,0,4,64,0,4,90,16,128,80,183,128,135,89,65,0,4,64,0,4,64,0,4,64,0,4,64
,0,66,141,54,0,
2,32,0,2,32,0,2,32,0,2,32,0,2,45,8,64,168,91,192,195,172,32,0,2,32,0,2,32,0,2,32,0,2,32,0,161
,70,27,0,1,16,0,1,16,0,1,16,0,1,16,0,129,22,4,32,212,45,224,97,86,16,0,1,16,0,1,16,0,1,16,0
,1,16,128,80,163,
13,128,0,8,128,0,8,128,0,8,128,0,8,128,64,11,2,16,234,22,240,48,43,8,128,0,8,128,0,8,128,0,8
,128,0,8,64,168,209,6,64,0,4,64,0,4,64,0,4,64,0,4,64,160,5,1,8,117,11,120,152,21,4,64,0,4
,64,0,4,64,0,4,64,0,
4,32,212,104,3,32,0,2,32,0,2,32,0,2,32,208,130,0,132,186,5,60,204,10,2,32,0,2,32,0,2,32
,0,2,32,0,2,16,106,180,1,16,0,1,16,0,1,16,0,1,16,0,1,16,104,65,0,66,221,2,30,102,5,1,16,0,1
,16,0,1,16,0,1,
16,0,1,8,53,218,0,8,128,0,8,128,0,8,128,0,8,180,32,0,161,110,1,15,179,130,0,8,128,0,8
,128,0,8,128,0,8,128,0,132,26,109,0,4,64,0,4,64,0,4,64,0,4,90,16,128,80,183,128,135
,89,65,0,4,64,0,4,
64,0,4,64,0,4,64,0,66,141,54,0,2,32,0,2,32,0,2,32,0,2,32,0,2,45,8,64,168,91,192,195,172,32,0,2
,32,0,2,32,0,2,32,0,161,70,27,0,1,16,0,1,16,0,1,16,0,129,22,4,32,212,45,224
,97,86,16,0,1,16,
0,1,16,0,1,16,0,1,16,128,80,163,13,128,0,8,128,0,8,128,0,8,128,0,8,128,64,11,2,16,234,22,240,48
,43,8,128,0,8,128,0,8,128,0,8,128,0,8,64,168,209,6,64,0,4,64,0,4,64,0,4,64,0,4,64,160,5,1,8
,117,11,120,152,
21,4,64,0,4,64,0,4,64,0,4,32,212,104,3,32,0,2,32,0,2,32,0,2,32,208,130,0,132,186
,5,60,204,10,2,32,0,2,32,0,2,32,0,2,16,106,180,1,16,0,1,16,0,1,16,0,1,16,0,1,16,104
,65,0,66,221,2,30,
102,5,1,16,0,1,16,0,1,16,0,1,8,53,218,0,8,128,0,8,128,0,8,128,0,8,180,32,0,161
,110,1,15,179,130,0,8,128,0,8,128,0,8,128,0,132,26,109,0,4,64,0,4,64,0,4,64,0,4,64
,0,4,90,16,128,80,
183,128,135,89,65,0,4,64,0,4,64,0,4,64,0,4,64,0,66,141,54,0,2,32,0,2,32,0,2,32,0,2,32,0,2,45,8
,64,168,91,192,195,172,32,0,2,32,0,2,32,0,2,32,0,161,70,27,0,1,16,0,1,16,0,1,16,0,1
```

The GUI – Motion

- You can add motion with a few simple tweaks:

```
{
    // Set the image position based on the value of the slider:
    int x = pathSlider->getValue() * 240 + 97, y = 175 - distanceSlider->getValue() * 60,
        width = 50, height = 50;
    //[UserPaintCustomArguments] Customize the painting arguments here..
