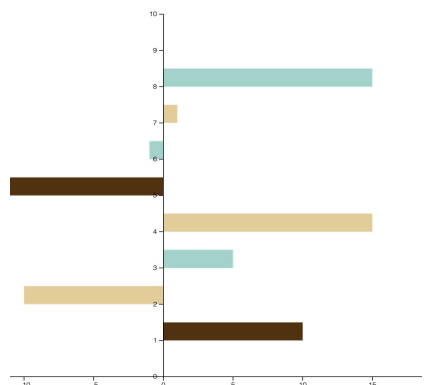
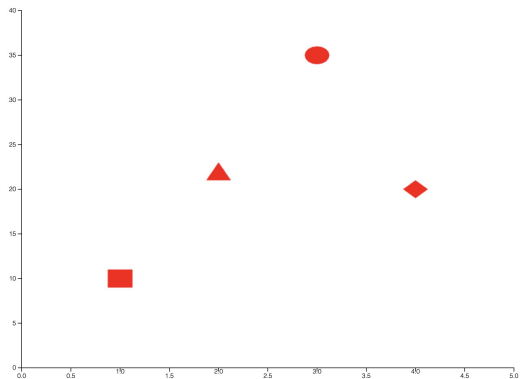
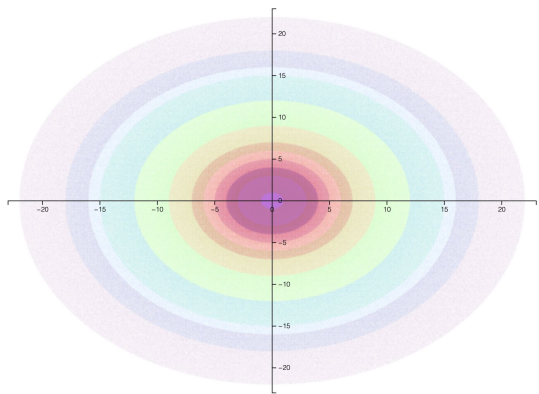
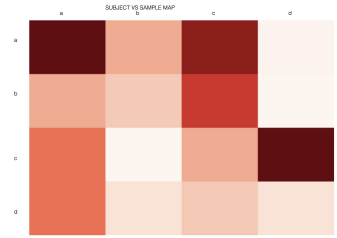
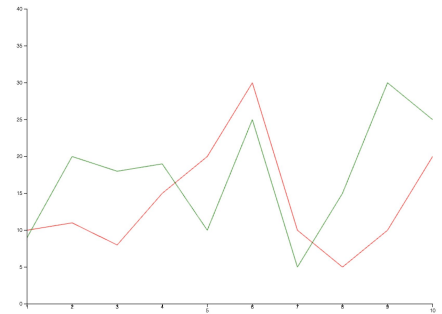
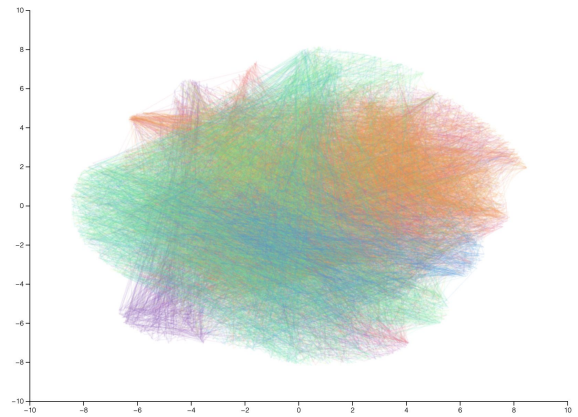
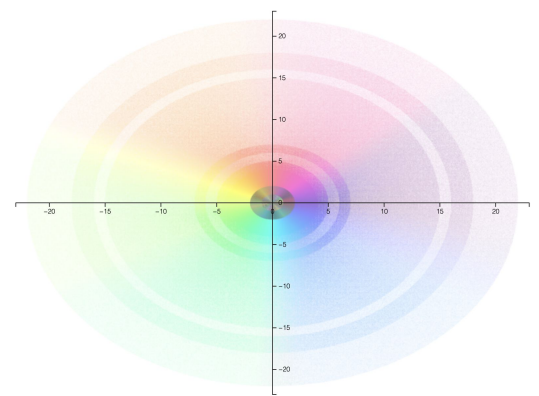

