

EXT.GRAPH.VISUALEDITOR

Graph editing in VisualEditor

WHO'S THIS GUY?!

- Frédéric Bolduc *aka ferdbold*
- Google Summer of Code 2015 student
- Senior undergraduate at
Université du Québec à Chicoutimi
in video game development

THE GRAPH EXTENSION

- Enables graphs, maps and other visualizations powered by Vega.js
- Takes data specifications in JSON and renders them in HTML5 Canvas or SVG
- Works with Graphoid service to serve static images for older browser support

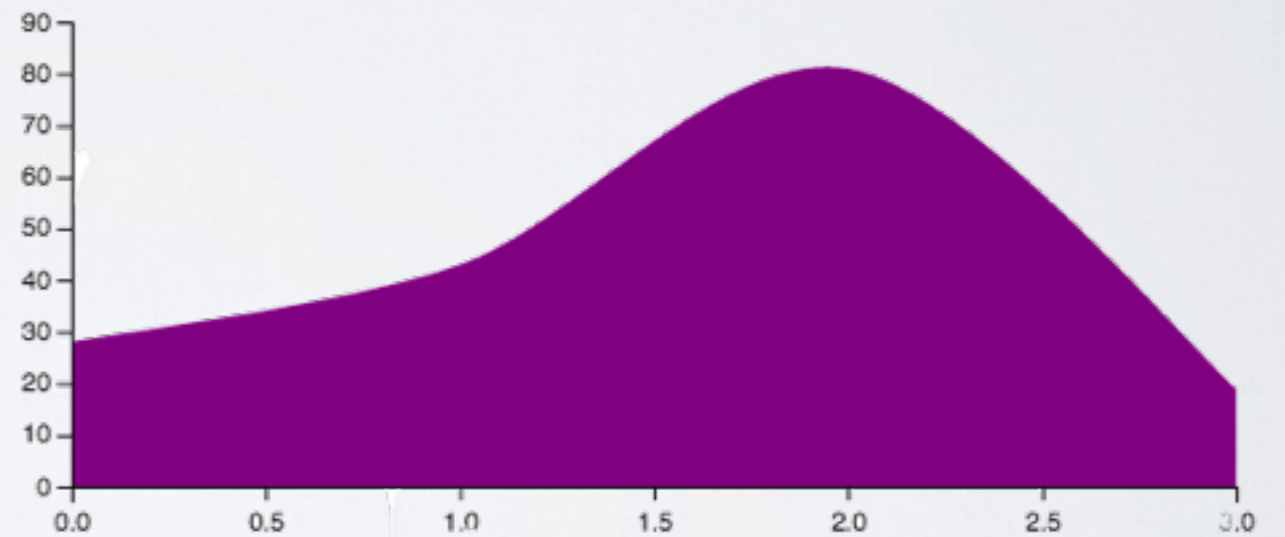
THE PROBLEM

- Vega gives total control (i.e.: too much control?) over the output and styling
- Even basic specifications still remain terrifying things for the average user
- Who wants to edit source anyway?

THE PROBLEM

- 95 lines of JSON for this simple graph!

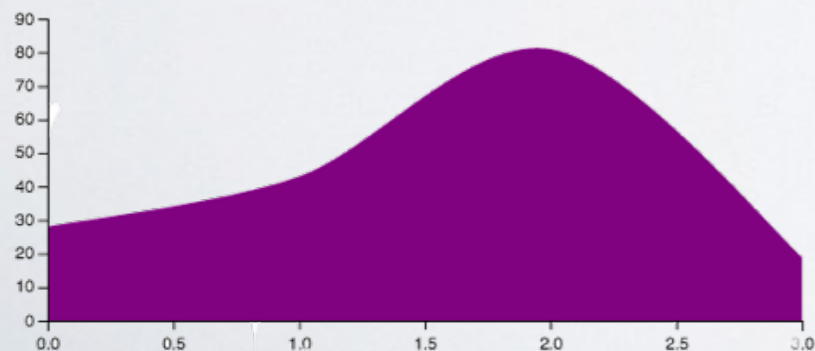
```
<graph>{
  "width": 500,
  "height": 200,
  "padding": {
    "top": 10,
    "left": 30,
    "bottom": 30,
    "right": 10
  },
  "data": [
    {
      "name": "table",
      "values": [
        { "x": 0, "y": 28 },
        { "x": 1, "y": 43 },
        { "x": 2, "y": 81 },
        { "x": 3, "y": 19 }
      ]
    }
  ],
  "scales": [
    {
      "name": "x",
      "type": "linear",
      "range": "width",
      "zero": false,
      "domain": {
        "data": "table",
        "field": "data.x"
      }
    },
    {
      "name": "y",
      "type": "linear",
      "range": "height",
      "nice": true,
      "domain": {
        "data": "table",
        "field": "data.y"
      }
    }
  ],
  "axes": [
    {
      "type": "x",
      "scale": "x"
    },
    {
      "type": "y",
      "scale": "y"
    }
  ],
  "marks": [
    {
      "type": "area",
      "from": {
        "data": "table"
      },
      "properties": {
        "enter": {
          "interpolate": {
            "value": "monotone"
          },
          "x": {
            "scale": "x",
            "field": "data.x"
          },
          "y": {
            "scale": "y",
            "field": "data.y"
          },
          "y2": {
            "scale": "y",
            "value": 0
          },
          "fill": {
            "value": "purple"
          }
        }
      }
    }
  ]
}
```