    //[/UserPaintCustomArguments]
    g.setColour (juce::Colours::black);
    g.setOpacity(directionSlider->getValue() == 0 ? 1 : 0);
    g.drawImageWithin (cachedImage_rocketR_png_4,
                       x, y, width, height,
                       juce::RectanglePlacement::centred,
                       false);
}
```

- If you want to go deeper or make it work with automation:

```
class C74projucerGenPluginAudioProcessorEditor  : public AudioProcessorEditor,
public juce::Slider::Listener, private juce::Timer
```

```
void C74projucerGenPluginAudioProcessorEditor::paint (Graphics& g)
{
    //Start a timer @ 60Hz
    juce::Timer::startTimerHz (60);
```

```
void C74projucerGenPluginAudioProcessorEditor::timerCallback()
{
    //Repaint the GUI 60 times per second
    repaint();
}
```

- If you add some creative logic, you can make things move!

```
void C74projucerGenPluginAudioProcessorEditor::timerCallback()
{
    // Get the value of the thrust Slider and calculate the maximum x
    // a particle can go.
    particlesMaxX = thrustSlider->getValue() * 90 + 270;
    // Do a for loop for each rectangle in the thruster
    for (int i = 0; i < std::size(rectangles); i++){
        // If the particles didn't reach their max X position and the DAW is playing...
        if(rectangles[i].getX() < particlesMaxX && processor.isCurrentlyPlaying())
        {
            // Move the rectangles forward by a random number between 1 and 3
            rectangles[i].setX(rectangles[i].getX()+ rand() % 3 + 1);
        }
        // Else if the rectangles x position is above the maximum allowed...
        else if (rectangles[i].getX() >= particlesMaxX)
        {
            //Set the particles position to 270 (the beginning)
            rectangles[i].setX(270);
        }

    }
    repaint();
}
```
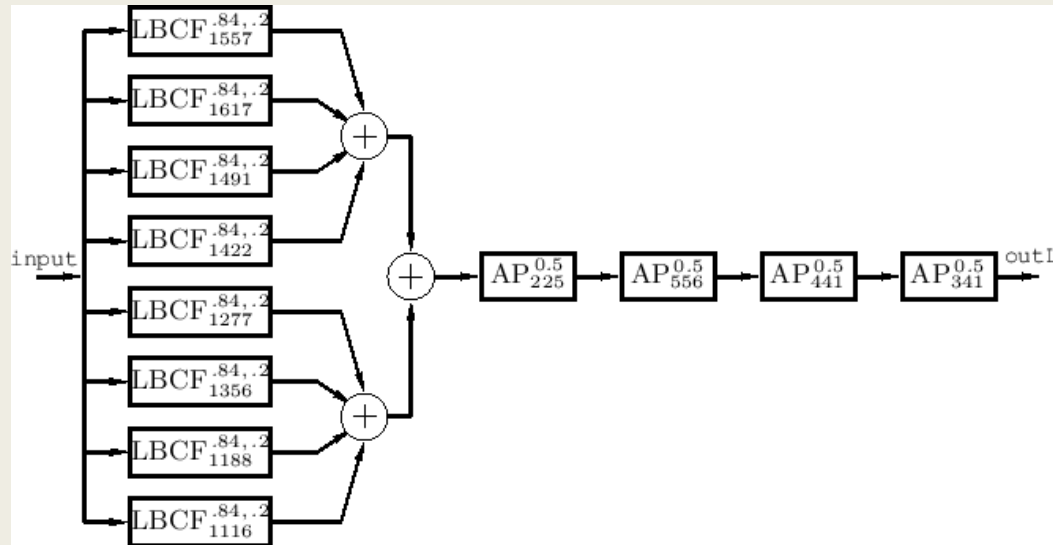
Many thanks to Adam for helping with this.