Epiviz.gl: A declarative approach to high-performance visualizations

Presentation by: Sam Rosen



10 Million Points!
 (~10 fps zoomed out, 60 fps zoomed in)

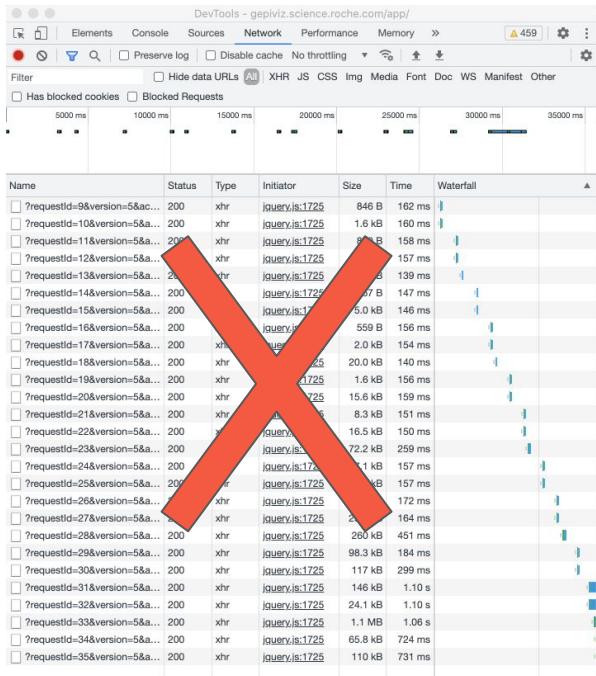
All 60 fps



Motivation - Requirements

Simple Summary: Ease-of-use for visualizing (genomic) data

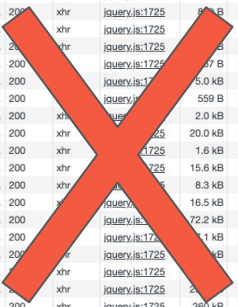
Avoid use of a backend



Motivation - Requirements

Simple Summary: Ease-of-use for visualizing (genomic) data

Avoid use of a backend



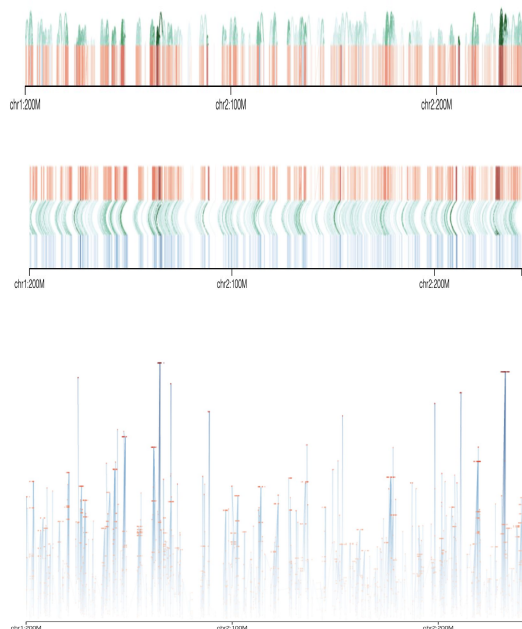
DevTools - gepiviz.science.roche.com/app/

Filter: ☐ Hide data URLs ☒ All XHR JS CSS Img Media Font Doc WS Manifest Other

☐ Has blocked cookies ☐ Blocked Requests

Name	Status	Type	Initiator	Size	Time	Waterfall
<input type="checkbox"/> ?requestId=9&version=5&ac...	200	xhr	jquery.js:1725	846 B	162 ms	
<input type="checkbox"/> ?requestId=10&version=5&ac...	200	xhr	jquery.js:1725	1.6 kB	180 ms	
<input type="checkbox"/> ?requestId=11&version=5&ac...	200	xhr	jquery.js:1725	978 B	158 ms	
<input type="checkbox"/> ?requestId=12&version=5&ac...	200	xhr	jquery.js:1725	978 B	157 ms	
<input type="checkbox"/> ?requestId=13&version=5&ac...	200	xhr	jquery.js:1725	978 B	139 ms	
<input type="checkbox"/> ?requestId=14&version=5&ac...	200	xhr	jquery.js:1725	978 B	147 ms	
<input type="checkbox"/> ?requestId=15&version=5&ac...	200	xhr	jquery.js:1725	5.0 kB	146 ms	
<input type="checkbox"/> ?requestId=16&version=5&ac...	200	xhr	jquery.js:1725	559 B	156 ms	
<input type="checkbox"/> ?requestId=17&version=5&ac...	200	xhr	jquery.js:1725	2.0 kB	154 ms	
<input type="checkbox"/> ?requestId=18&version=5&ac...	200	xhr	jquery.js:1725	20.0 kB	140 ms	
<input type="checkbox"/> ?requestId=19&version=5&ac...	200	xhr	jquery.js:1725	1.6 kB	156 ms	
<input type="checkbox"/> ?requestId=20&version=5&ac...	200	xhr	jquery.js:1725	15.6 kB	159 ms	
<input type="checkbox"/> ?requestId=21&version=5&ac...	200	xhr	jquery.js:1725	8.3 kB	151 ms	
<input type="checkbox"/> ?requestId=22&version=5&ac...	200	xhr	jquery.js:1725	16.5 kB	150 ms	
<input type="checkbox"/> ?requestId=23&version=5&ac...	200	xhr	jquery.js:1725	72.2 kB	259 ms	
<input type="checkbox"/> ?requestId=24&version=5&ac...	200	xhr	jquery.js:1725	7.1 kB	157 ms	
<input type="checkbox"/> ?requestId=25&version=5&ac...	200	xhr	jquery.js:1725	7.1 kB	157 ms	
<input type="checkbox"/> ?requestId=26&version=5&ac...	200	xhr	jquery.js:1725	172 ms		
<input type="checkbox"/> ?requestId=27&version=5&ac...	200	xhr	jquery.js:1725	2.0 kB	164 ms	
<input type="checkbox"/> ?requestId=28&version=5&ac...	200	xhr	jquery.js:1725	260 kB	451 ms	
<input type="checkbox"/> ?requestId=29&version=5&ac...	200	xhr	jquery.js:1725	98.3 kB	184 ms	
<input type="checkbox"/> ?requestId=30&version=5&ac...	200	xhr	jquery.js:1725	117 kB	299 ms	
<input type="checkbox"/> ?requestId=31&version=5&ac...	200	xhr	jquery.js:1725	146 kB	1.10 s	
<input type="checkbox"/> ?requestId=32&version=5&ac...	200	xhr	jquery.js:1725	24.1 kB	1.10 s	
<input type="checkbox"/> ?requestId=33&version=5&ac...	200	xhr	jquery.js:1725	1.1 MB	1.06 s	
<input type="checkbox"/> ?requestId=34&version=5&ac...	200	xhr	jquery.js:1725	65.8 kB	724 ms	
<input type="checkbox"/> ?requestId=35&version=5&ac...	200	xhr	jquery.js:1725	110 kB	731 ms	

