

# WebMapping

Webmapping = « Cartographie en ligne » : solution d'affichage et/ou d'analyse de l'information géographique sur internet qui repose sur les services web.

## Les principales solutions libres de WebMapping

- **GoogleMaps**: Fournisseur de services de carte et fournit aussi une API de webmapping.
- **Bing Maps**: Fournisseur de services de mapping et propose aussi
- **ESRI ArcGIS** : Fournisseur / API
- **OpenStreetMap**: Fournisseur
- **OpenLayers** : API
- **Leaflet** : API

## Création d'une carte

Une carte est composée de couches (layers).

la première couche est la carte de base

les attributs de la carte peuvent être fournis dans le constructeur ou bien à l'aide des accessors

```
var carte = new ol.Map();
```

les attributs d'une carte

- target: l'id de l'élément html dans lequel la carte doit être affichée.
- layers: collection des couches dans la carte.
- view: définit les paramètres d'affichage de la carte

## Exemple OpenLayers

Structure du dossier

```
▼ exemple2
  <> index.html
  # ol.css
  # ol.css.map
  JS ol.js
  JS ol.js.map
  JS script.js
  # style.css
```

*fichiers openlayer*

*← code pour changer la carte*

*← style.*

### Fichiers

Index.html

```
<!DOCTYPE html>
<html lang="Exemple1">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemple 1</title>
  <link rel="stylesheet" href="ol.css">
  <link rel="stylesheet" href="style.css">
  <script src="ol.js"></script>
</head>
<body>
  <div id="viewDiv"></div>
  <script src="script.js"></script>
</body>
</html>
```

*emplacement d'affichage de la carte*

style.css

```
#viewDiv{
  height: 600px;
  width: 100%;
}
```

script.css

```
#viewDiv{
  height: 600px;
  width: 100%;
}
```

} style du div dans lequel la carte sera affichée

Script.js

```
var layer1=new ol.layer.Tile({
  source: new ol.source.OSM()
});
var vue=new ol.View({
  center:ol.proj.fromLonLat([-7,31]),
  zoom:0
})
var map=new ol.Map({
  target: "viewDiv",
  layers:[layer1],
  view:vue
});
```

← couche de base de la carte  
← source de la carte (OpenStreetMap & OpenLayers)  
} Vue qui définit les paramètres d'affichage de la carte.  
← id de l'élément qui contiendra la carte.

Interaction avec une carte et marqueurs.

```
let marqueur = new ol.Feature({
  geometry: new ol.geom.Point( ol.proj.fromLonLat(point))
});
```

} marqueur de type

```
let carre=new ol.style.Style({
  image: new ol.style.RegularShape({
    fill:new ol.style.Fill({color:'gold'}),
    stroke:new ol.style.Stroke({color:'black', width:2}),
    points:5,
  })
});
```

} style du marqueur

```
    radius:10,  
    angle: Math.PI/5  
  })  
});
```

```
marqueur.setId('esma');  
  
let source=new ol.source.Vector({  
  features:[marqueur]  
});  
marqueur.setStyle(carre);  
  
let couche=new ol.layer.Vector({  
  source: source,  
  maxResolution: 4  
})  
let layer1=new ol.layer.Vector({  
  source: source,  
  maxResolution: 4  
}) ;  
let base=new ol.layer.Tile({source:new ol.source.OSM()})  
  
let vue=new ol.View({  
  center:ol.proj.fromLonLat(esma),  
  zoom: 10});  
  
let carte=new ol.Map({  
  target:"carte",  
  layers:[base, layer1],  
  view: vue  
});
```

```
carte.addEventListener('click',(evt)=>{  
  
  carte.forEachFeatureAtPixel(evt.pixel, (feature,layer)=>{  
    console.log(feature.getId());  
    alert (feature.getId());  
    console.log(JSON.stringify(evt.pixel));  
  })  
});
```