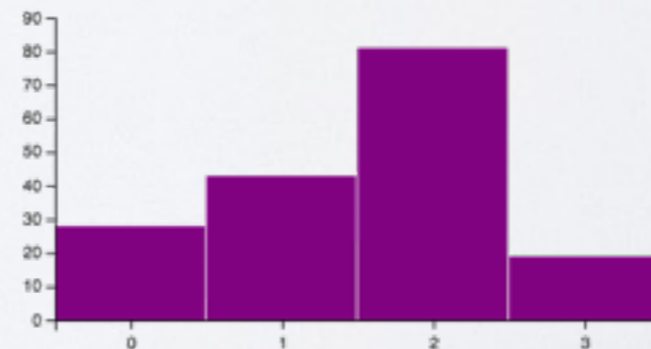
THE PROBLEM

- Basic operations aren't as straight-forward as they seem, even for developers
ex. : Switch from a line graph to a bar graph

```
marks[0].type="area"  
marks[0].properties.enter = {  
  interpolate: { value: "monotone" }  
  width: undefined  
}  
scale.type = "linear"
```



```
marks[0].type="rect"  
marks[0].properties.enter = {  
  interpolate: undefined  
  width: {  
    scale: "x",  
    band: true,  
    offset: -1  
  }  
}  
scale.type = "ordinal"
```



THE MODULE

- Conditionally loads if VE is detected on the installation
- Renders Vega graphs as GraphNodes within VE
- Adds graph editing dialog in VE

THE MODULE

- Easily switch between area, bar and line graph
- (soon!) Edit padding around the graph

- Edit graph data, add and delete rows of data
- Custom-made TableWidget, potentially usable in OOjs-UI core?

The screenshot shows the 'Edit graph' dialog box with the 'General' tab selected. The dialog has 'Cancel' and 'Apply' buttons. The 'General' tab contains the following fields:

- Name:
- Graph type:
- Padding:
 - Top:
 - Right:
 - Bottom:
 - Left:

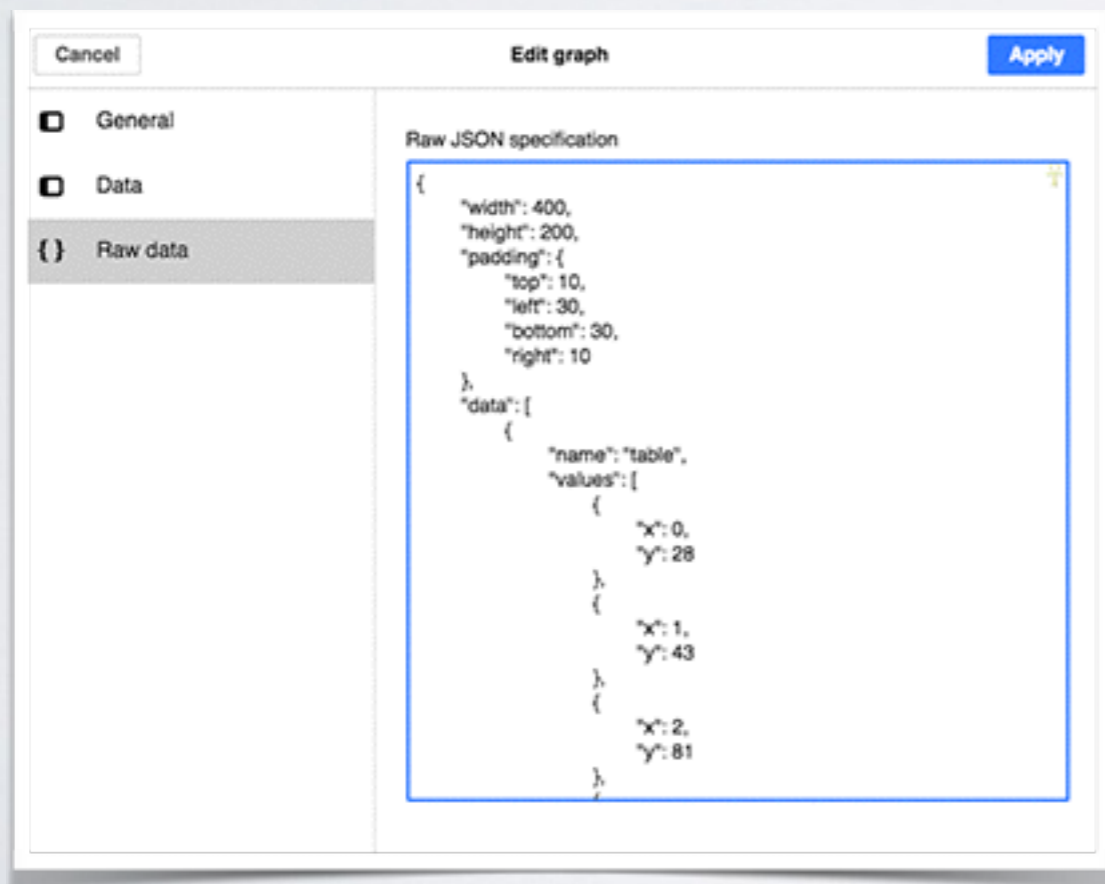
The screenshot shows the 'Edit graph' dialog box with the 'Data' tab selected. The dialog has 'Cancel' and 'Apply' buttons. The 'Data' tab contains a table with the following data:

x	y	
0	28	
1	43	
2	81	
3	19	

THE MODULE

- Raw mode displays the original spec
- Syncs automatically with changes made elsewhere in the dialog
- Will remain in the dialog until all desired features are implemented in the dialog

- What's next? Who knows :)



Thanks :)
(question time)

Frédéric Bolduc
twitter.com/itsferdbold | github.com/ferdbold