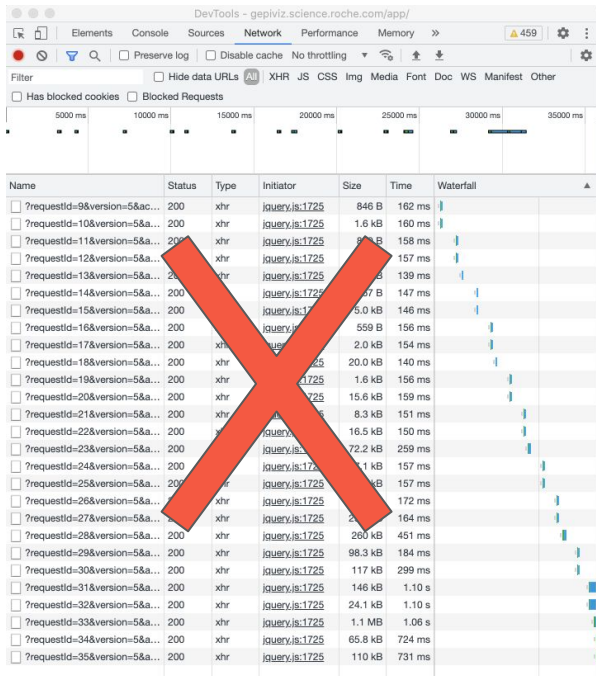
Declarative for flexibility
and user choices



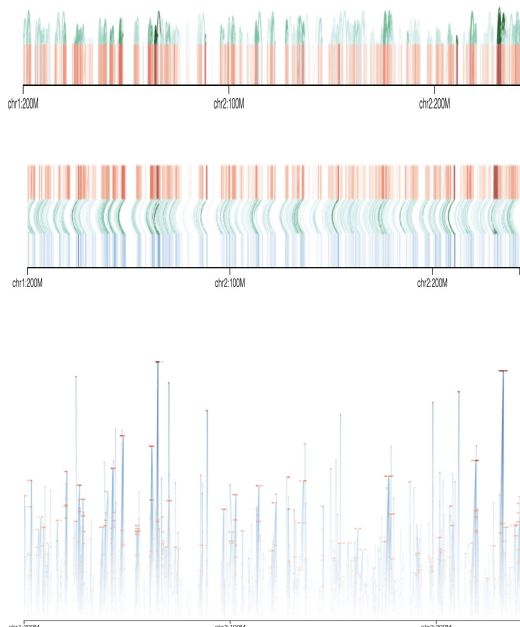
Motivation - Requirements

Simple Summary: Ease-of-use for visualizing (genomic) data

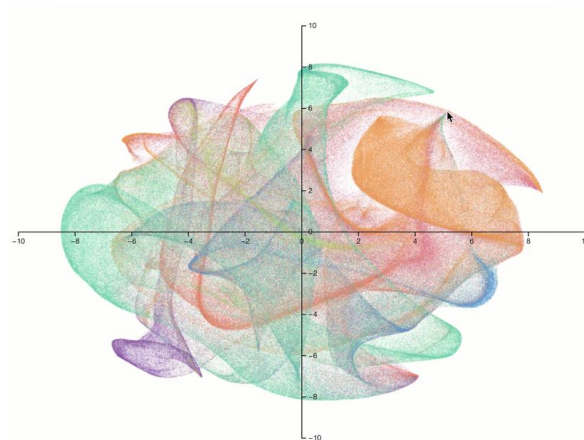
Avoid use of a backend



Declarative for flexibility
and user choices



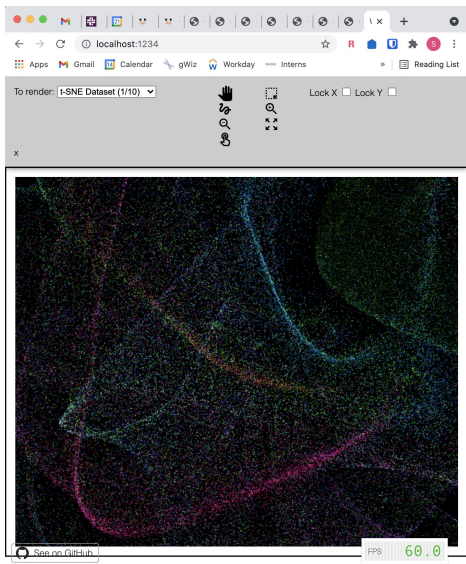
Enable visualization of large
amounts of data with precision



Motivation - Technologies

Web Workers and OffScreenCanvas

- Rendering on own thread, avoids UI processing
- Keeps application responsive



Main Thread

Creates worker

Transfers canvas to
worker

Web Worker

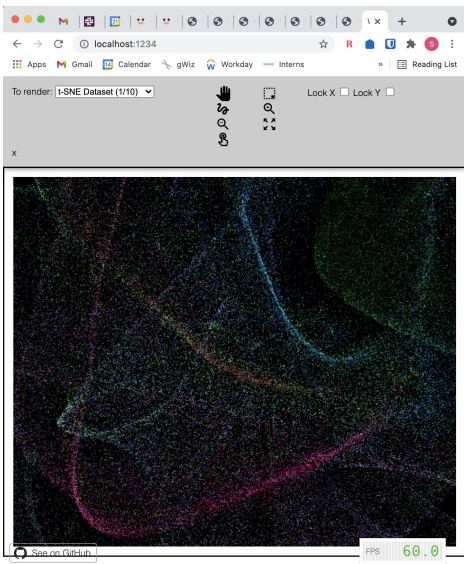
No access to DOM
Independent namespace
Runs on own thread

Main thread and
Worker can post
messages to each
other with basic
JavaScript objects
copied

Motivation - Technologies

Web Workers and OffScreenCanvas

- Rendering on own thread, avoids UI processing
- Keeps application responsive



Main Thread
Creates worker
Transfers canvas to
worker



Web Worker
No access to DOM
Independent namespace
Runs on own thread

Main thread and
Worker can post
messages to each
other with basic
JavaScript objects
copied

WebAssembly (WASM)

