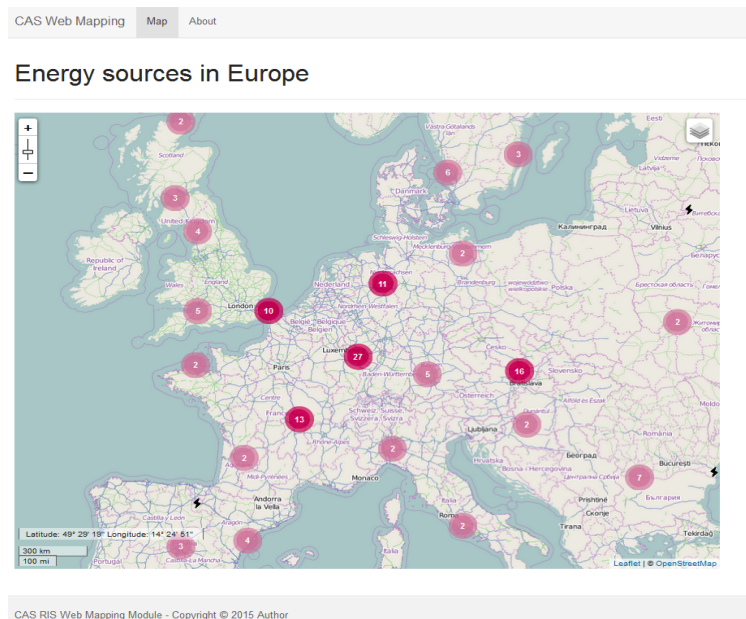


Optional Exercise A: Clustering in Leaflet

The main goal for this optional exercise is to display the distribution of nuclear power plants in Europe using a symbolization based on clusters at small scales, which then progressively separates the features at larger scales. Therefore, please study in detail the documentation of the Marker Clustering plugin for Leaflet, available at: <https://github.com/Leaflet/Leaflet.markercluster>.



We will be working with the same GeoJSON file from exercise 4, however we will use the procedure from exercise 2, for linking the feature collection of nuclear power plants as a script. Therefore, similarly as in exercise 2, make a copy of the GeoJSON data file, change its extension into .js, edit the file and add “var geojsonNPP =” in front of the feature collection inside the file and then finally link the resulted .js file as a script in your map's index.html. Please do not forget to also link the plugin.

Based on the documentation, we see that in order to use the plugin we have to create a new MarkerClusterGroup, to add our markers to it, and then add it to the map:

//loading the nuclear power plants using a js file for the power plant as a MarkerClusterGroup

```
var markers = new L.MarkerClusterGroup({
  showCoverageOnHover: false,
  maxClusterRadius: 80});
markers.addLayer(L.geoJson(geojsonNPP));
map.addLayer(markers);
```

Study the parameters offered by the plugin and feel free to add additional options or to change the colors defined in the CSS file of the plugin (Marker.Cluster.Default.css lines 1 to 20)

Then add a pop-up with additional information that is displayed by defining the function for the pop-up window similarly to exercise 4, and then use it with the “onEachFeature” option:

```
markers.addLayer(L.geoJson(geojsonNPP,{
  onEachFeature: popUpNPP}));
```

Finally, use a custom icon for the markers:

```
// icon for the single markers
var cityIcon = L.icon({
  iconUrl: 'icon/flash.png',
  iconSize: [20, 20],
  iconAnchor: [0, 0]
});

var markers = new L.MarkerClusterGroup({
  showCoverageOnHover: false,
  maxClusterRadius: 80
});
markers.addLayer(L.geoJson(geojsonNPP,{
  onEachFeature: popUpNPP,
  pointToLayer: function(feature, latlng) {
    return L.marker(latlng,{icon:cityIcon})
  }
}));
map.addLayer(markers);
```