```
Render {
  RenderPipeline(vertexShader: vertexShader, fragmentShader: fragmentShader) {
    Draw { encoder in
      encoder.setVertexBytes(<...>)
      encoder.drawPrimitives(type: .triangle, vertexStart: 0, vertexCount: 3)
    }
    .parameter("color", SIMD4<Float>([1, 0, 0, 1]))
  }
}
```

Shows how Utraviolence code dramatically simplifies a lot of Metal setup and boilerplate code while retaining flexibility.
The real benefit of Ultraviolence is in the ease of composing complex render graphs from simpler components in the same way SwiftUI allows composible user interfaces.
For the purposes of this diagram the “vanilla” Metal code is shown out of order