

Programmation WEB

api (javascript) de googlemap

Programmation licence

IUT de Fontainebleau

14 avril 2015

Sommaire

- 1 Le début : afficher une carte
- 2 Overlays
- 3 Les événements
- 4 Geocodage

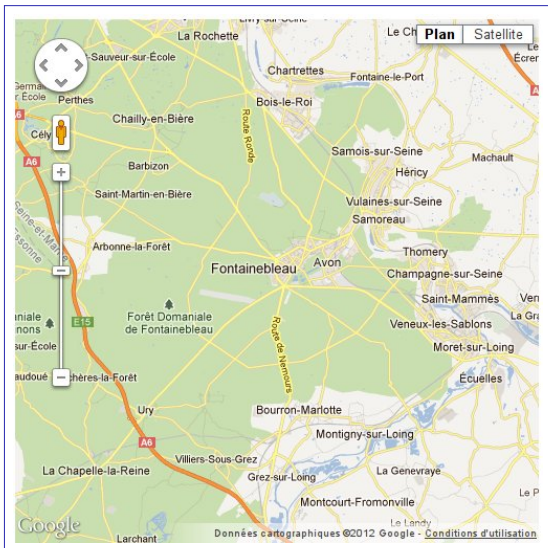
Sommaire

- 1 Le début : afficher une carte
- 2 Overlays
- 3 Les événements
- 4 Geocodage

Hello world

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no" />
    <script type="text/javascript"
      src="http://maps.googleapis.com/maps/api/js?sensor=false">
    </script>
    <script type="text/javascript">
      function initialize() {
        var myOptions = {
          center: new google.maps.LatLng(48.3990184,2.687389),
          zoom: 11,
          mapTypeId: google.maps.MapTypeId.ROADMAP
        };
        var map = new google.maps.Map(document.getElementById("map_canvas"),
          myOptions);
      }
    </script>
  </head>
  <body onload="initialize()">
    <div id="map_canvas" style="width:500px; height:500px"></div>
  </body>
</html>
```

Ce qui donne ...



- Le chargement de l'api :

```
<script type="text/javascript"  
src="http://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY  
      &sensor=SET_TO_TRUE_OR_FALSE">  
</script>
```

- Le div où s'affichera la carte :

```
<div id="map_canvas" style="width:500px;height:500px"></div>
```

- Creation et affichage :

```
var map = new google.maps.Map(  
  document.getElementById("map_canvas"),  
  myOptions);
```

avec les options :

```
var myOptions = {  
  center: new google.maps.LatLng(48.3990184,2.687389),  
  zoom: 11,  
  mapTypeId: google.maps.MapTypeId.ROADMAP  
};
```

- Le centre de la carte : on utilise ici le constructeur de la `LatLng` qui crée un point sur la carte à partir de sa latitude et longitude.
- Le facteur d'échelle : `zoom`.
- Le type de la carte : `google.maps.MapTypeId.ROADMAP`.
- Il y a (beaucoup) d'autres options.

Sommaire

- 1 Le début : afficher une carte
- 2 Overlays**
- 3 Les événements
- 4 Geocodage

Création d'un marker

Les overlays sont des éléments graphiques que l'on peut poser ou dessiner sur une carte Google Maps.

Les overlays auxquels on s'intéresse ici sont les suivants :

- les marqueurs qui permettent d'indiquer un point sur la carte à la manière d'un drapeau
- les polylines qui permettent, par exemple, de faire des tracés de vols d'avion
- les polygones qui permettent de dessiner sur la carte une zone géographique

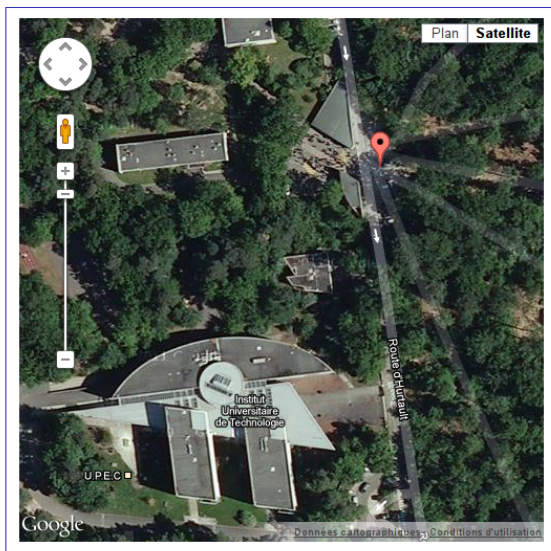
Makers

Il représente par le biais d'un icône un point précis sur la carte :

Création

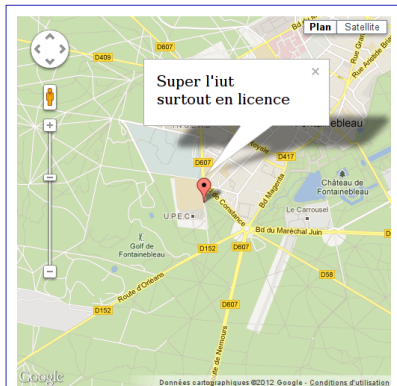
```
var myLatLng=new google.maps.LatLng(48.3990184,2.687389);  
var marker = new google.maps.Marker({  
  position: myLatLng,  
  map: map,  
  title:"près de l'iut !",  
  draggable:true,  
  animation: google.maps.Animation.DROP  
});
```

Notez les propriétés `draggable` et `animation`.



Bulle d'information