THE BACKEND – BIT CRUSHER

"DISTORTION IS CAUSED WHEN VERY LOW-LEVEL AUDIO SIGNALS ARE QUANTIZED"
- DR. FERIDOUN SHAWN SHAHMORADIAN

"QUANTIZATION IS THE PROCESS OF CONVERTING A CONTINUOUS SIGNAL INTO A DISCRETE ONE BY ASSIGNING A FINITE NUMBER OF VALUES TO REPRESENT THE AMPLITUDE OF THE SIGNAL"
- DR. LEANDRO STEFANAZZI
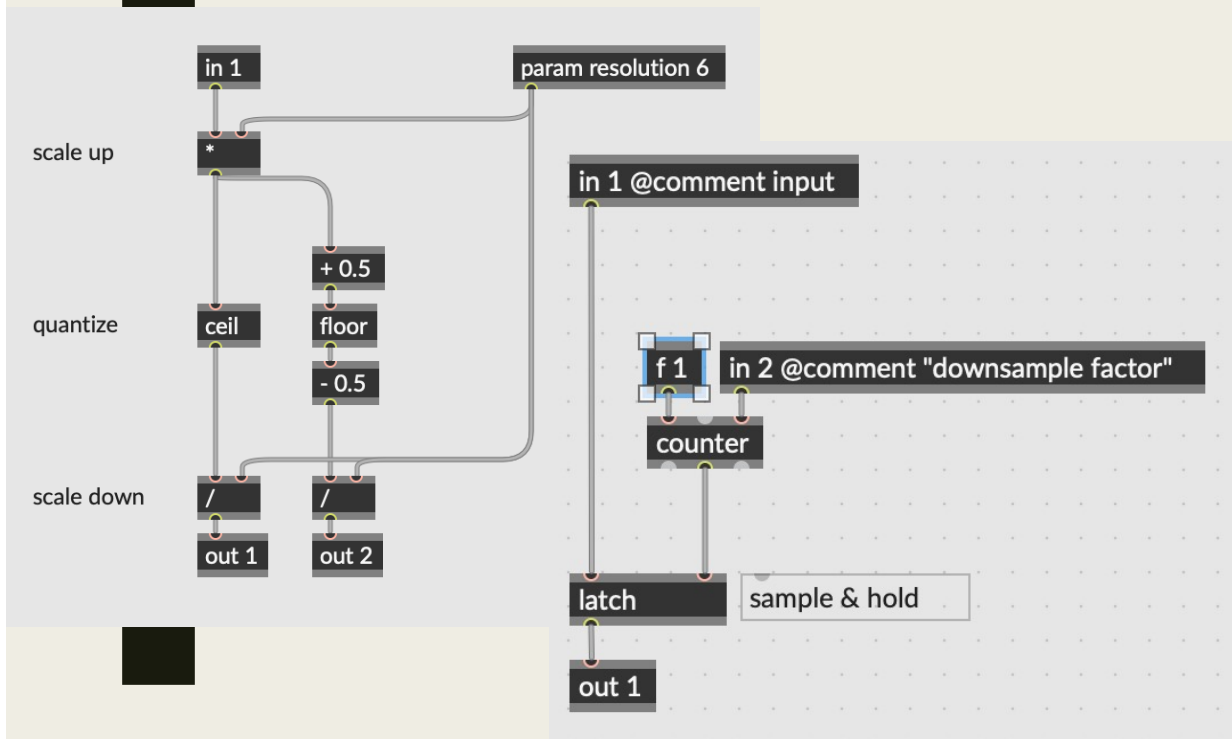
From Cycling 74 > Examples > Gen

The backend – Freeverb

FREEVERB ALLPASS APROXIMATION

Each of the four Freeverb allpass sections is really a feedback comb-filter FBCF_N^g in series with a feedforward comb-filter $\mathrm{FFCF}_N^{-1,1+g}$

LOWPASS-FEEDBACK COMB FILTER $\mathrm{LBCF}_N^{f,d}$

Lowpass scale-factor f is called feedback $\quad f \;=\; \texttt{roomsize = initialroom * scaleroom + offsetroom}$

Damping d parameter controls how rapidly T60 shortens as a function of increasing frequency.