- Allows native performance for compiled JavaScript
- Not yet used in epiviz.gl, but would make a noticeable difference

```
1  ;; INFO asc module.ts --textFile module.wat --binaryFile module.wasm -O3 --runtime stub
2  (module
3    (type $i32_=>i32 (func (param i32) (result i32)))
4    (memory $0 0)
5    (export "fib" (func $module/fib))
6    (export "memory" (memory $0))
7    (func $module/fib (param $0 i32) (result i32)
8      (local $1 i32)
9      (local $2 i32)
10     (local $3 i32)
11     i32.const 1
12     local.set $1
13     local.get $0
14     i32.const 0
15     i32.gt.s
16     if
17       loop $while-continue|0
18         local.get $0
19         i32.const 1
20         i32.sub
21         local.set $0
22       if
23         local.get $1
24         local.get $2
25         i32.add
26         local.set $1
27         local.set $2
28     )
```

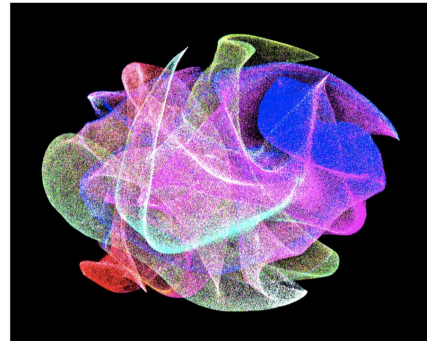
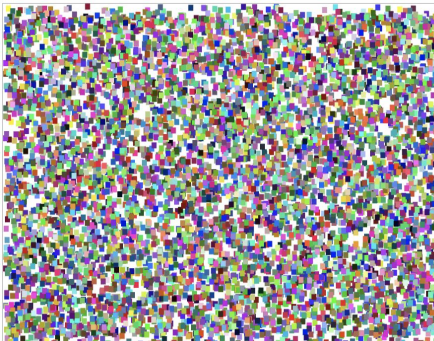
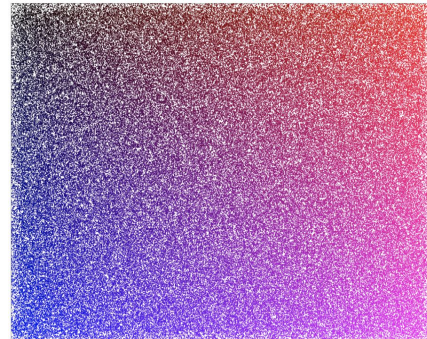
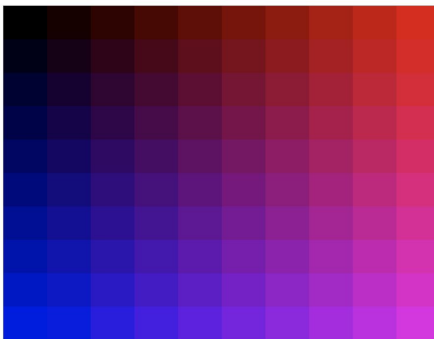
Credit: assemblyscript.org

Initial Investigation

Live demo: <https://samgrosen.github.io/offscreen-canvas-scrolling/>

Framework	Interactive	Offscreen	Render all test sets
WebGL	Yes	Yes	Yes
Base Canvas	Yes	Yes	Somewhat
PIXI.js	Yes	With polyfill	No

- Also investigated **THREE.js** and **Babylon.js**, but both were deemed insufficient before final tests
- PIXI.js is used in Gosling.js, another WebGL accelerated declarative graphics framework



Test sets used: Square gradient, jittered gradient, random points, and 1.5M points

Why WebGL?

- Offset **most** calculations (rasterization, clipping, viewport) to GPU
- Use of WebGL only has less memory overhead than popular libraries
- [twgl.js](#) makes writing WebGL much less verbose
- Ultimately **very good** at rendering millions of **simple shapes** very fast

What's the catch?

- Write shader code (hard)
- Limit transfer of data to GPU (slow)
- Special consideration for complex shapes (outside libraries)
- Build vertices/buffers in JavaScript (bottleneck)