```
var infowindow = new google.maps.InfoWindow({
  content: "Super l'iut<br/>surtout en licence"
});
infowindow.open(map,marker);
```



trajectoire

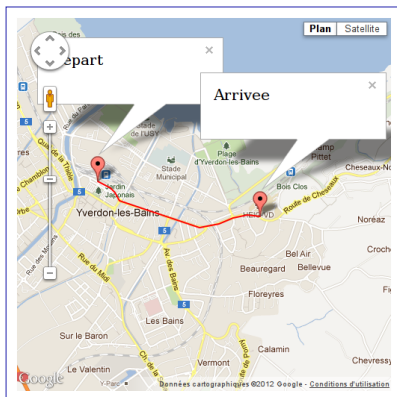
Les polylines permettent de dessiner des segments attachées les unes aux autres sur la carte. Ceci peut permettre, par exemple, de dessiner un itinéraire sur la carte Google Maps.

Un exemple

```
var parcoursBus = [  
  new google.maps.LatLng(46.781367900048, 6.6401992834884),  
  new google.maps.LatLng(46.780821285011, 6.6416348016222),  
  new google.maps.LatLng(46.780496546047, 6.6421830461926),  
  new google.maps.LatLng(46.779835306991, 6.6426765713417),  
  new google.maps.LatLng(46.777748677169, 6.6518819126808),  
  new google.maps.LatLng(46.778027878803, 6.6541349682533),  
  new google.maps.LatLng(46.778484884759, 6.6557324922045),  
  new google.maps.LatLng(46.778752327087, 6.6573654211838),  
  new google.maps.LatLng(46.778605381016, 6.6588674582321)  
];  
var traceParcoursBus = new google.maps.Polyline({  
  path: parcoursBus, //chemin du tracé  
  strokeColor: "#FF0000", //couleur du tracé  
  strokeOpacity: 1.0, //opacité du tracé  
  strokeWeight: 2 //grosseur du tracé  
});  
traceParcoursBus.setMap(map);
```

On peut inscrire la trajectoire dans un rectangle "optimal" avec `LatLngBounds` qui représente une zone rectangulaire sur la carte.

```
zone=new google.maps.LatLngBounds();  
for (i in parcoursBus) {  
    zone.extend(parcoursBus[i]);  
}  
map.fitBounds(zone);
```



Il y a bien d'autres possibilités en ce qui concerne les *overlays*. Il y a même la possibilité de dessiner sur la carte.

Sommaire

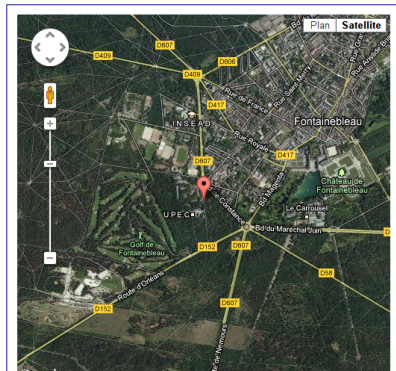
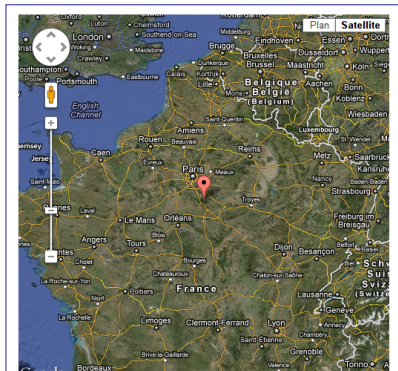
- 1 Le début : afficher une carte
- 2 Overlays
- 3 Les événements**
- 4 Geocodage

On enregistre une fonction réflexe exécutée à la notification d'un événement grâce à la méthode

`addListener`

Voici un exemple, où un clic sur le marker provoque un zoom.

```
google.maps.event.addListener(marker, 'click', function() {
  map.setZoom(14);
});
```



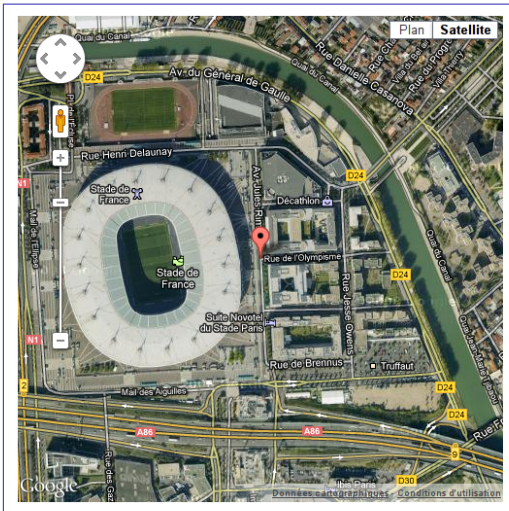
Sommaire

- 1 Le début : afficher une carte
- 2 Overlays
- 3 Les événements
- 4 Geocodage**

L'api fournit la classe geocoder qui permet de géocoder (transformer une adresse en ses coordonnées)

Avenue Jules Rimet, Saint Denis

```
geocoder = new google.maps.Geocoder();
geocoder.geocode( { 'address':
  "Avenue Jules Rimet,93200, SAINT-DENIS"},
  function(results, status) {
    if (status == google.maps.GeocoderStatus.OK) {
      map.setCenter(results[0].geometry.location);
      map.setZoom(16);
      var marker = new google.maps.Marker({
        map: map,
        position: results[0].geometry.location
      });
    } else {
      alert("Erreur : " + status);
    }
  });
```



Il existe également un service web à cette adresse :

<http://maps.googleapis.com/maps/api/geocode/output?parameters>