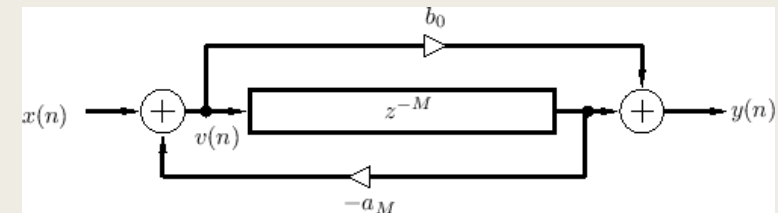
Physical Audio Signal Processing: for Virtual Musical Instruments and Digital Audio Effects by Julius O. Smith III
https://www.dsprelated.com/freebooks/pasp

The backend – Doppler Effect

NOT a gen~ patch...
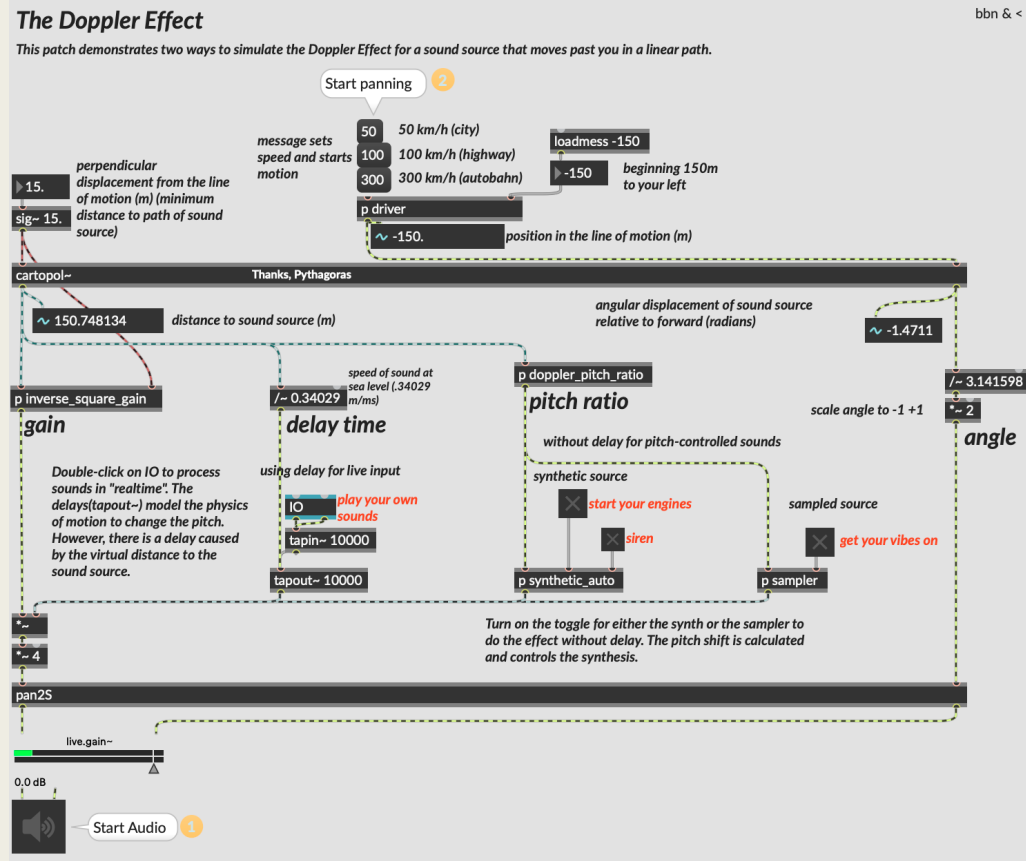
Things that needed to be translated:
- Pitch shift
- tapin~, tapout~
- Panning

From Cycling 74 > Examples > Spatialization

Reflection – if I were to do this further

The good -

- GUI Editor
- Max as a tool for DSP
- Amount of research material
- Potential in the industry

Needs improvement -

- Easier/more animations
- Streamlined way to create sliders
- ProJucer auto generating the code

From gearspace.com:

Thank you,
time to discuss.