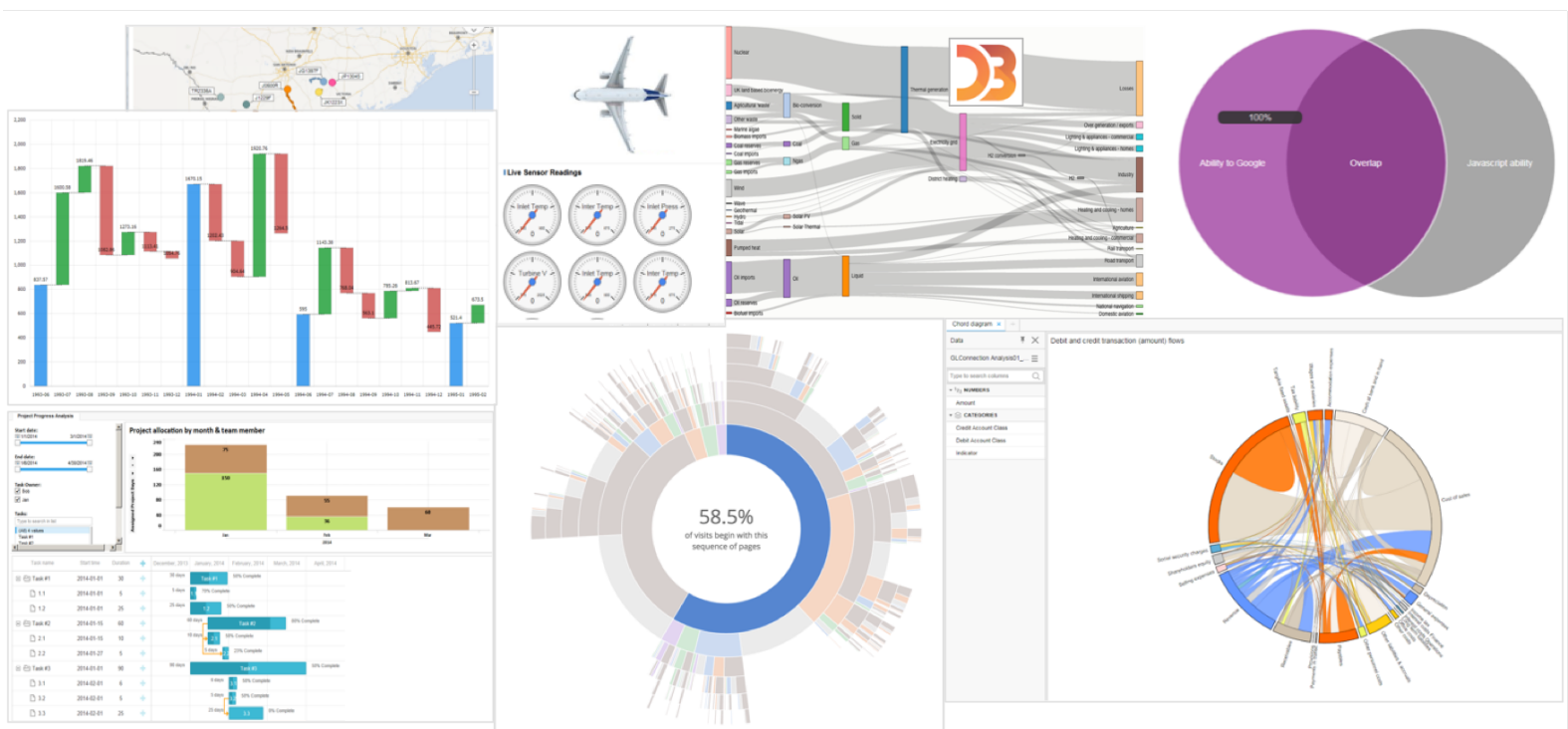


JavaScript Visualization Framework – JSViz V3.4

Samples



1. Introduction	3
2. Circle Packing.....	4
3. d3 Chord Diagram	6
4. d3 Gauges.....	8
5. Doughnut Chart	10
6. Doughnut Chart Multiple	12
7. Doughnut Chart with Configuration.....	14
8. Doughnut Chart with Custom HTML	16
9. Epicyclic Gears	18
10. Google Area Chart.....	20
11. Google Gauges	22
12. Horizon Chart.....	24
13. JSON View	26
14. Polar Chart	28
15. Rotating World.....	30
16. Ternary Chart	32
17. Tree Control.....	34
18. Venn Diagram	36
19. Word Cloud.....	38

1. Introduction

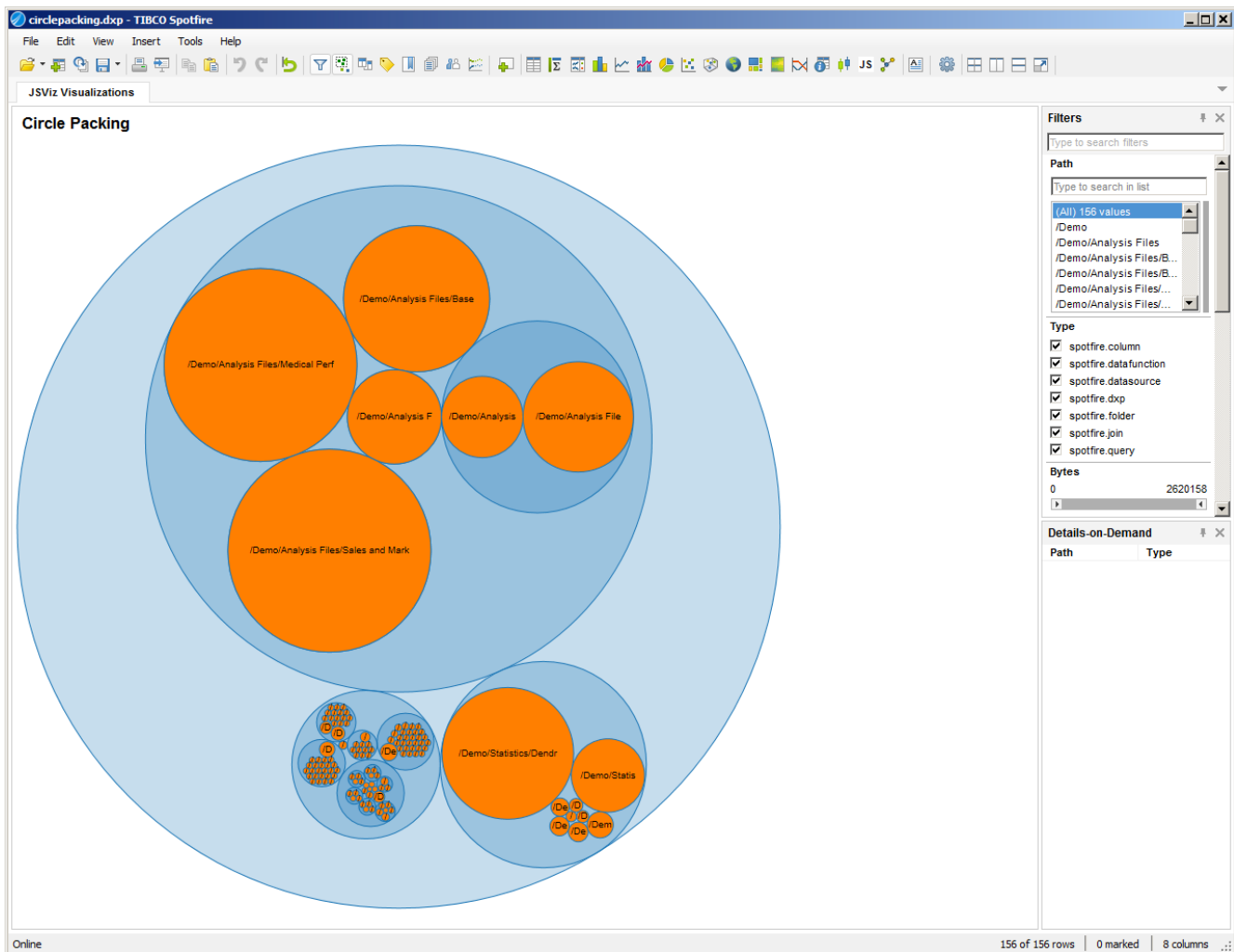
The JavaScript Visualization Framework – JSViz – ships with a number of samples that demonstrate various aspects of creating custom visualizations.

Each sample is depicted along with the following information:

- Description – a brief overview of the visualization
- Status – what visualization features are implemented
- Data – the data file used to build the visualization
- Files – the JavaScript and CSS files used to build the visualization

All samples were originally built using Linked Content items which were then converted to embedded. The original base URL used to setup the Linked Content items was <http://localhost/jsviz>.

2. Circle Packing



Description

This sample shows how to use parent-child table data to draw a hierarchical structure where data values (in this case the size of items in the Spotfire Library) control the nesting and size of items.

This sample demonstrates:

- Converting a parent-child data table into a hierarchical tree
- Rendering using D3 circle-packing layout
- d3 Tooltips

Status

Only initial rendering and resizing are implemented (no marking or filtering).

Data

The sample is based on a data set containing information on the Spotfire Demo Library contents. The data was extracted via a .NET program using the Spotfire Server Web Service API. It is shipped with JSViz as "TSSLibrarySize.sbdf".

