BabiaXR
Virtual Reality Data Visualizations using only Front-End

David Moreno
dmoreno@bitergia.com
dmorenolumb@gmail.com

https://babiaxr.gitlab.io
BabiaXR is a GitLab organization where the development of the different visualizations are allocated. BabiaXR has the aim of aggregate different components, using A-Frame, that can create different types of chart in a modern browser, the web is a universal environment and any device that has a modern browser that supports WebXR can visualize the charts that the BabiaXR components produce, making it more universal and easy to use.

Lastest News

How to use Totem Component
Aug 27, 2020
Remix this example on Glitch. Or view the demo. Let's learn how to create a totem component and link it to different babiaxr-components.
<html>
  <head>
    <meta charset="utf-8">
    <title>Hello, WebVR! • A-Frame</title>
    <meta name="description" content="Hello, WebVR! • A-Frame">
    <script src="https://aframe.io/releases/0.8.2/aframe.min.js"></script>
  </head>
  <body>
    <a-scene background="#ECECEC">
      <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4C30D9" shadow></a-box>
      <a-sphere position="0 1.25 -5" radius="1.25" color="#EF205E" shadow></a-sphere>
      <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5"
                  color="#FCC65D" shadow></a-cylinder>
      <a-plane position="0 0 -4" rotation="90 0 0" width="4" height="4"
                color="#7BC8A4" shadow></a-plane>
    </a-scene>
  </body>
</html>
**Entity** - `<a-entity>`

**Components** - HTML attributes on `<a-entity>`‘s

**System** - Systems are similar to components in definition

https://aframe.io/docs/1.1.0/introduction/entity-component-system.html
Component

```javascript
AFRAME.registerComponent('foo', {
  schema: {
    bar: {type: 'number'},
    baz: {type: 'string'}
  },
  init: function () {
    // Do something when component first attached.
  },
  update: function () {
    // Do something when component's data is updated.
  },
  remove: function () {
    // Do something the component or its entity is detached.
  },
  tick: function (time, timeDelta) {
    // Do something on every scene tick or frame.
  }
});

<a-entity foo="bar: 5; baz: bazValue"></a-entity>
```

https://aframe.io/docs/1.1.0/introduction/writing-a-component.html
queriers

babiaxr-querier_json

<a-entity id="data" babiaxr-querier_json="url: ./data.json;"></a-entity>

babiaxr-querier_es

babiaxr-querier_github
babiaxr-filterdata
(for filtering) - WIP

<a-entity babiaxr-filterdata="from: data; filter: name=David" /></a-entity>
mapping

**babiaxr-vismapper**

(maps data properties to visualization properties)
Visualizations

1. Pie
2. Doughnut
3. 3D/2D bars
4. 3D/2D cylinders
5. Bubbles
1. Pie
2. Doughnut
3. 3D/2D bars
4. 3D/2D cylinders
5. Bubbles
6. CodeCity
Charts

Pie chart          Bars chart          3D Bars chart          Bubbles chart

Multiple charts
Mixing of different type of charts

Cylinder charts    3D Cylinders chart    Doughnut chart
Components API

https://gitlab.com/babiax/aframe-babia-components/-/blob/master/README.md

Components User Guide

https://gitlab.com/babiax/aframe-babia-components/-/blob/master/docs/HOW_TO_CHARTS_WITH_QUERIER.md
CodeCity

https://babiaxr.gitlab.io/aframe-babia-components/examples/codecityjs/basic/
**Time evolution**

**City Layout**
From present to past and from past to present
Empty zones = files that used to exist/will exist

**UI Navbar**
For moving between time snapshots

Date: 6/16/2020
commit: 1bbefb30117f279be9adb91f40e76735802fb1a3
**Configuration**

- **data**: mandatory, the data retrieved from the 2. section.
- **absolute**: Absolute size (width and depth will be used for proportions)
- **width**: Width of the entire city.
- **depth**: Depth of the entire city.
- **split**: Algorithm to split rectangle in buildings: naive, pivot
- **farea**: Field in data items to represent as area
- **fmaxarea**: Field in data items to represent as max_area
- **fheight**: Field in data items to represent as area
- **titles**: Titles on top of the buildings when hovering
- **buffered**: Use buffered geometries (improves performance)
- **building_color**: Color of the buildings
- **base**: build the base or not
- **base_color**: color of the base
- **border**: Size of border around buildings (streets are built on it)
- **extra**: Extra factor for total area with respect to built area
- **zone_elevation**: Zone: elevation for each “depth” of quarters, over the previous one
- **unicolor**: Unique color for each zone
- **wireframe**: Show materials as wireframe
- **time_evolution_delta**: Time evolution time changing between snapshots
- **time_evolution_init**: Time evolution time changing between snapshots
- **time_evolution_past_present**: Time evolution direction
- **ui_navbar**: UI navbar ID if exists.
Configuration

Time evolution commit by commit city with ui navbar:

```html
<a-entity position="0 0 -3" id="codecity"
    babiaxr-codecity="width: 20; depth: 20; streets: true; color: green;
    extra: 1.5; base_thick: 0.3; split: pivot; titles: true; time_evolution_delta: 500;
    data: main_data.json; time_evolution_init: data_0; ui_navbar: navigationbar">
</a-entity>
```

https://gitlab.com/babiaxr/aframe-babia-components/-/blob/master/docs/HOW_TO_TIME_EVOLVE_CITY.md
Get the Data

1. Graal

2. ElasticSearch

3. main_data.json

Tutorial: https://gitlab.com/babiaxr/aframe-babia-components/-/tree/master/tools
https://gitlab.com/babiaxr/aframe-babia-components/-/blob/master/docs/EXAMPLE_CREATE_CITY_WORKFLOW.md
Islands (representing code as islands, WIP)
Terrain

**Terrain Examples**

- Terrain Elevation
- Terrain Elevation Filled

**Totem** (for dynamically changing between set of data)

... 

More coming soon!
bonus
Contribute

**GPLv3 License**

**Contributing Guide**

https://gitlab.com/babiaxr/aframe-babia-components/-/blob/master/docs/contributing.md
BabiaXR
Virtual Reality Data Visualizations using only Front-End

David Moreno
dmoreno@bitergia.com
dmorenolumb@gmail.com

https://babiaxr.gitlab.io