

Creating a Web Based Map using Google Maps API

Background

Geospatial data viewing tools and associated API's (Application Programming Interface) are used to create web based maps. Among various available tools, the most popular ones are by Google, Microsoft and Yahoo. One of the main advantages of using these tools is that, base data (ex. aerial imagery, street network, etc.) are already stored remotely on web servers and are made available to users. Google Maps are especially popular, because of the open API they provide which allow for rapid customization of the map application. Although, base data are available, users need to store the application data in XML or database format (for large datasets). This tutorial provides the steps of creating a simple web based map using Google Maps (API), HTML and JavaScript.

Note: This tutorial assumes that a website is available for implementing the HTML/Javascript code presented here. To create your own website on MASON server please visit: <http://itusupport.gmu.edu/STG/webpage.asp>

1. Getting a Google Maps Key

Google Maps API is a free service. The API key can be obtained by signing-up [here](http://code.google.com/apis/maps/signup.html) (<http://code.google.com/apis/maps/signup.html>). Save this unique web directory key by **copy** → **paste** (in notepad) for use in developing Google maps applications.

2. In your destination web page (ex. Index.html) and within the scripttag `<script></script>` give a reference to the Google Maps library and the map key as:

```
<head>
```

```
<script src = "http://maps.google.com/maps?file=api&v=2&key=Map Key" type = "text/javascript">
```

```
</head>
```

Note: The above line of code is to be inserted after the standard doc type declaration which tells the browser the details of the markup language and the best way to render it, for example:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>
```

```
<html xmlns=http://www.w3.org/1999/xhtml>
```

3. Create a place holder for your map using the `<div>` tag within the `<body>` of your code.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>
```

```
<html xmlns=http://www.w3.org/1999/xhtml>
```

```
<html>
```

```

<head>
<script src = "http://maps.google.com/maps?file=api&v=2&key=Map Key" type= "text/javascript">
</head>
<body>
<div id = "map" style="width: 800px; height:800px"></div>
</body>
</html>

```

4. All the remaining code of creating and customizing your Google map, including any scripts will be placed within the <body> tag after the place holder is set.

a. Create (instance of) google map using a constructor method ex: Gmap()

```
var map = new GMap2(document.getElementById("map"));
```

b. Center the map and set the zoom level to 10, for example (Google allows a zoom range between 0-17)

```
map.setCenter(new GLatLng(38.82936 , -77.30969), 16);
```

c. Adding basic map controls for zooming and base data selection

```
map.addControl (new GLargeMapControl ());
```

```
map.addControl (new GMapTypeControl ());
```

d. Adding points of interest and pop-up information window

```
var pt = new GLatLng (a, b); [Creates a point of interest centered at a (latitude), b(longitude)]
```

```
var marker = new GMarker(pt); [Creates a Symbol for the point of interest]
```

```
map.addOverlay(marker); [Overlay point of interest on base map]
```

```
var dist = new GSize (18, -22); [Specifies spacing for information window]
```

```
var info = "George Mason University<br>Go Patriots!"; [Content of the Information window]
```

```
map.openInfoWindowHtml(pt,info,dist); [Place the info window relative to the point of interest]
```

Code:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<script src = "http://maps.google.com/maps?file=api&v=2&key=Map Key "
type="text/javascript">
</script>
</head>
<body>
<div id="map" style="width: 680px; height: 600px"></div>

```

```

<script type="text/javascript">

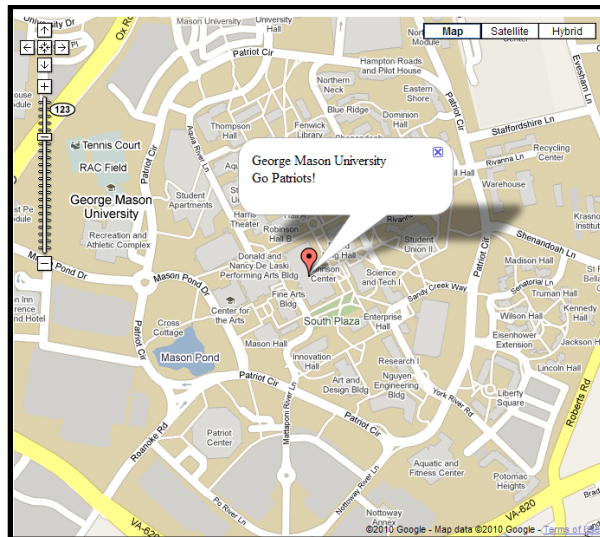
    var map = new GMap2(document.getElementById("map"));
    map.setCenter(new GLatLng( 38.829889 , -77.307454),16);
    map.addControl(new GLargeMapControl ());
    map.addControl(new GMapTypeControl ());
    var pt = new GLatLng(38.829889 , -77.307454);
    var marker = new GMarker(pt);
    map.addOverlay(marker);