Files

DXP File: circlepacking.dxp

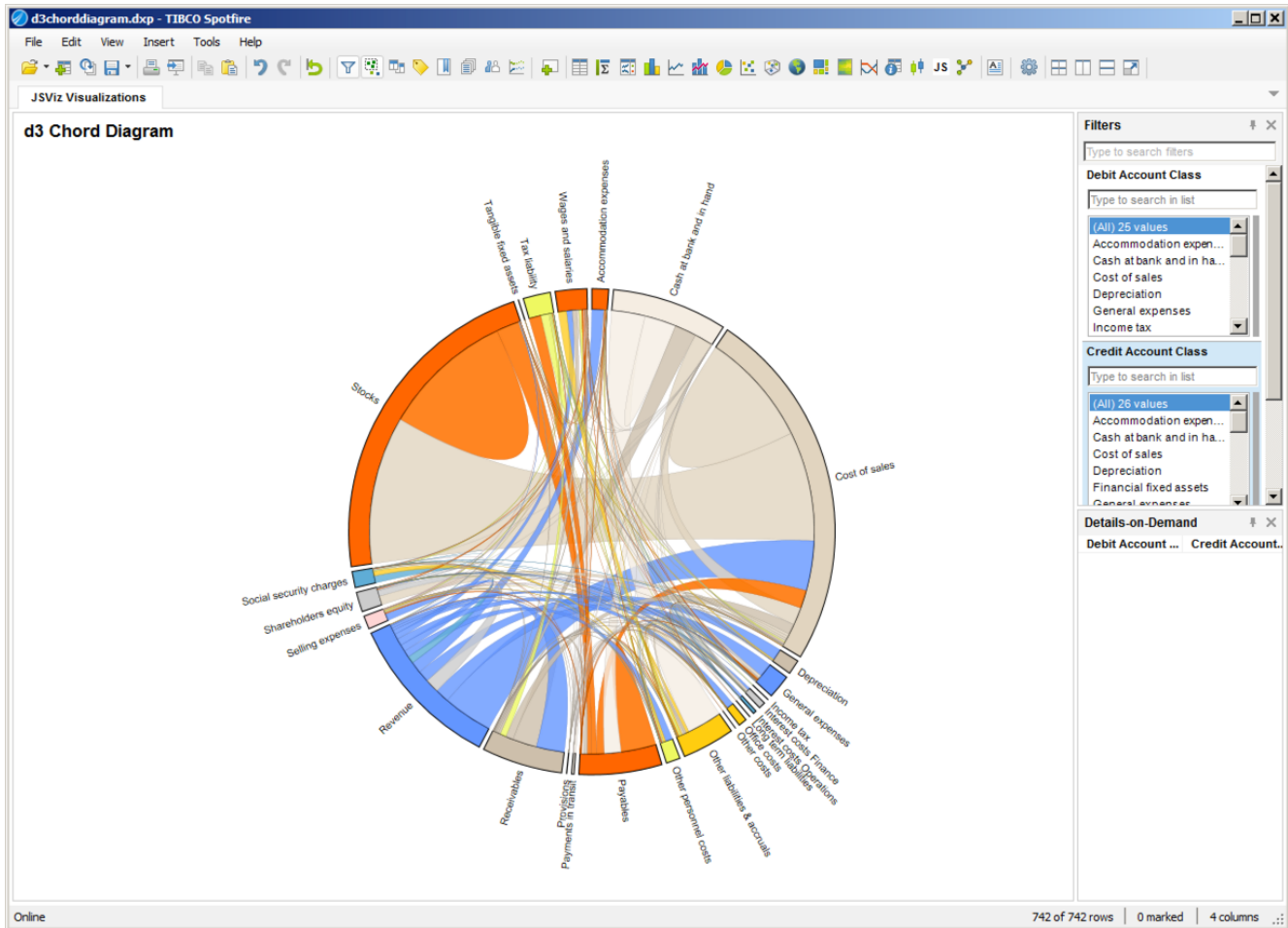
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/CirclePacking.css

JS File: /jsviz/src/js/CirclePacking.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js
 /jsviz/src/js/base.js
 /jsviz/src/js/DataStructures.Tree.librarysize.js

3. d3 Chord Diagram



Description

This sample shows how to use the d3 libraries to create a Chord Diagram.

This sample demonstrates:

- Using the Spotfire Engine to deliver un-aggregated data values
- Using d3 API's
- Support for printing and exporting using the Export Timing option

Status

This is a very basic sample with fixed sizing. It does use the d3 capabilities to change the diagram through mouse movements.

Data

The sample data is based on a custom data set "ChordDiagram.sbdf", which is shipped with JSViz.

Files

DXP File: `d3chorddiagram.dxp`

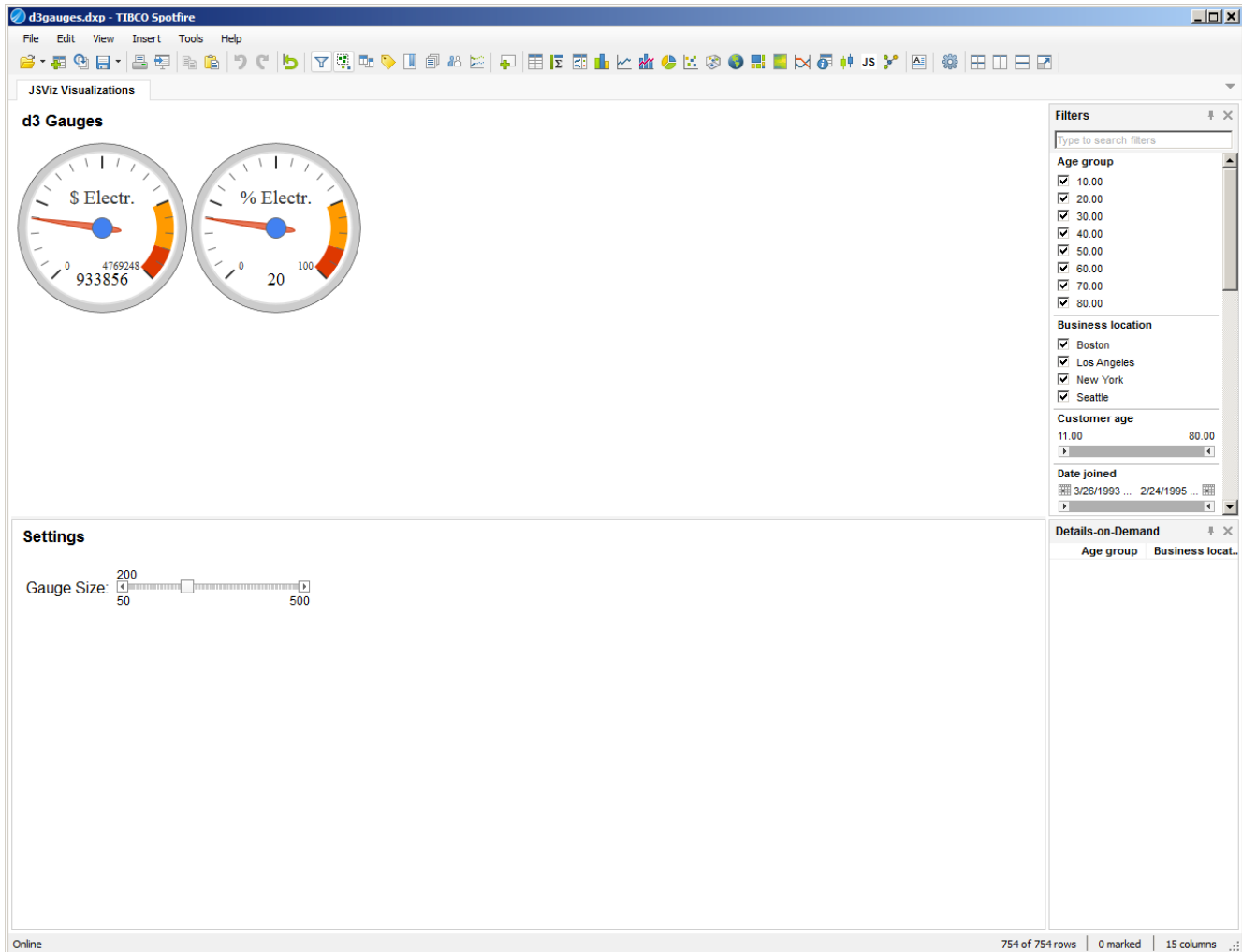
JSViz File: `/jsviz/src/lib/JSViz/JSViz.js`

CSS File: `/jsviz/src/css/D3ChordDiagram.css`

JS File: `/jsviz/src/js/D3ChordDiagram.js`

Includes: `/jsviz/lib/jquery/jquery.js`
`/jsviz/lib/d3/d3.v3.min.js`
`/jsviz/lib/d3/d3.chord.mapper.js`
`/jsviz/lib/underscore/underscore.js`

4. d3 Gauges



Description

This sample shows how to use the d3 libraries to create Gauge Controls.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using custom expressions in the data configuration
- Using document properties in the configuration parameters
- Using d3 API's
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented (no marking or filtering). Additionally, changing the value of the "GaugeSize" Document Property will change the size of the gauges.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbd".

Files

DXP File: d3gauges.dxp

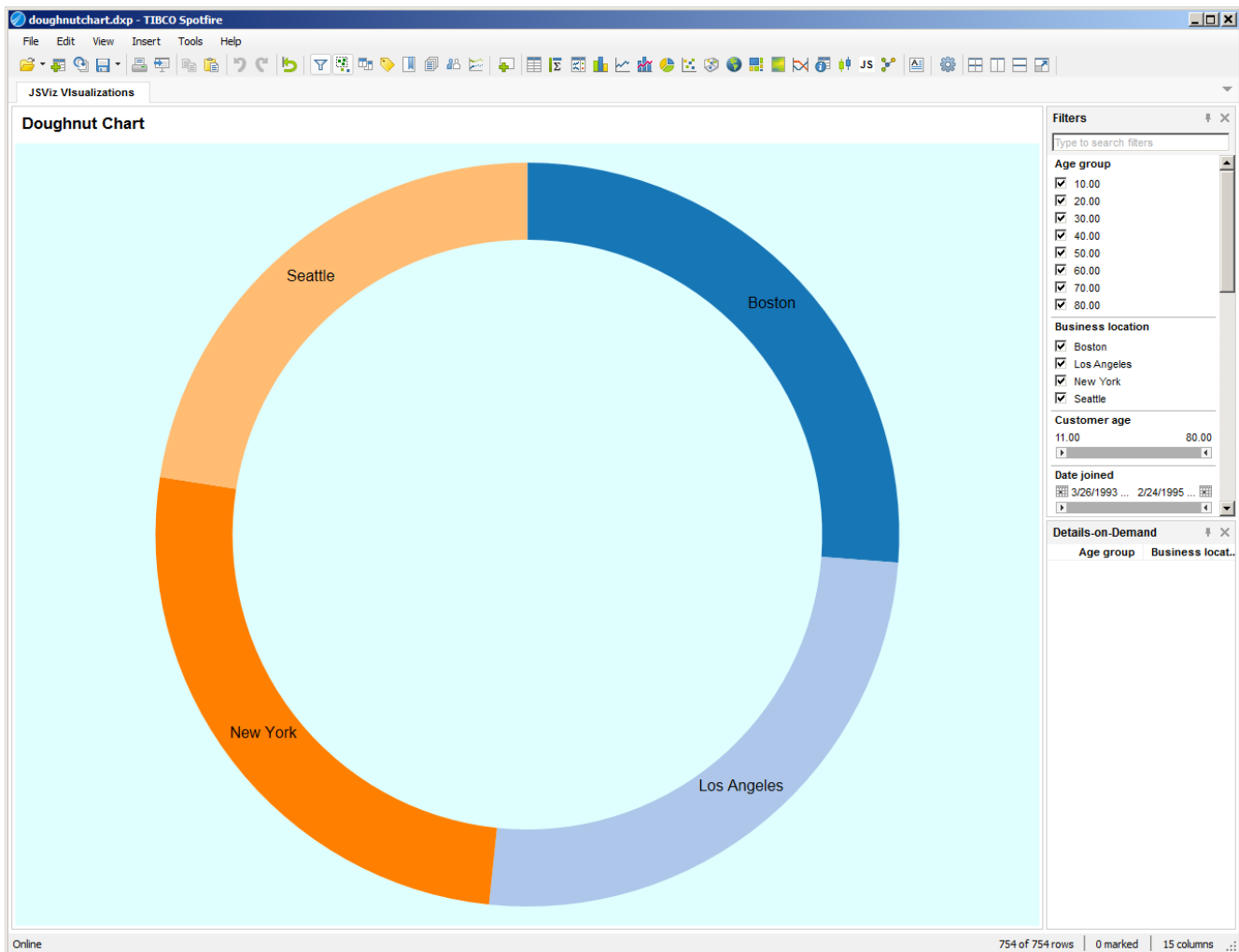
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: N/A

JS File: /jsviz/src/js/D3Gauges.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js
 /jsviz/lib/d3/d3.gaugecontrol.js

5. Doughnut Chart



Description

The sample shows how to create a doughnut chart using d3. This is the most complete sample and forms the basis for the tutorial set that ships with JSViz.

