

GENETICS, SOFTWARE

LEAFLET.JS – LAYER ORDER, LAYER ADDRESS AND LINKS

13.08.2015

Leaflet is great for mapping in epidemiology with quick results of just cut & pasting a few lines. Problems do start, however, whenever running a more advanced project. It's a pain, as plugins overwrite functions and basic css layouts. Or layers do not allow clickable links (as propagation is being prohibited). Or geojson data that are rejected for whatever reason. [A showcase project](#), that had been planned for 2 days, took more than 1 week as the documentation is frequently unclear, incomplete and often hard to understand without any (jsfiddle) example. Numerous Google searches helped, as well as peaking into the source-code, while also other stack overflow posters have been very helpful.

The following four problems took me several days (with everything referring to [leaflet.js 0.7.3](#)) Take care to have NO plugins enabled for testing. And it makes always sense to declare variables before putting the main map setup into some document ready brackets.

```
$(document).ready(function() {  
    initmap();  
})  
var markers = new L.MarkerClusterGroup...
```

Links in popups

Doesn't work out of the box. [Stackoverflow helped](#): Basically a hack, see the comments by Asad, as this is not good practice.

```
var link = $('<a href="#"  
class="speciallink">TestLink</a>').click(function() {  
    alert("test");  
})[0];  
marker.bindPopup(link);
```

Order of layers

Cryptic documentation of z-indices. [Stackoverflow helped](#): Undocument feature.

```
//Put explicit zIndexes on your mbstyle and mblabels layers:
var mbstyle = L.mapbox.tileLayer('statecodemog.aa380654', {
  'zIndex': 1
});
var mblabels = L.mapbox.tileLayer('statecodemog.798453f5', {
  'clickable': 'false',
  'zIndex': 99
});
//Then tell your L.control.layers not to mess around with the
zIndexes:
L.control.layers(baseLayers, groupedOverlays, {
  'autoZIndex': false
}).addTo(map);
```

Addressing of layers

Unclear documentation. [Blocks.org helped](#): Again a hack, the original code is missing a comma.

```
var MyMarkerGroup = L.FeatureGroup.extend({
  // override the 'getLayerId' function to return your ids rather than
  leaflets internal ids
  getLayerId: function(layer){
    return layer.id;
  },
  getLayersById: function(arrayOfLayerIds){
    var layers = [];
    // some logic to loop over layers and select them by id
    for (var i = arrayOfLayerIds.length - 1; i >= 0; i--) {
      var id = arrayOfLayerIds[i];
      layers.push(this._layers[id]);
    }
    return layers;
  }
});
var markers = new MyMarkerGroup();
var marker = new L.marker(latlng, options);
marker.id = "something"; // assign your id
markers.addLayer(marker); // add your layer
var aMarker = markers.getLayer("something"); // gets marker by id
var someMarkers = markers.getLayersById(["foo", "bar"]); // get
markers by their ids
```

Error in geojson output

Basically not a leaflet problem. But something that can be easily tested at
<http://geojsonlint.com>.

Cool stuff: [d3.js integration](#) and [R data processing](#)...