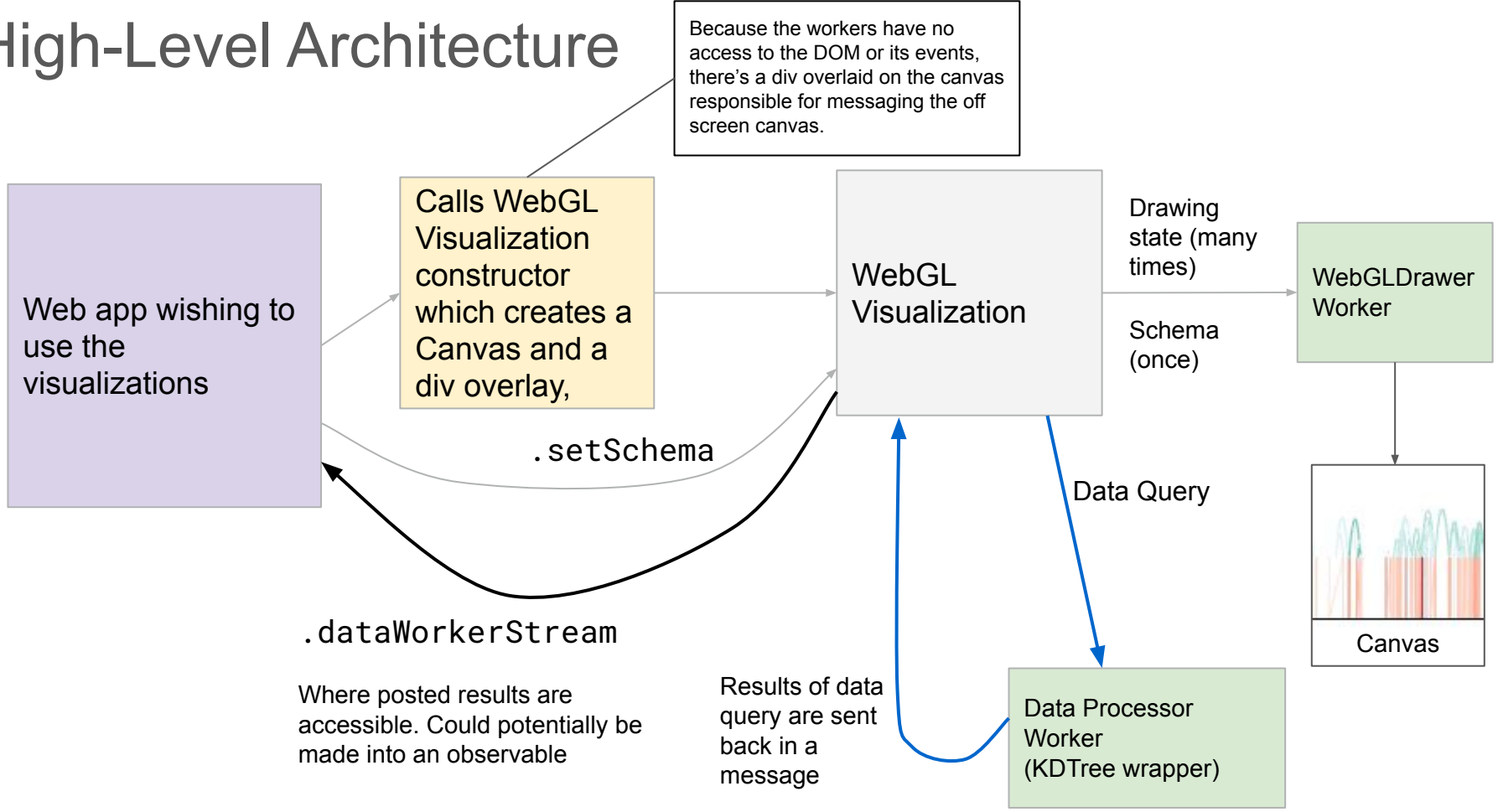
GPU



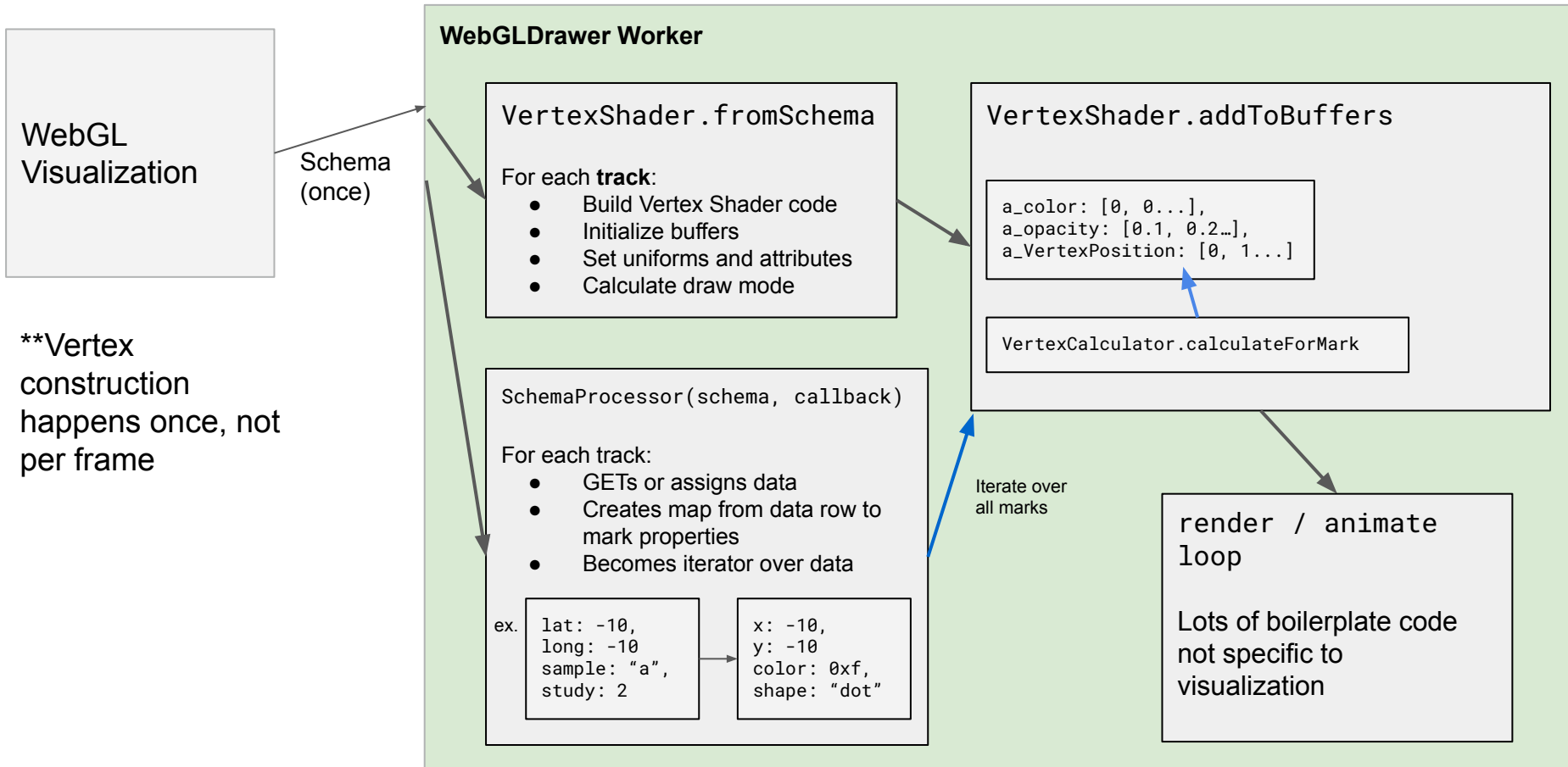
- 8GB 256-Bit GDDR6X
- Boost Clock 1830 MHz
- 1 x HDMI 2.1 3 x DisplayPort 1.4a
- 6144 CUDA Cores