This sample demonstrates:

- Using the Spotfire Data Engine to calculate aggregated data values
- Using marking to control the display of the JS visualization
- Using the JS visualization to control marking within Spotfire
- Resizing
- Using iPad gestures for marking

Status

This is the most complete sample. It does not currently demonstrate script execution or runtime state.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbdf".

Files

DXP File: doughnutchart.dxp

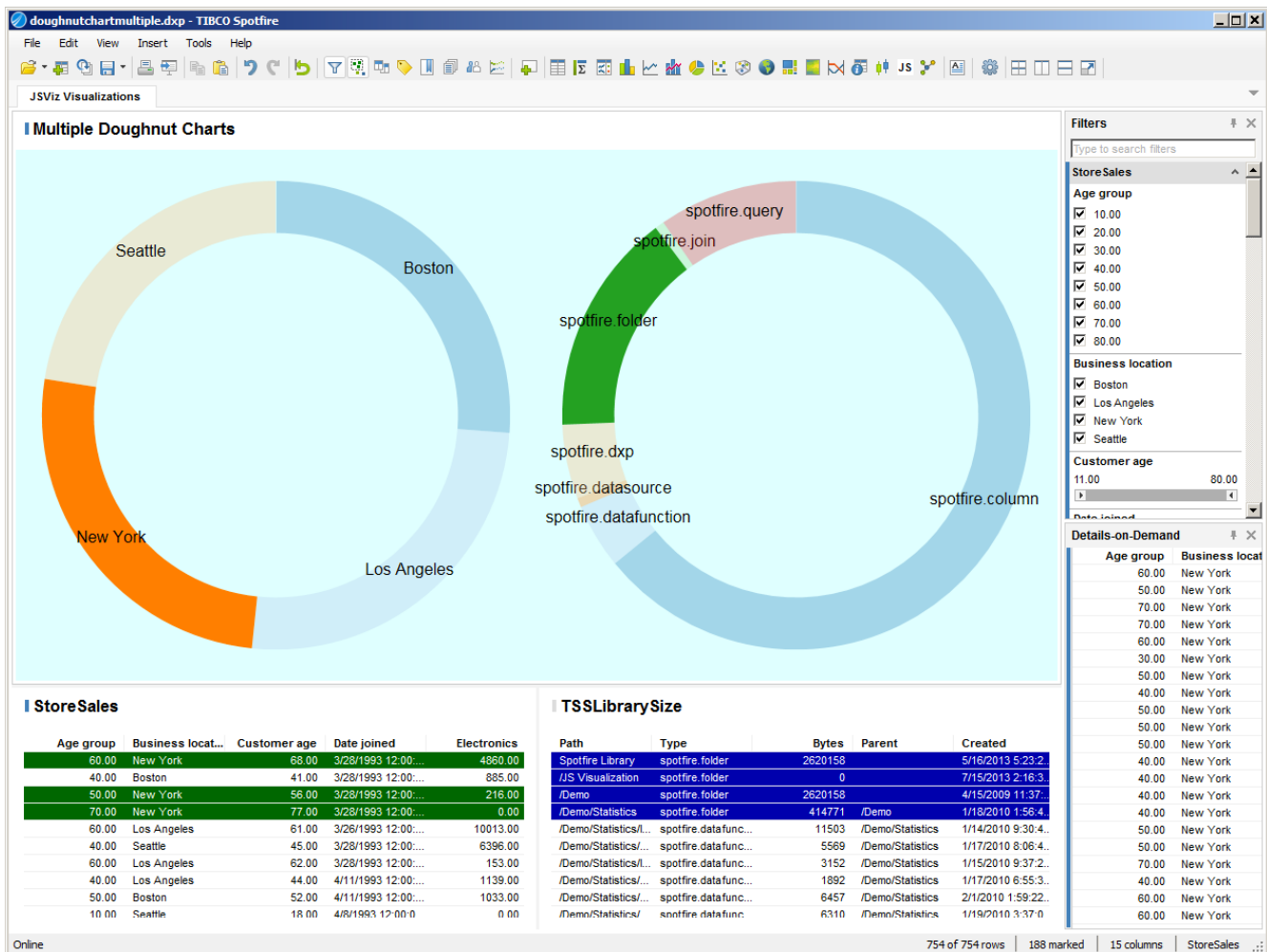
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/DoughnutChart.css

JS File: /jsviz/src/js/DoughnutChart.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js

6. Doughnut Chart Multiple



Description

The sample builds on the Doughnut Chart sample to demonstrate using multiple data tables to create a single visualization. In this instance, the two doughnut charts are drawn from different data tables with different marking sets.

This sample demonstrates:

- Using the Spotfire Data Engine to calculate aggregated data values
- Using multiple data tables within the same JSViz visualization
- Using marking to control the display of the JS visualization
- Using the JS visualization to control marking within Spotfire

- Resizing

Status

This sample demonstrates initial drawing, resizing and marking across multiple data tables.

Data

This sample uses two data sets that ship with JSViz:

- Store Sales data that comes with Spotfire Analyst - "StoreSales.sbdf".
- a data set containing information on the Spotfire Demo Library contents - "TSSLibrarySize.sbdf".

Files

DXP File: doughnutchartmultiple.dxp

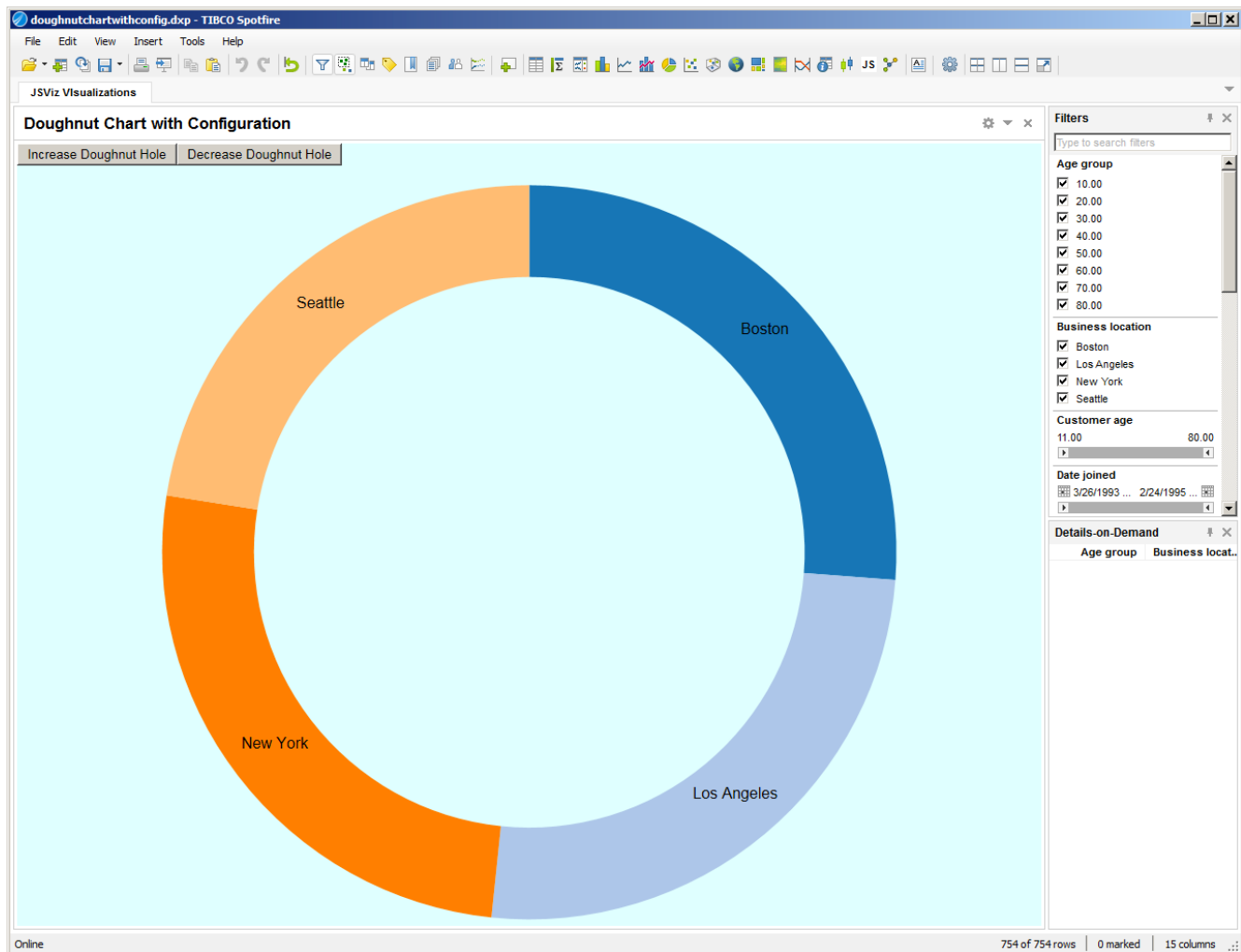
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/DoughnutChart.css

JS File: /jsviz/src/js/DoughnutChartMultiple.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js

7. Doughnut Chart with Configuration



Description

This sample expands the basic doughnut chart sample to demonstrate:

- Chart configuration write back – the configuration data in the JSViz plugin is updated from the visualization using the `setConfig()` method.
- Logging – sample logging statements have been added to the JS code. Enable logging in the JSViz configuration to see these written to a logfile in Spotfire Analyst.

Status

This is the most complete sample. It does not currently demonstrate script execution or runtime state.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbdf".

