Integrating Plotly charts into Firefly visualization system

Tatiana Goldina
IPAC/Caltech
Other tools with connected views

Brushing & Linking
propagating selections in any graph to all others

Supporting tools

gluviz.org
uwdata/falcon
Bokeh
Bokeh
Plotly Dash
...

Image Credit: gluviz.org
Tools that tightly integrate data access, versioning, and management are preferred over those that put all of the data management steps purely outside of the Information Visualization system.

Dan Russell
Google research scientist
Common charts

Must support

- Scatter charts;
- Error/limits
- Line plots
- Histograms
- Density plots

Bonus: other charts
Ideal chart tech requirements

- Visually attractive
- Interactive (dynamic tooltips, selections)
- Customizable (custom actions and looks)
- Responsive (no lag, no flicker)
- Executed in a browser (SVG, <CANVAS>, WebGL)
- Events monitoring and custom code hooks
- Easy mapping of our data to chart data
- Optimizations for iterative updates
Firefly Chart API

Do not limit API users to common charts

Allow any chart or attribute supported by the charting library

How do we maintain and document these charts?
First scientific JS charting library on the web
Open sourced 2015
Declarative charts in JSON
Language and platform independent
Supported schema translations into

- JavaScript
- Python
- R
- MATLAB

License
Declarative Chart Example

```javascript
chart = {
  data: [
    {
      name: 'Female',
      mode: 'markers',
      marker: {
        color: 'rgba(223, 83, 1)',
        x: [161.2, 167.5, 159.5, 1],
        y: [51.6, 59.0, 49.2, 63.0],
      },
    },
    {
      name: 'Male',
      mode: 'markers',
      marker: {
        color: 'rgba(119, 152, 1)',
        x: [174.0, 175.3, 193.5, 1],
        y: [65.6, 71.8, 80.7, 72.6],
      },
    },
  ],
  layout: {
    title: 'Height Versus Weight of 507 Individuals by Gender',
    xaxis: {title: 'Height (cm)'},
    yaxis: {title: 'Weight (kg)'}
  }
};
```
Declarative Chart Example

```javascript
{
  series: [{
    name: 'Female',
    color: 'rgba(223, 83, 83, .5)',
    data: [[161.2, 51.6], [167.5, 59.0]],
  }, {
    name: 'Male',
    color: 'rgba(119, 152, 191, .5)',
    data: [[174.0, 65.6], [175.3, 71.8]],
  }],
  chart: {
    type: 'scatter',
    zoomType: 'xy'
  },
  title: {
    text: 'Height Versus Weight of 507 Individuals by Gender'
  },
  xAxis: {
    title: {
      text: 'Height (cm)'
    }
  },
  yAxis: {
    title: {
      text: 'Weight (kg)'
    }
  }
}
```

Declarative Chart Definitions

```javascript
{
    series: [{
        name: 'Female',
        color: 'rgba(223, 83, 83, .5)',
        data: [[161.2, 51.6], [167.5, 59.0]],
    }, {
        name: 'Male',
        color: 'rgba(119, 152, 191, .5)',
        data: [[174.0, 65.6], [175.3, 71.8]],
    }],
    chart: {
        type: 'scatter',
        zoomType: 'xy'
    },
    title: {
        text: 'Height Versus Weight of 507 In
    },
    xAxis: {
        title: {
            text: 'Height (cm)'
        },
    },
    yAxis: {
        title: {
            text: 'Weight (kg)'
        }
    }
}
```

```javascript
{
    data: [
        {
            name: 'Female',
            mode: 'markers',
            marker: {
                color: 'rgba(223, 83, 83, .5)'
            },
            x: [161.2, 167.5, 159.5, 157.0, 15],
            y: [51.6, 59.0, 49.2, 63.0, 53.6],
        }, {
            name: 'Male',
            mode: 'markers',
            marker: {
                color: 'rgba(119, 152, 191, .5)'
            },
            x: [174.0, 175.3, 193.5, 186.5, 18],
            y: [65.6, 71.8, 80.7, 72.6, 78.8],
        }
    ],
    layout: {
        title: 'Height Versus Weight of 507 In
        xaxis: {
            title: 'Height (cm)'
        },
        yaxis: {
            title: 'Weight (kg)'
        }
    }
}
```
Declarative Charts Approach

Used by popular chart libraries:
Plotly, Highcharts, Vega-lite, many more

More than 1000 properties in each

Incompatible data models / vocabularies:
- Series in Highcharts
- Traces in Plotly
- Layers in Vega-Lite

Forces teams to choose one charting library and build on top of it
Plotly Figure Hierarchy

**DATA [ ]**
- TRACE {}
  - x, y, z [ ]
  - text [ ]
  - ERROR_X, ERROR_Y {}
    - array, arrayminus [ ]
- MARKER {}
  - color ABC or [ ]
  - size 123 or [ ]
  - colorscale ABC or [ ]
  - symbol ABC

**LAYOUT {}**
- showlegend True/False
- autosize True/False
- XAXIS, YAXIS {}
  - SCENE {}
  - XAXIS, YAXIS, ZAXIS {}
- TITLE {}
- LEGEND {}
- ANNOTATIONS {}

{ } = dictionary
[ ] = list
ABC = string
123 = number

Credit: plotly_js_cheat_sheet.pdf
Sample charts

Box on photometry

histogram2dcontour: ra vs. dec
Example – creating box chart

data = [
    {
        type: 'box',
        name: 'w1+w2',
        tbl_id: "wiseCatTbl",
        y: "tables::w1mpro+w2mpro"
    },
    {
        type: 'box',
        name: 'w2+w3',
        tbl_id: "wiseCatTbl",
        y: "tables::w2mpro+w3mpro"
    }
];

layout = {
    title: 'Box on photometry',
};

firefly.getViewer().showChart({layout, data}, 'boxContainer');
Firefly Enhancements

- Aggregating data on the server
- Upper/lower limits display
- Default chart: switching between heatmap and scatter based on the number of points
- UI for basic actions and chart modifications
Firefly Enhancements

Multi-trace UI: add, remove, rearrange traces
Plotly proved to be a good choice to support our chart display and brushing and linking capabilities.

Its declarative chart schema made our charting framework lightweight, flexible, and easy to extend.

Any chart that Plotly supports can be used to visualize table data in Firefly API.

Thank you!
Displaying Upper and Lower limits
Firefly Enhancements

Aggregating data on the server

Firefly Histogram

Firefly Heatmap