- Specialized computing
- Runs very few programs at time
- Limited API
- Built with graphics as priority

High-Level Architecture



Declaration to Visualization Pipeline

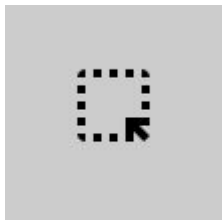


Current Tools



Panning

MouseReader



Box select

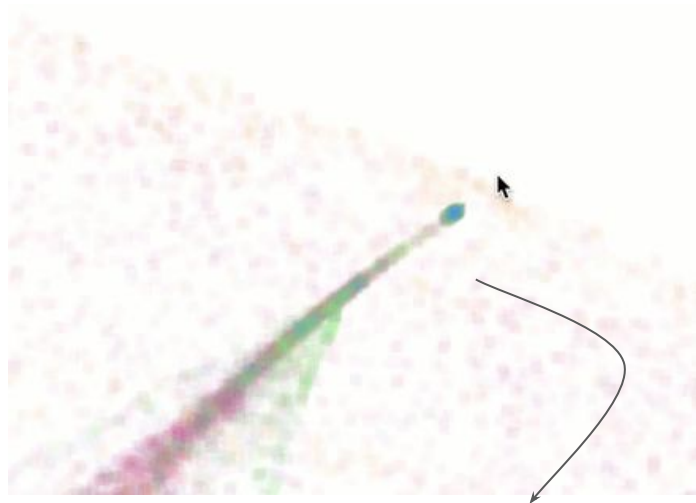
- Supercluster



Lasso select

- Supercluster
- simplify.js
- turf.js

DataProcessor
Web Worker

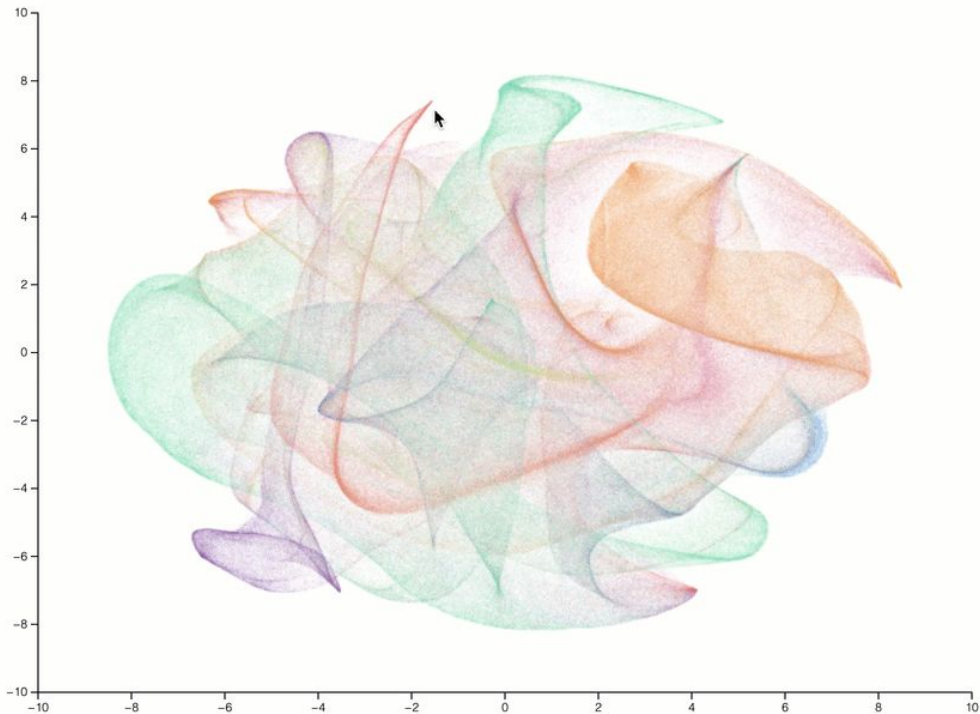


```
▼ 14: MessageEvent
  bubbles: false
  cancelBubble: false
  cancelable: false
  composed: false
  ▶ currentTarget: Worker {onerror: null, onmessage: f}
  ▼ data:
    ▶ bounds: (622) [5.197144959736991, 5.819957673843858, 5.197144959736991, 5.819957673843858]
    ▼ selection: Array(182)
      ▼ [0 ... 99]
        ▶ 0: {geometry: {...}, sample: "R", x: "5.17818", y: "5.81358"}
        ▶ 1: {geometry: {...}, sample: "E", x: "5.17905", y: "5.82112"}
```

Scatter Plot

```
{
  "xAxis": "center",
  "yAxis": "center",
  "defaultData":
  "http://localhost:1234/tsne.a6dfdc6.csv"
,
  "tracks": [
    {
      "mark": "point",
      "x": {
        "attribute": "x",
        "type": "quantitative",
        "domain": [-10,10],
      },
      "y": {
        "attribute": "y",
        "type": "quantitative",
        "domain": [-10,10],
      },
      "color": {
        "attribute": "sample",
        "type": "categorical",
        "cardinality": 32,
        "colorScheme":
        "interpolateRainbow"
      },
      "opacity": {
        "value": 0.05
      }
    }
  ]
}
```