Files

DXP File: doughnutchartwithconfig.dxp

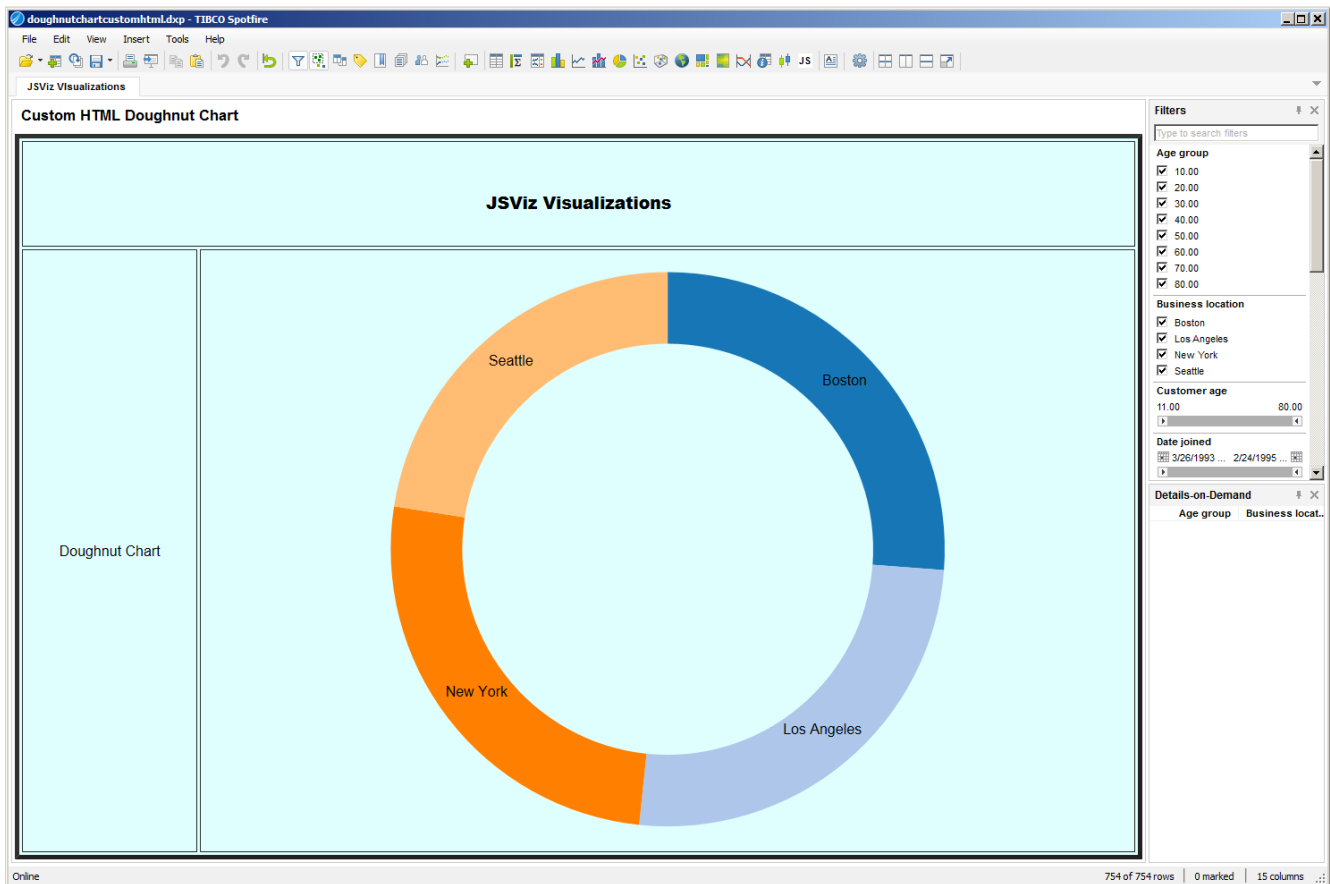
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/DoughnutChart.css

JS File: /jsviz/src/js/DoughnutChartWithConfig.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js

8. Doughnut Chart with Custom HTML



Description

This sample builds on the basic doughnut chart sample to demonstrate:

- Use of a Custom HTML Template
- Updated Marking, Sizing and Resizing code required when using Custom HTML

Status

This sample is complete as an introduction to using a Custom HTML template. It does not address all potential issues that could arise.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbdf".

Files

DXP File: doughnutchartwithconfig.dxp

JSViz File: /jsviz/src/lib/JSViz/JSViz.js

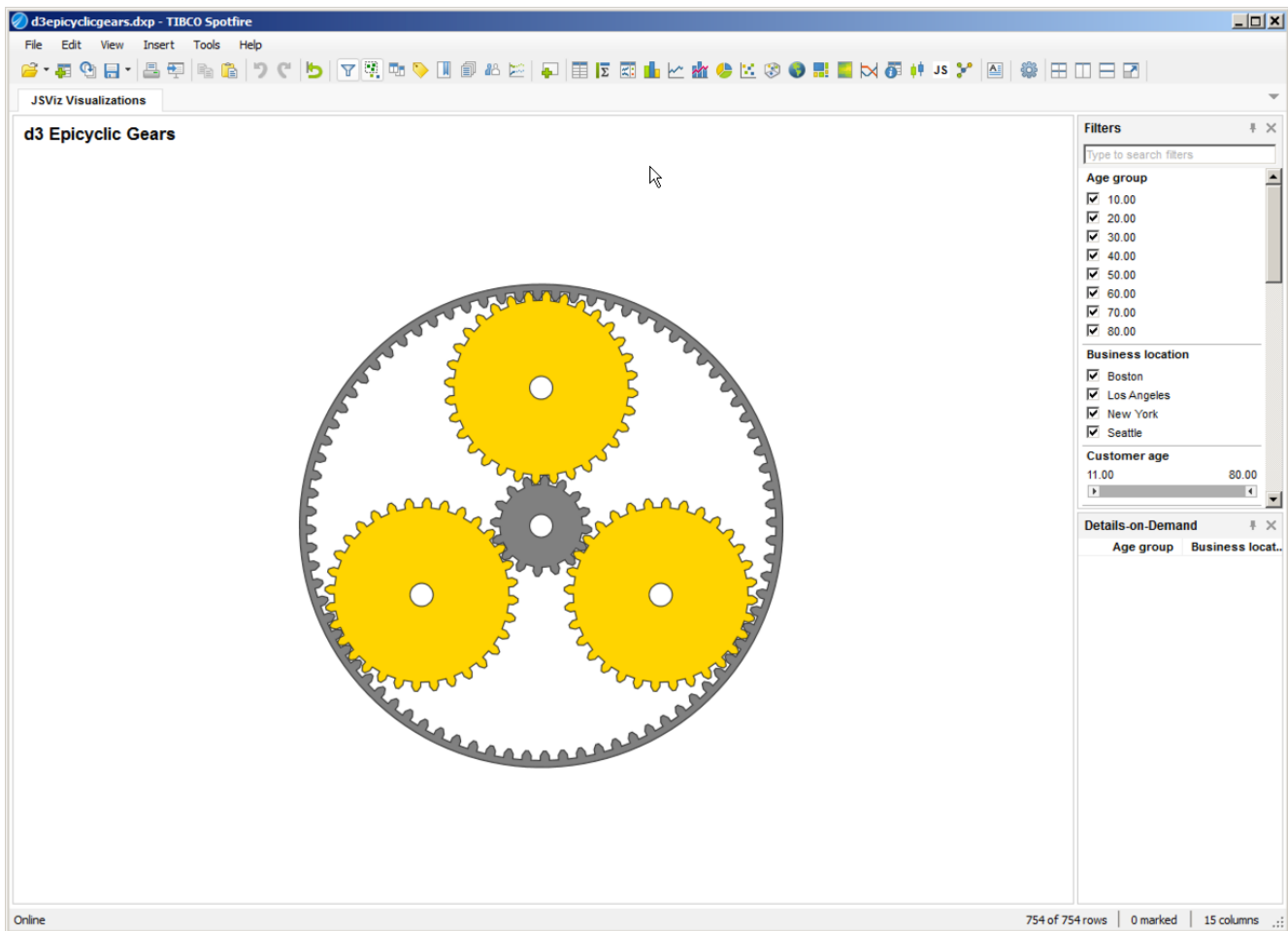
CSS File: /jsviz/src/css/DoughnutChartCustomHTML.css

JS File: /jsviz/src/js/DoughnutChartCustomHTML.js

HTML File: /jsviz/src/js/DoughnutChartCustomHTML.html

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js

9. Epicyclic Gears



Description

This sample uses the topojson d3 libraries to draw a set of epicyclic or planetary gears. While the sample is running, the user can continue to interact with Spotfire. This demonstrates the ability of JavaScript to execute threads independently of Spotfire.

This sample demonstrates:

- Using timers within JavaScript to update visualizations
- Resizing
- Support for printing and exporting using the Export Timing option

Status

This is a basic sample with fixed sizing.

Data

There is no data contained within this sample file.

Files

DXP File: `d3epicyclicgears.dxp`

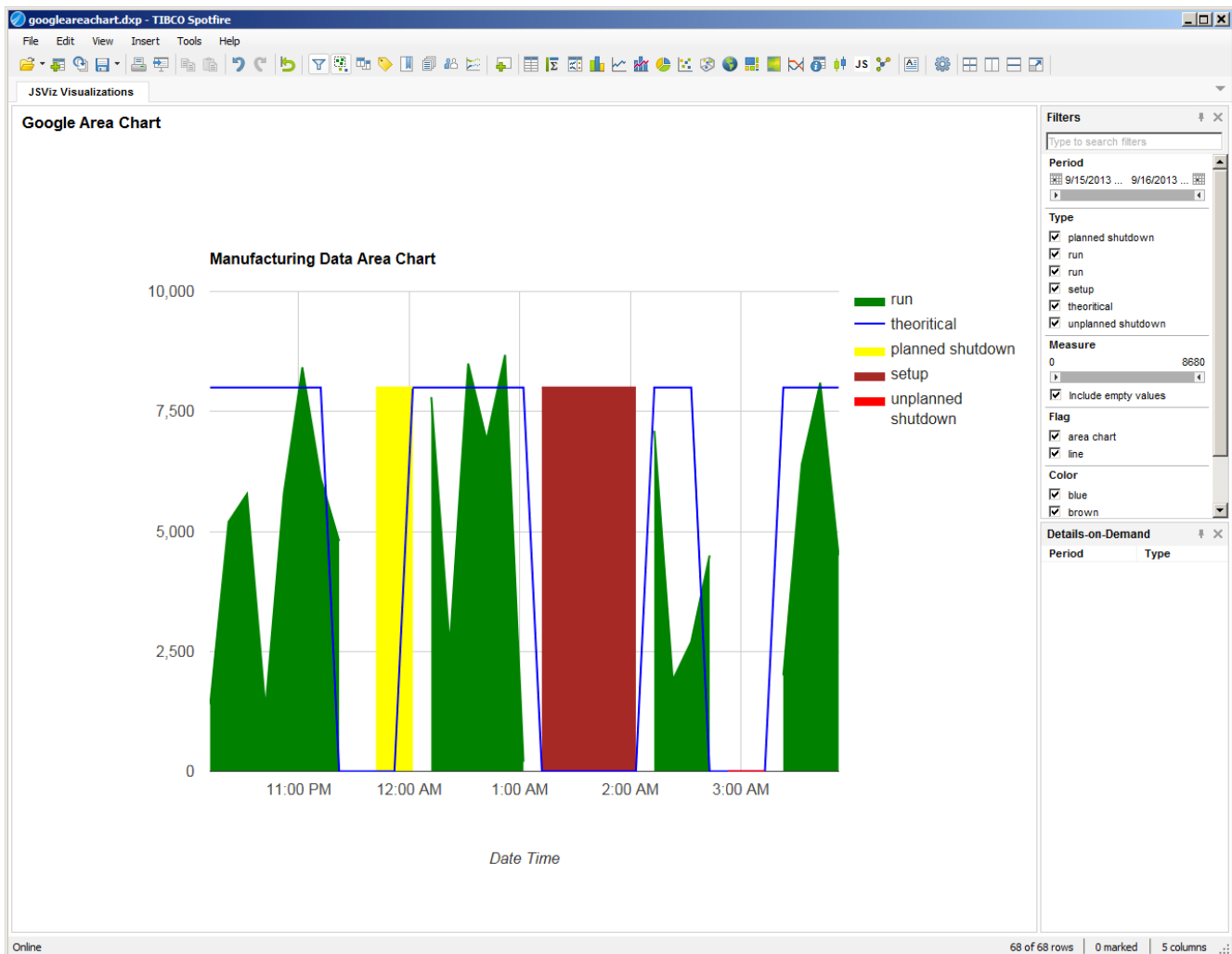
JSViz File: `/jsviz/src/lib/JSViz/JSViz.js`

