A fun interactive way to present multidimensional data

Every now and then you might come across the need to present your user with data that can be sliced by multiple criteria. Say, for instance, you are so lucky as to write an app that follows the departures and the introduction of new characters in 'Game of Thrones/Song of ice and fire'. Your data might be sliced by house, by character gender, by geographic location, show season, book volume etc.

Try Me

This control will give you user what they need. Have fun (just drag and drop stuff, you will get the hang of it):

How to use

Just include the JavaScript and its dependencies (D3.js and require.js) in your page. Call the setup function and you are good to go (or just have a look at the site code). Read on for a description of the variable the setup function expects.

Series

This is the data displayed. For each one of these your user would see a bar in each group of the chart. For our example these are the number of characters killed off and the number of characters introduced.

Segment

This is the way you 'slice and dice' your data (the dimensions). In our case this include the Gender, Location, House etc. These can be used as the categories for the X axis or as filters that limit the scope of the displayed data (the breadcrumbs). The filtering is performed by dragging a specific bars group onto a segment or vice versa. The user may also swap the X axis segment for another segment, in which case the total of the data should is not expected to change but it would be distributed by the new segmentation.

Segments may be arranged in a hierarchy, such as the Chapters are child elements of the Seasons and the Continent, Region, Location hierarchy.

Data Callback

Each time the user interacts with the data the callback you provide is invoked to retrieve additional data. You might want to go to the server from your callback or you might want to send all the data at once and provide it as needed (see Mock data). The callback takes in the following data:

Requested segment

The X axis segment of the new data.

Filters

If the user had narrowed the field of the data they wish to be presented, you get the filtering in the request. These filters map to the breadcrumbs on the top of the chart.

The callback is expected to return the data in a format the holds a collection of Segment Values each containing the data that should be plotted to the X axis and a collection of values, one for each series displayed.

Here is an example:

Segment House

SegmentValue Stark

Killed 42

Introduced 2

Mock data

Customizing the look

Everything is CSSable. Series and Segment descriptions include the css class name.

Touch (not so) ready

How it is done

Check out he github project.

D3

The control is based on [D3](http://d3js.org/) library by [Mike Bostock](http://bost.ocks.org/mike/). If you haven't yet, go to [the gallery](https://github.com/mbostock/d3/wiki/Gallery) and check out what you can do with this awesome stuff.

Typescript

The code is written in Typescript (and transpiled to javascript). Typescript is the way to write and maintain complex client side stuff without losing your mind. If you are not familiar with it, you might want to check [their site](http://www.typescriptlang.org/).

Note that the code can be consumed by any javascript, so it is grate for you even if you don't like that new thing.