~30 fps zoomed out, 60 fps otherwise

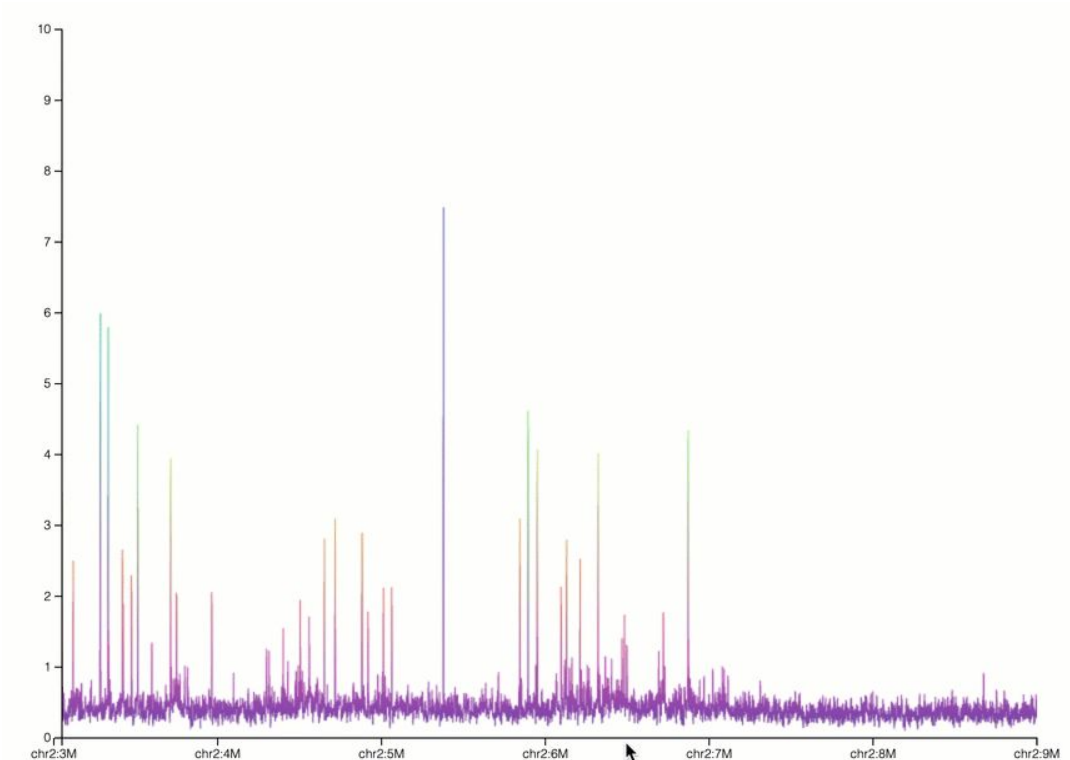


~1.5 million points

Line Track

60 fps

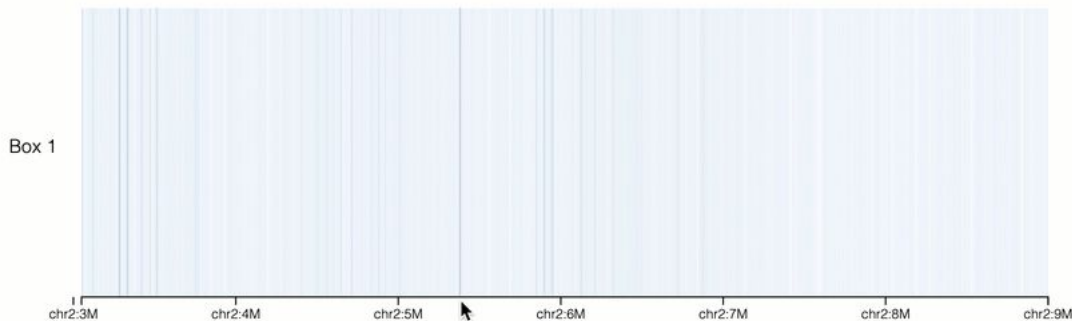
```
{
  "defaultData":
    "http://localhost:1234/box-track.4acbc46c.csv"
,
  "tracks": [
    {
      "mark": "line",
      "x": {
        "type": "genomic",
        "chrAttribute": "chr",
        "geneAttribute": "start",
        "domain": [
          "chr2:3049800",
          "chr2:9001000"
        ],
        "genome": "hg38"
      },
      "y": {
        "type": "quantitative",
        "attribute": "score",
        "domain": [0, 10]
      },
      "color": {
        "type": "quantitative",
        "attribute": "score",
        "domain": [0, 8],
        "colorScheme": "interpolateBlues"
      }
    }
  ]
}
```



Box Track

60 fps

```
{
  "margins": {"left": "10%"},
  "labels": [
    {
      "y": 0.05,
      "x": -1.1,
      "text": "Box 1",
      "fixedX": true
    }
  ],
  "xAxis": "zero",
  "yAxis": "none",
  "defaultData":
    "http://localhost:1234/box-track.4acbc46c.csv",
  "tracks": [
    {
      "mark": "rect",
      "x": {
        "type": "genomicRange",
        "chrAttribute": "chr",
        "startAttribute": "start",
        "endAttribute": "end",
        "domain": [
          "chr2:3049800",
          "chr2:9001000"
        ],
        "genome": "hg38"
      },
      "y": {
        "value": 0
      },
      "height": {
        "value": 10
      },
      "color": {
        "type": "quantitative",
        "attribute": "score",
        "domain": [0, 8],
        "colorScheme": "interpolateBlues"
      }
    }
  ]
}
```