CSS File: `/jsviz/src/css/D3EpicyclicGears.css`

JS File: `/jsviz/src/js/D3EpicyclicGears.js`

Includes: `/jsviz/lib/jquery/jquery.js`
`/JSViz/lib/d3/d3.v3.min.js`

10. Google Area Chart



Description

This sample shows how to use the Google Chart APIs to draw an Area Chart.

This sample demonstrates:

- Using the Spotfire Data Engine to calculate aggregated data values
- Using Google Chart APIs with JSViz
- Support for printing and export using the Export Timing option

Status

Only initial rendering and resizing are implemented.

Data

The sample data is based on a custom data set "AreaChart.sbf", which is shipped with JSViz.

Files

DXP File: `googleareachart.dxp`

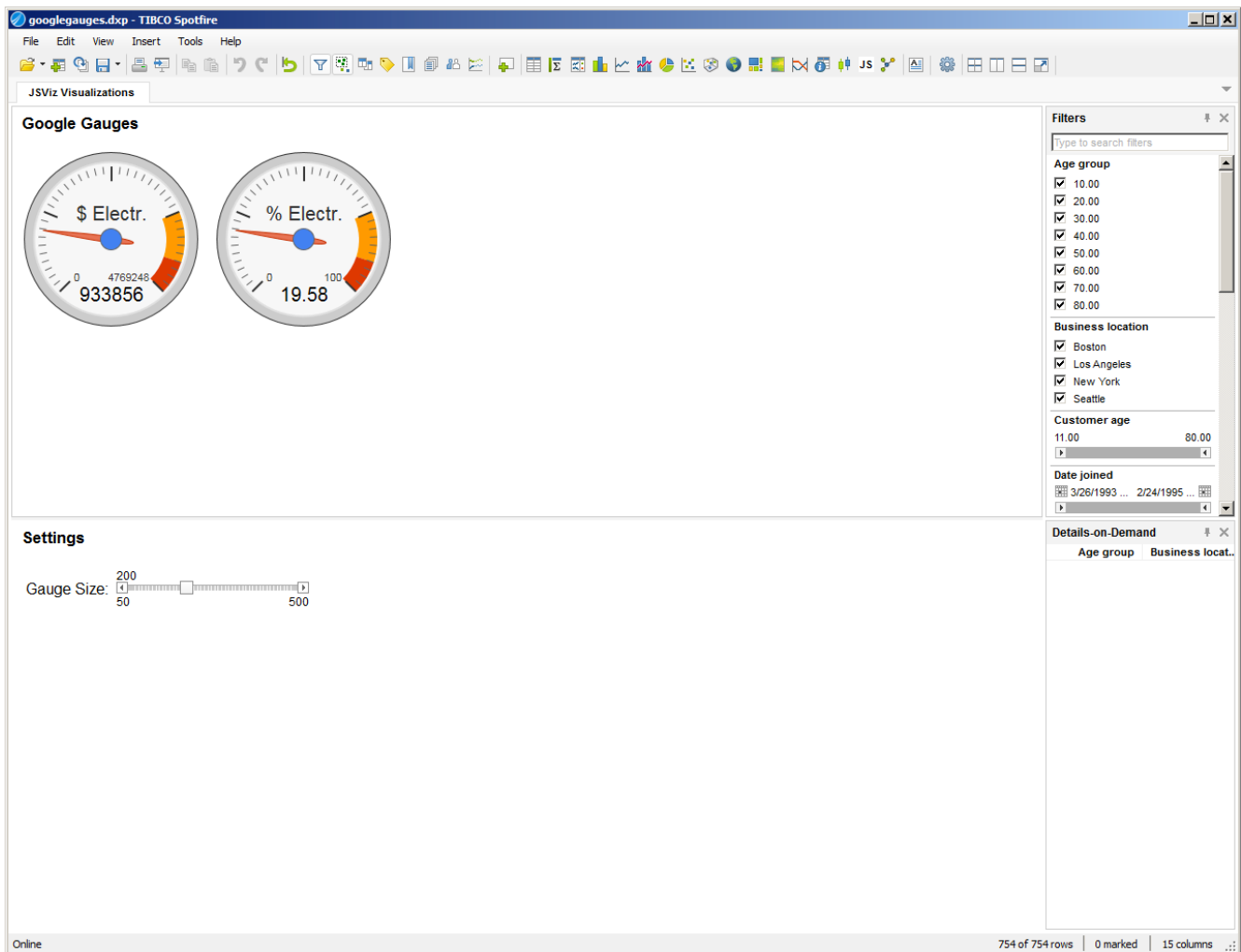
JSViz File: `/jsviz/src/lib/JSViz/JSViz.js`

CSS File: `N/A`

JS File: `/jsviz/src/js/GoogleAreaChart.js`

Includes: `/jsviz/lib/jquery/jquery.js`
`https://www.google.com/jsapi?__id=js`

11. Google Gauges



Description

This sample shows how to use the Google Chart APIs to create Gauge Controls.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using custom expressions in the data configuration
- Using document properties in the configuration parameters
- Using the Google Chart API's
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented (no marking or filtering). Additionally, changing the value of the "GaugeSize" Document Property will change the size of the gauges.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbdf".

Files

DXP File: `googlegauges.dxp`

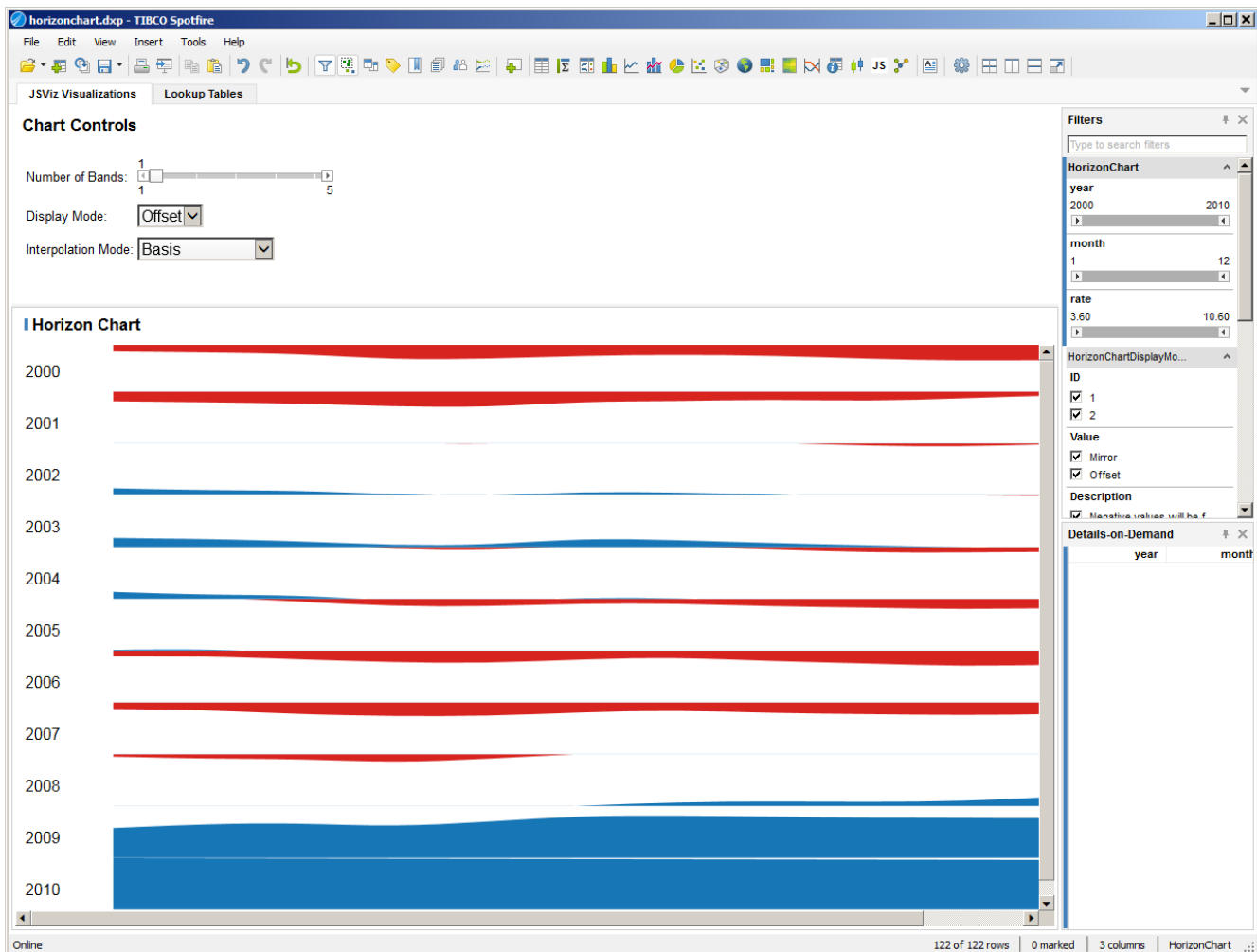
JSViz File: `/jsviz/src/lib/JSViz/JSViz.js`

CSS File: `N/A`

JS File: `/jsviz/src/js/GoogleGauges.js`

Includes: `/jsviz/lib/jquery/jquery.js`
`https://www.google.com/jsapi?__id=js`

12. Horizon Chart



Description

This sample shows how to use create a Horizon Chart using the d3 libraries. A Horizon Chart folds larger vertical values within the same space to make better use of screen real estate.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using the d3 API's
- Using IronPython Scripting to update the Configuration Properties of a JSViz visualization

- Resizing
- Support for printing and exporting using the Export Timing option

Status

The sample does not use the simpler technique of passing Document Property values directly to the JSViz Visualization. This is not intended to be a best practice, but simply a way to demonstrate using scripts with JSViz.