</script>

</body>

</html>

```



5. Adding **Event Listener** for executing embedded content (images/video, etc) on map markers

Step 1. Define the embedded content with HTML as its value

Examples:

```

var content1 = '<b> Ex: Use your camera to stitch the world: Photosynth</b><br><iframe
src="http://photosynth.net/embed.aspx?cid=fb94bb5-5ee2-44d2-9986-345064a0b2ab" height="300"
width="400"></iframe>'

```

Or:

```
var content1 = '<b> Ex: Video</b><div id="popup"><object type="application/x-shockwave-flash" style="width:400px; height:300px;" data="http://www.youtube.com/v/nh1JSPaXyZc"><param name="movie" value="http://www.youtube.com/v/nh1JSPaXyZc" /></object></div>'
```

Note: Most of the online multimedia and graphics objects have the embed code available for inclusion on your personal websites. The embedded content that is especially easy to acquire, are usually hosted on public and networking websites such YouTube, Picassa among others.

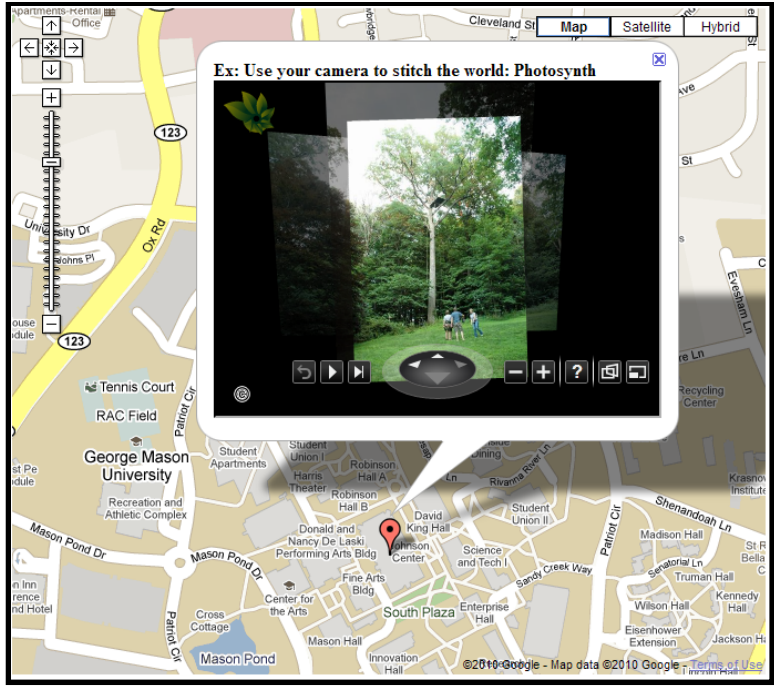
Step 2. Add an event listener and specify the method's parameters as **target, event, and function**

```
GEvent.addListener(marker, "click", function() {  
  
    marker.openInfoWindowHtml(content1);  
  
});
```

Note: This displays the HTML information in the balloon when you click on the marker

Code:

```
</script>  
  
    </head>  
    <body>  
        <div id="map" style="width: 680px; height: 600px"></div>  
  
        <script type="text/javascript">  
  
            var map = new GMap2(document.getElementById("map"));  
            map.setCenter(new GLatLng( 38.829889 , -77.307454), 16);  
            map.addControl(new GLargeMapControl ());  
            map.addControl(new GMapTypeControl ());  
            var pt = new GLatLng(38.829889 , -77.307454);  
            var marker = new GMarker(pt);  
            map.addOverlay(marker);  
  
            var content1 = ' ' // add the content from the examples above  
            GEvent.addListener(marker, "click", function()  
            {  
                marker.openInfoWindowHtml(content1);  
            });  
  
        </script>
```



Please contact GIS Librarian at George Mason University, if you have any questions on this. Email: hsuh1@gmu.edu
December 2010