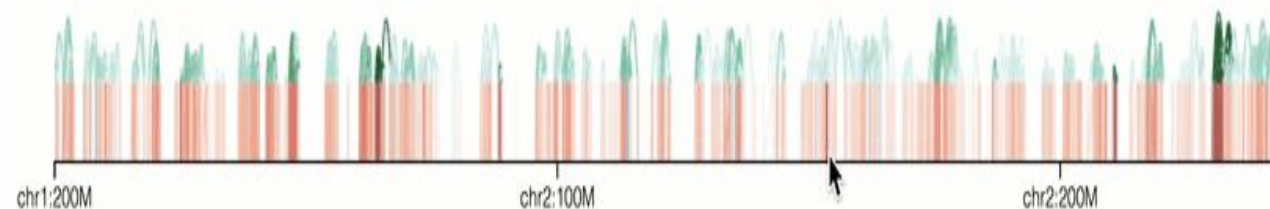


Arc Track

```
{
  "mark": "arc",
  "x": {
    "type": "genomicRange",
    "chrAttribute": "region1Chrom",
    "startAttribute": "region1Start",
    "endAttribute": "region1End",
    "domain": [
      "chr2:38000",
      "chr2:243149000"
    ],
    "genome": "hg19"
  },
  "width": {
    "type": "genomicRange",
    "chrAttribute": "region2Chrom",
    "startAttribute": "region2Start",
    "endAttribute": "region2End",
    "domain": [
      "chr2:38000",
      "chr2:243149000"
    ],
    "genome": "hg19"
  },
  "y": {
    "value": 0.1
  },
  "height": {
    "value": 0
  },
  "color": {
    "type": "quantitative",
    "attribute": "value",
    "domain": [0, 60],
    "colorScheme": "interpolateBuGn"
  }
}
```

~242 million base pairs in the 2nd chromosome

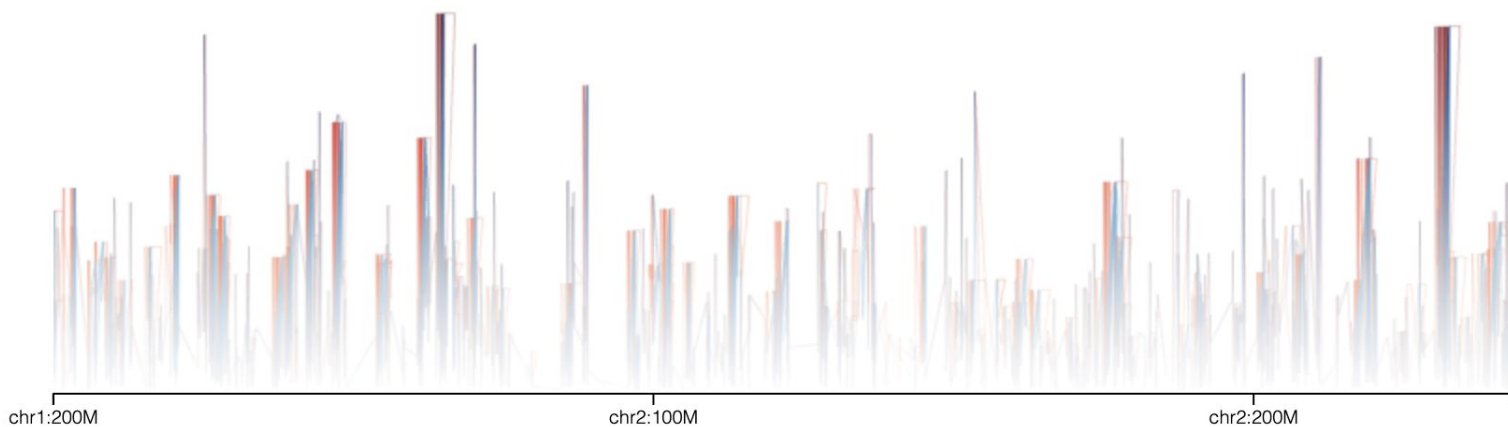
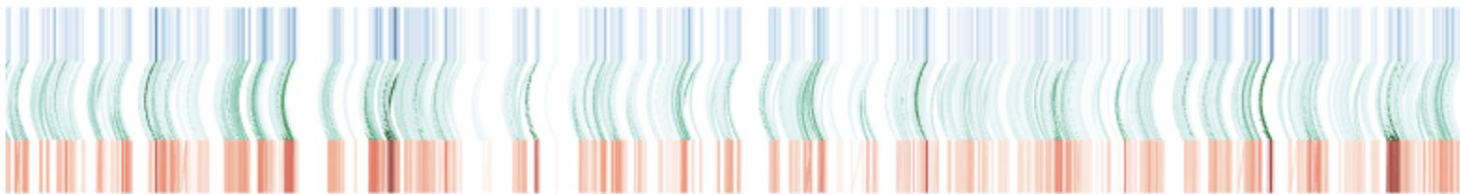
60 fps



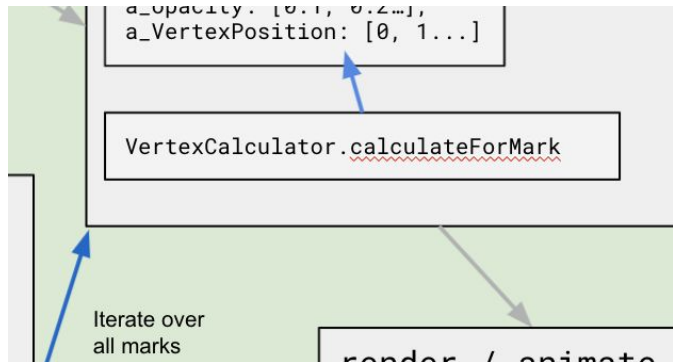
~400 base pairs in this region

Put them all together!

60 fps



Future Features



WASM to speedup vertex generation

```
"x": {  
  "attribute" : "x",  
  "type": "quantitative"  
  "domain": [0, 100000]  
  "map": "log(x) ** 2",  
  "filters": ["x < 10"]  
}
```

Inline declaration functions

- Tooltips
- Legends
- Deployment as component
- Semantic Zooming
- Increasing chrome memory?
- Whatever your heart desires! (almost)
- Make a GitHub issue!