Data

The data is based on sample stock ticker data retrieved from the internet and anonymized. It is shipped with JSViz as "HorizonChart.sbf". The additional lookup tables are included as "HorizonChartDisplayModes.txt" and "HorizonChartInterpolationModes.txt".

Files

DXP File: horizonchart.dxp

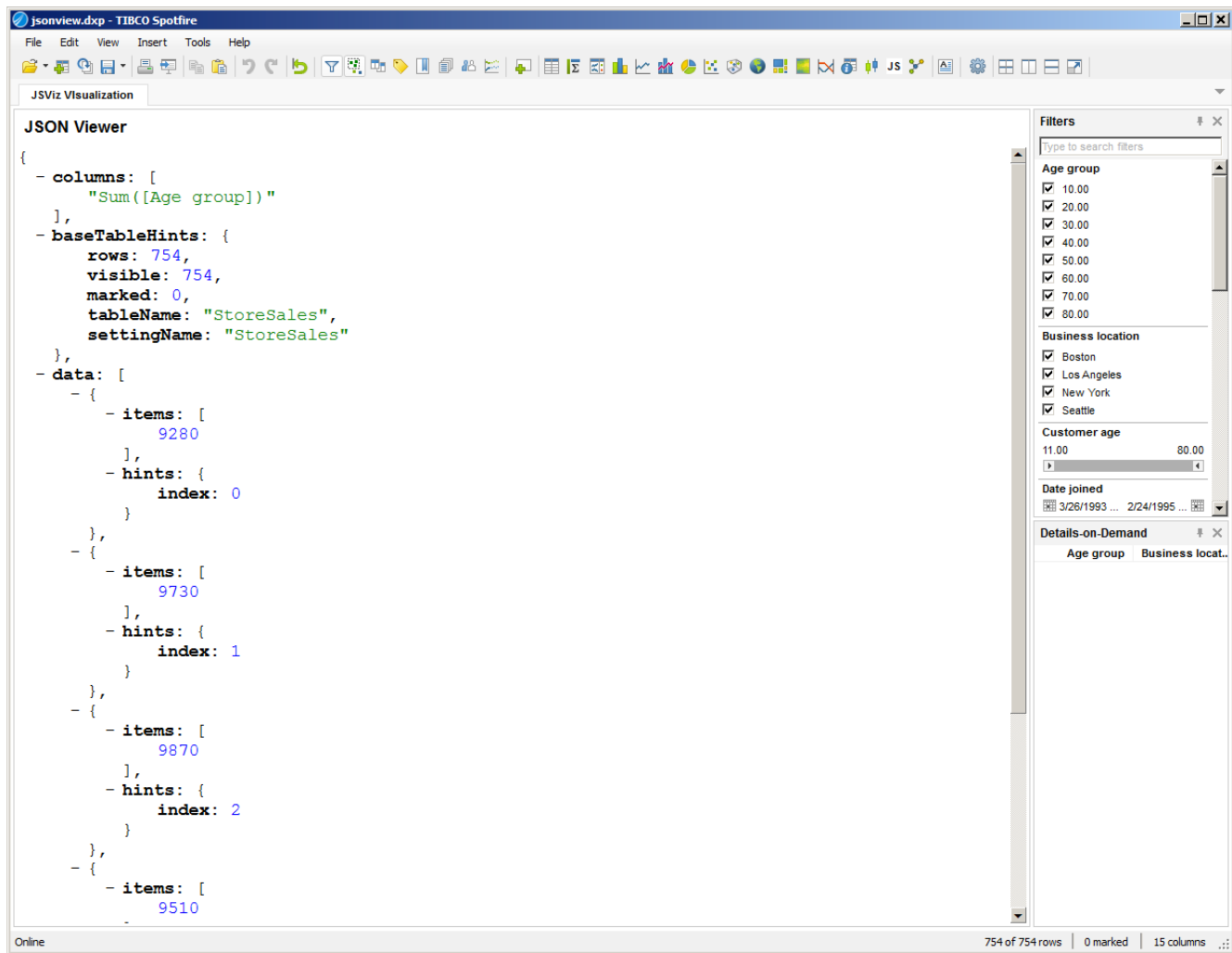
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/HorizonChart.css

JS File: /jsviz/src/js/HorizonChart.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js
 /jsviz/lib/d3/d3.horizon.js

13. JSON View



Description

This sample is provided as a debugging tool. It displays the JSON being passed to the visualization by JSViz in an indented format with the option to expand and collapse each nested tree. The library used is `jquery.jsonview` from <https://github.com/yesmeck/jquery-jsonview>.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using a 3rd party visualization library – `jquery.jsonview`
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented, no marking.

Data

Store Sales data that comes with Spotfire Analyst. It is shipped with JSViz as "StoreSales.sbdf".

Files

DXP File: jsonview.dxp

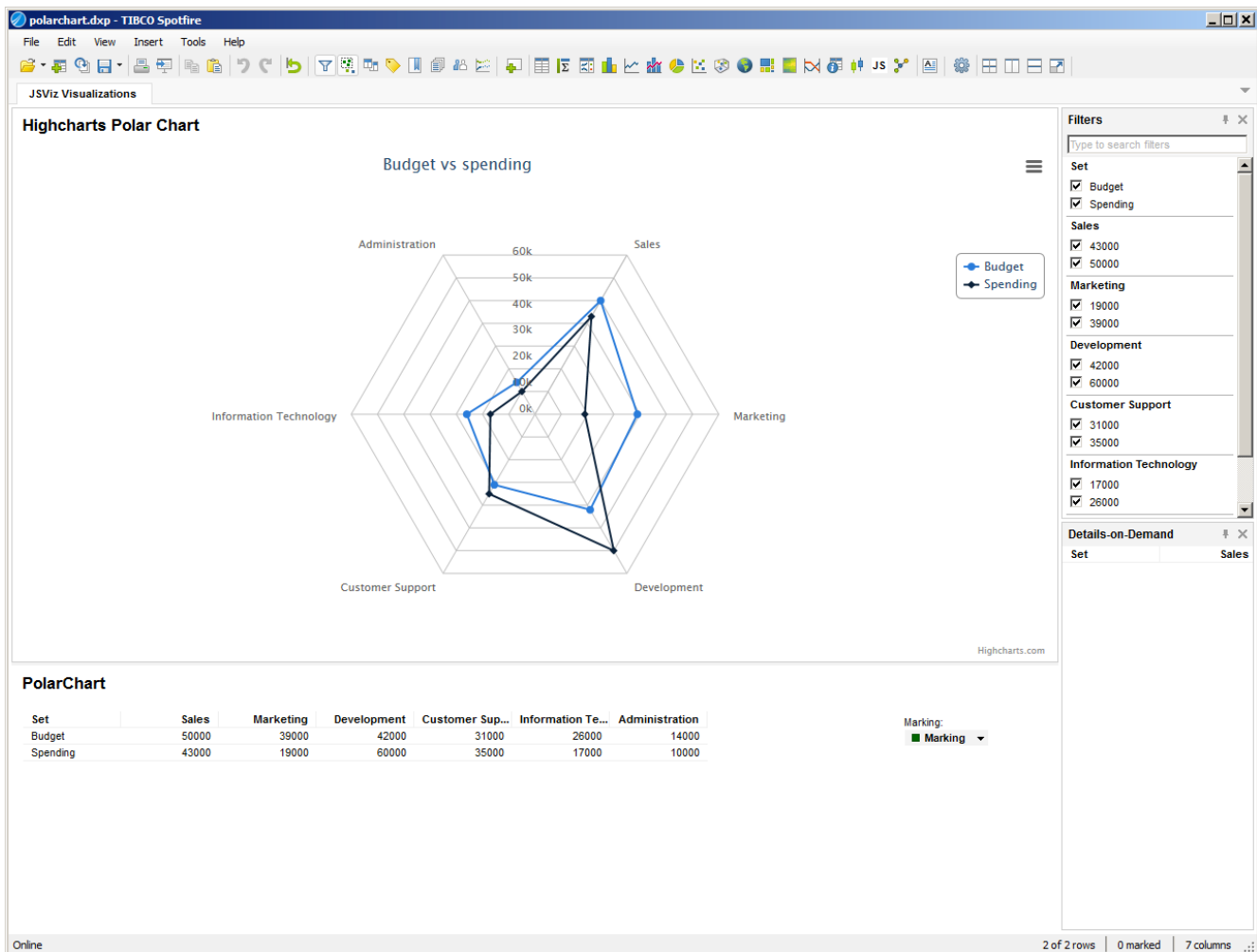
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: N/A

JS File: /jsviz/src/js/JSONViewer.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/jsonview/jquery.jsonview.css
 /jsviz/lib/jsonview/jquery.jsonview.js

14. Polar Chart



Description

This sample demonstrates how to embed a 3rd party commercial charting library, in this case a polar chart from Highcharts.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using the Highcharts API's
- Resizing
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented, no marking. Tooltips are implemented by Highcharts directly.

Data

The sample data is a custom data set "PolarChart.sddf", which is shipped with JSViz.

Files

DXP File: polarchart.dxp

JSViz File: /jsviz/src/lib/JSViz/JSViz.js

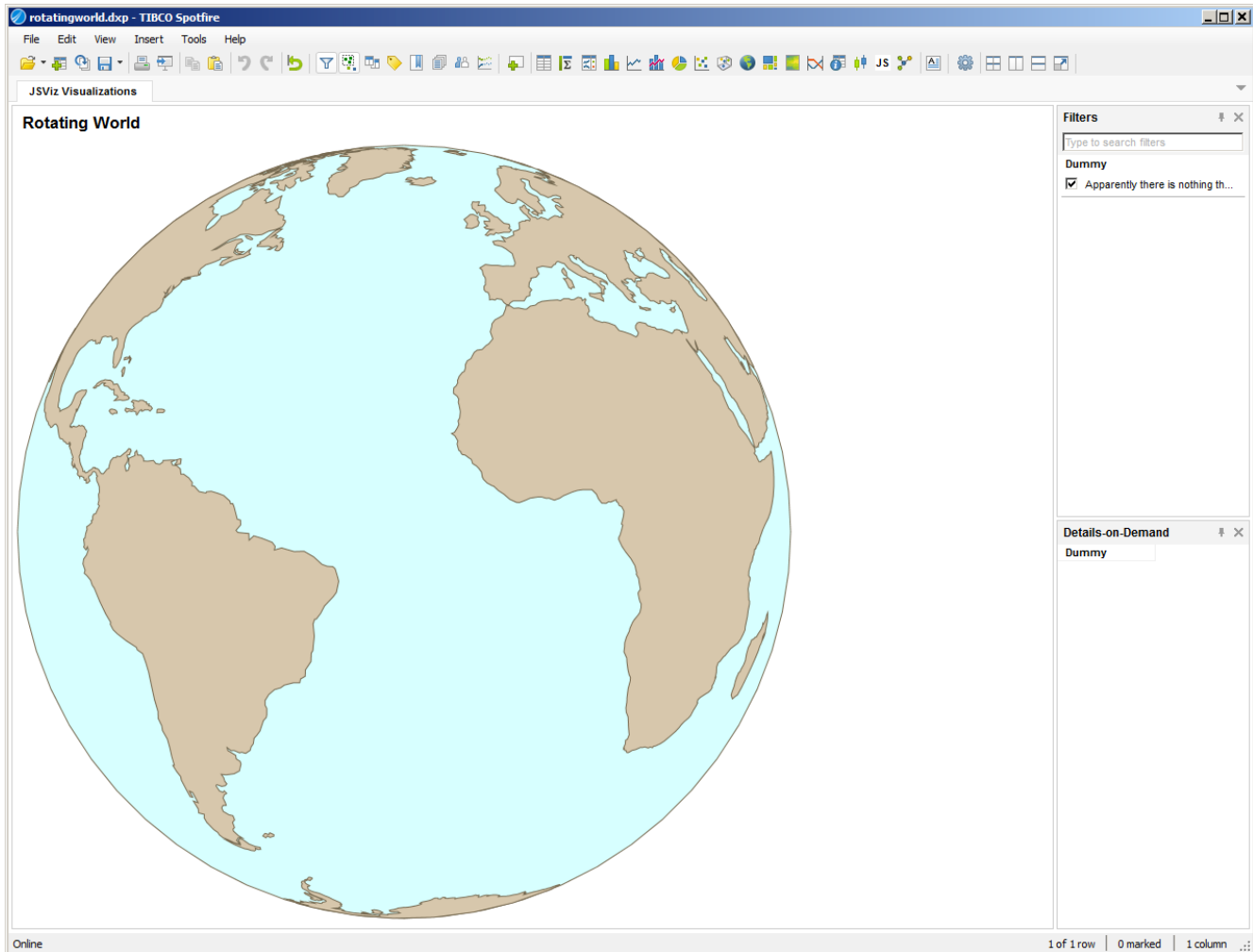
CSS File: N/A

JS File: /jsviz/src/js/PolarChart.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/highcharts/highcharts.src.js
 /jsviz/lib/highcharts/highcharts-more.src.js
 /jsviz/lib/highcharts/exporting.src.js

Note: *The user must download the Highcharts libraries themselves. They are not included with the JSViz installation*

15. Rotating World



Description

This sample uses the topojson d3 libraries to draw a rotating world map. While the sample is running, the user can continue to interact with Spotfire. This demonstrates the ability of JavaScript to execute threads independently of Spotfire.

This sample demonstrates:

- Using timers within JavaScript to update visualizations
- Resizing
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented, no marking.

Data

The data used to construct the world map is JSON data included via JavaScript. It is in the file "data/world-110m.json.js".

Files

DXP File: rotatingworld.dxp

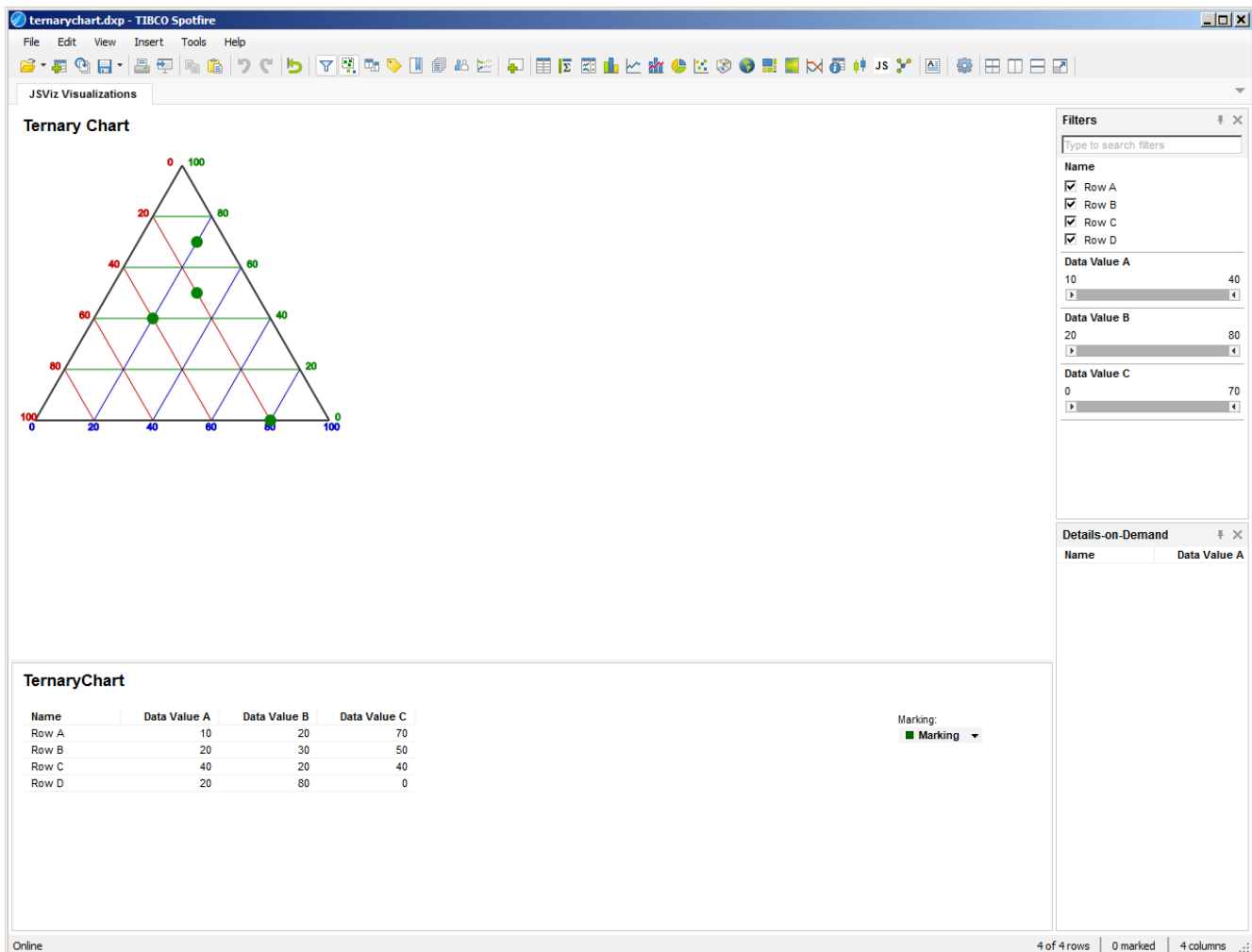
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: N/A

JS File: /jsviz/src/js/RotatingWorld.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js
 /jsviz/lib/topojson/topojson.v0.min.js
 /jsviz/data/world-110m.json.js

16. Ternary Chart



Description

This sample shows how to use d3 to create a completely new chart type. In this case a Ternary Chart. A description of a Ternary Chart can be found here: http://en.wikipedia.org/wiki/Ternary_plot.

This sample demonstrates:

- Using the Spotfire Engine to deliver un-aggregated data values
- Using the d3 API's
- Support for printing and exporting using the Export Timing option

Status

Only initial rendering is implemented and the chart is a fixed size.

Data

The sample data is a custom data set "TernaryChart.sbf", which is shipped with JSViz.

Files

DXP File: ternarychart.dxp

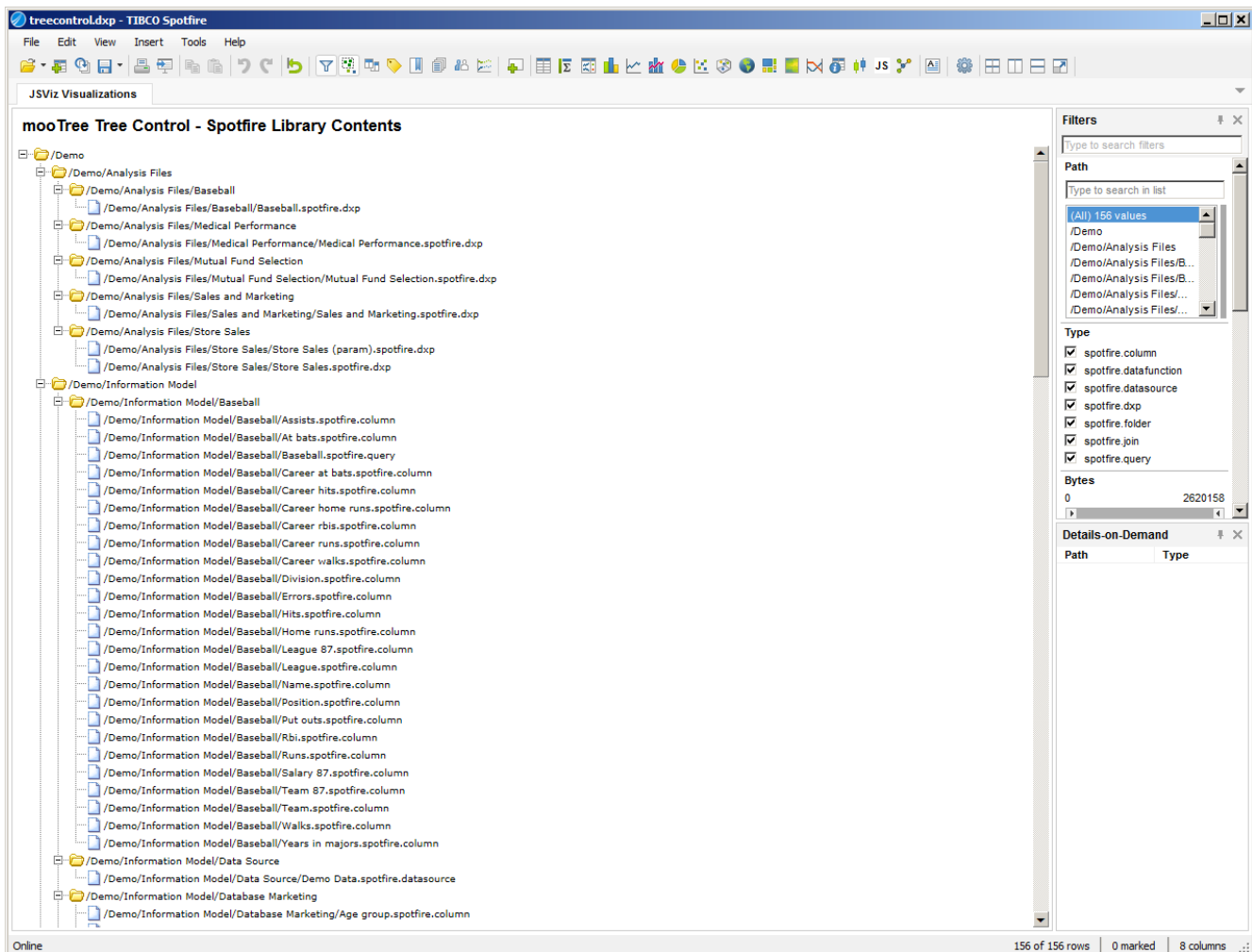
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/TernaryChart.css

JS File: /jsviz/src/js/TernaryChart.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js

17. Tree Control



Description

This sample shows how to display a hierarchical tree control in Spotfire. In this case using the mooTools mooTree control

This sample demonstrates:

- Using the Spotfire Engine to deliver un-aggregated data values
- Converting a parent-child data table into a hierarchical tree
- Using mooTools mooTree control API
- Using the Script Execution interface to run a script in Spotfire whenever a node in the tree is clicked

- Using the Runtime State interface to store and retrieve the currently selected node. Without this the selection would be lost every time the control was redrawn in Spotfire Web Player.
- Resizing
- Support for printing and exporting using the Export Timing option

Status

Initial rendering, resizing, script execution and runtime state are implemented. The sample does not support marking or filtering as filtering can result in orphaned nodes.

Data

The sample is based on a data set containing information on the Spotfire Demo Library contents. The data was extracted via a .NET program using the Spotfire Server Web Service API. It is shipped with JSViz as "TSSLibrarySize.sbd".

Files

DXP File: treecontrol.dxp

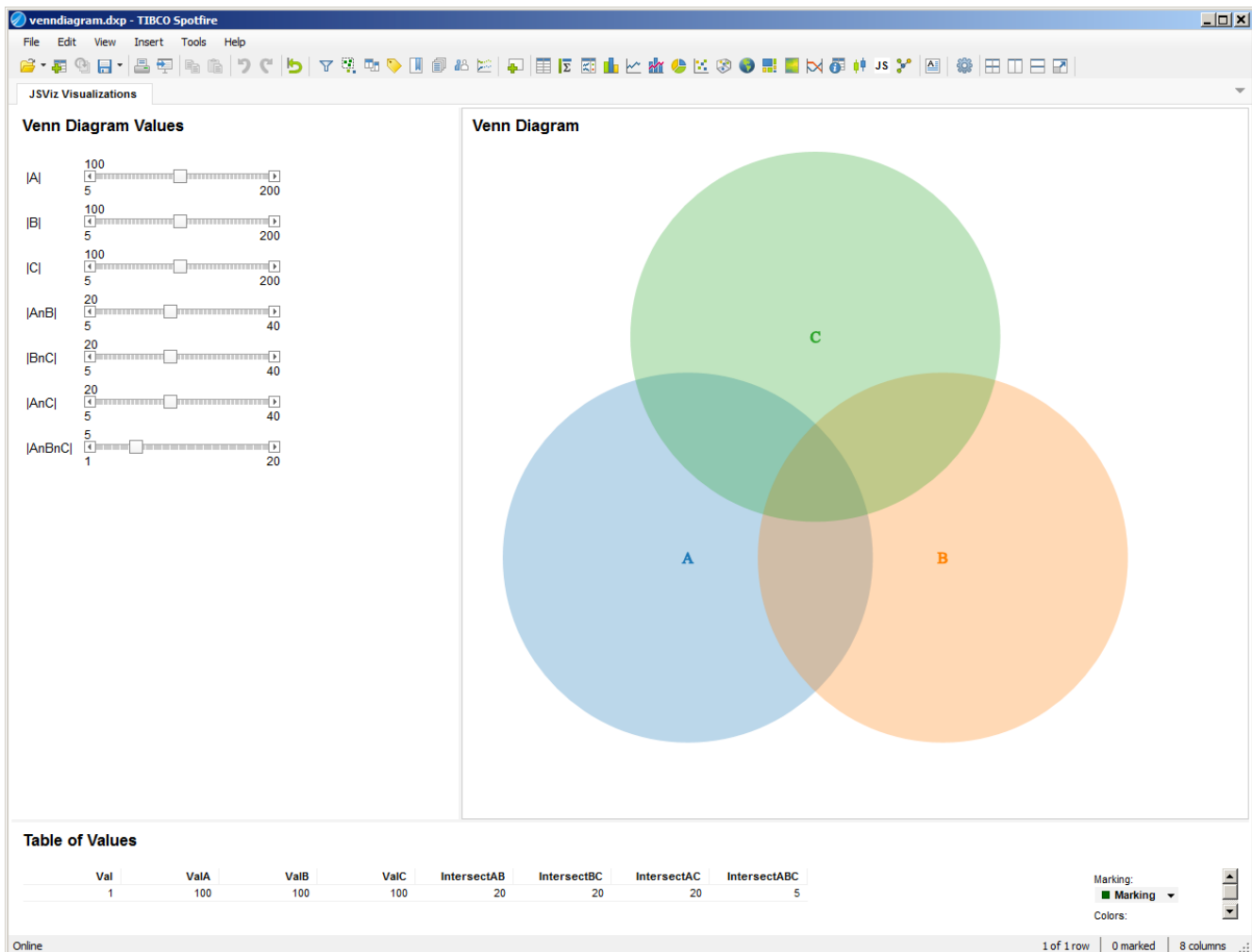
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: /jsviz/src/css/TreeControl.css

JS File: /jsviz/src/js/TreeControl.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/js/lib/d3/d3.v3.min.js
 /jsviz/js/lib/mootree/mootools/mootools.v1.11.js
 /jsviz/js/lib/mootree/mootree.js
 /jsviz/js/src/base.js
 /jsviz/js/src/DataStructures.Tree.librarysize.js

18. Venn Diagram



Description

This sample shows how to create a Venn Diagram chart using the d3 libraries..

This sample demonstrates:

- Using Calculated Columns and Document Properties to plot derived values
- Resizing
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented, no marking.

Data

The data within this sample is derived from Document Properties set via the Text Area.

Files

DXP File: venndiagram.dxp

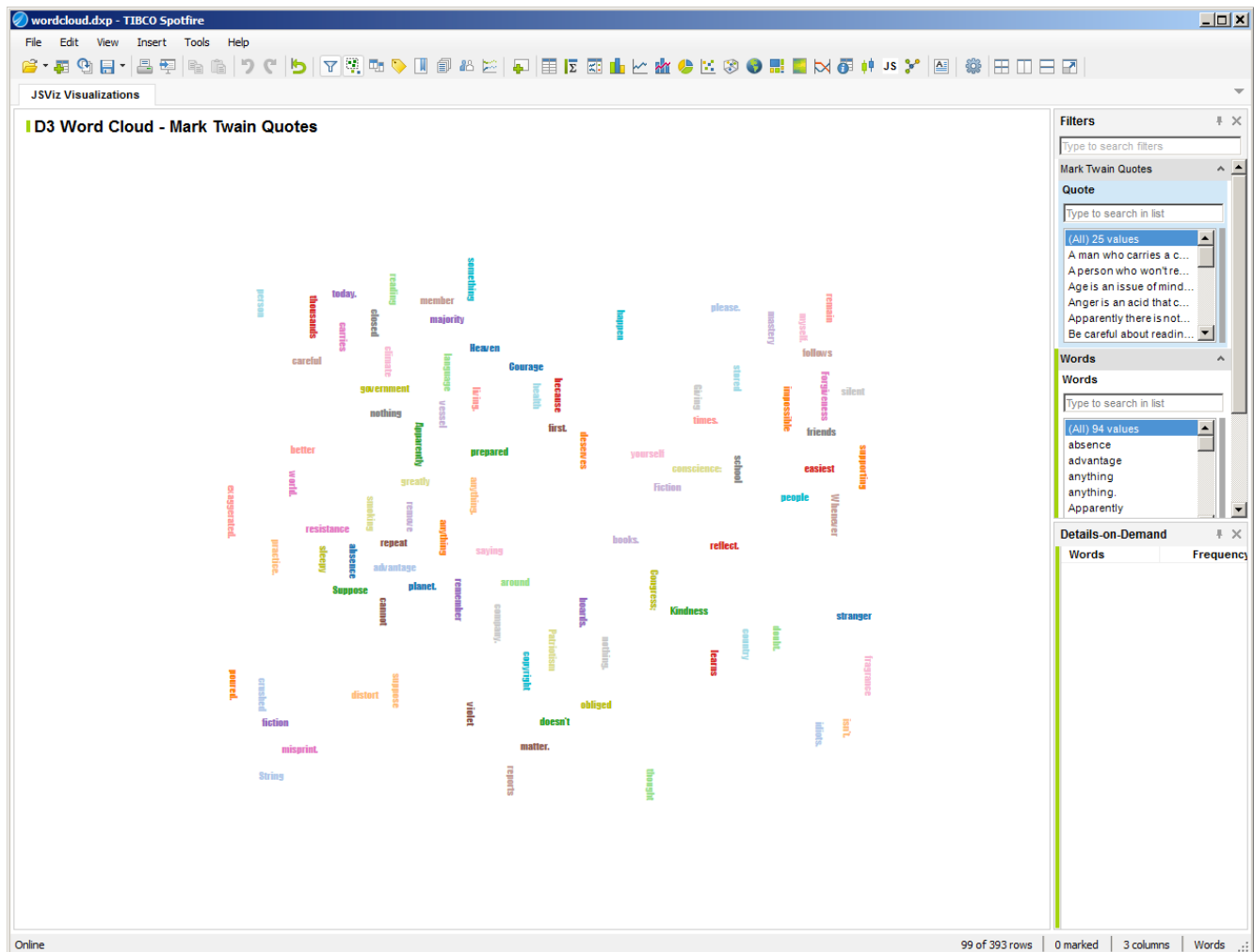
JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: N/A

JS File: /jsviz/src/js/VennDiagram.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/lib/d3/d3.v3.min.js
 /jsviz/lib/venn/d3.venn.js

19. Word Cloud



Description

This sample demonstrates using the d3 libraries to create a Word Cloud.

This sample demonstrates:

- Using the Spotfire Engine to calculate aggregated data values
- Using the d3 API, specifically the d3.layout.cloud layout
- Resizing
- Support for printing and exporting using the Export Timing option

Status

Initial rendering and resizing are implemented, no marking. The Word Cloud will respond to filtering but the layout is random every time.

Data

The sample data is a custom data set "WordCloud.sbf", containing words from quotes attributed to Mark Twain, which is shipped with JSViz.

Files

DXP File: wordcloud.dxp

JSViz File: /jsviz/src/lib/JSViz/JSViz.js

CSS File: N/A

JS File: /jsviz/src/js/WordCloud.js

Includes: /jsviz/lib/jquery/jquery.js
 /jsviz/js/lib/d3/d3.v3.min.js
 /jsviz/js/lib/d3/d3.layout.cloud.js