

# Использование Google Maps для публикации точечных данных и треков

Как использовать сервис Google Maps для публикации точечных данных на своем интернет сайте.

[Обсудить в форуме](#) Комментариев — 16

[Google Maps](http://maps.google.com/) <http://maps.google.com/> - картографический сервис работающий в режиме web страницы. Данный сервис представляет возможность легко размещать окно карты на своих страницах ([подробнее](#)).

Данная статья развивает эту тему и иллюстрирует как можно, дополнительно к данным Google Maps, использовать также свои точечные данные, в нашем случае - соединенные в линию-трек. Подразумевается, что Читатель знаком с установкой и настройкой Google Maps на своей странице, если это не так, рекомендуем сначала ознакомиться со статьей "[Размещение Google Maps на своей интернет странице](#)". В нижеследующей инструкции вопросы затронутые в этой статье вторично не рассматриваются.

Приводимые инструкции иллюстрируют конкретный пример, если ваши данные имеют другую природу, возможно, что может понадобится некая адаптация приводимого кода под ваш конкретный случай.

Для начала работы необходимо наличие файла с точечными данными в формате XML. Существуют и другие форматы данных, с которыми можно работать используя платформу Google Maps.

## 1. Модификация блока <html></html>

Для корректного отображения линейных объектов в Internet Explorer, необходимо, чтобы тэг <html> имел следующий вид:

```
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">
```

## 2. Модификация блока <body></body>

Так как наш метод связан с загрузкой данных, необходимо прописать в тэге <body> следующие функции, ответственные за загрузку данных по открытии страницы и их выгрузку по окончанию работы:

```
<body onload="load()" onunload="GUnload()">
```

## 3. Загрузка данных

Основные операции с картой перед ее отображением осуществляются с помощью подпрограммы на языке Javascript. Эта подпрограмма содержит несколько процедур, первая из которых - загрузка данных пользователя **function load()**. Данная процедура, помимо создания собственно объекта типа GMap2, включает также создание условного значка, используемого для отображения каждой точки:

```
<script type="text/javascript">
//
```

Далее идет операция собственно загрузки данных, загружаемые данные собираются в массив **points**, согласно соответственно значениям lat и long хранящимся в XML файле с исходными данными:

```
GDownloadUrl("/test.xml", function(data) {
var xml = GXml.parse(data);
var markers = xml.documentElement.getElementsByTagName("marker");
var numpoints = markers.length - 1;
var points = ;
for (var i = 0; i < markers.length; i++) {
var point = new GLatLng(parseFloat(markers[i].getAttribute("lat")),
parseFloat(markers[i].getAttribute("lng")));
points.push(point);
}
```

## 4. Получение географического охвата данных

Для того, чтобы отобразить на карте ВСЕ точки, необходимо правильно задать ее географический охват. На практике это означает, что географический охват карты должен быть равен географическому охвату всех точек из файл XML, например прямоугольник, определенный крайними значениями южной широты, северной широты, восточной и западной долготы. К сожалению прямой процедуры получить такой прямоугольник нет, но его достаточно просто вычислить зная координаты точек. Для этого надо отсортировать их массив сначала по долготе, а потом по широте и получить первое и последнее значение (экстремумы) для соответственно широты и долготы, из этих 4 координат нужно создать 2 точки (соответственно левую нижнюю и правую верхнюю), которых достаточно, чтобы создать прямоугольник (объект GLatLngBounds), который потом использовать для вычисления масштаба карты (уровня увеличения - зума):

```
var points2 = points.slice();
points2.sort(function(p1, p2) {
return p1.lng() - p2.lng();
});
var west = points2[0].x;
var east = points2[numpoints-1].x;
points2.sort(function(p1, p2) {
return p1.lat() - p2.lat();
});
var north = points2[numpoints-1].y;
var south = points2[0].y;
var sw = new GLatLng(south, west);
var ne = new GLatLng(north, east);
var bounds = new GLatLngBounds(sw, ne);
var centerpoint = new GLatLng((north + south)/2, (east + west)/2);
var zoomlevel = map.getBoundsZoomLevel(bounds);
```

Вычисленный уровень увеличения затем нужно будет задать созданному в самом начале объекту карты:

```
map.setCenter(centerpoint, zoomlevel, G_SATELLITE_MAP);
```

## 5. Отображение точек

Визуализация точек производится с помощью функции AddOverlay, с использованием созданного ранее значка (icon):

```
for (var i = 0; i < (numpoints + 1); i++) {
map.addOverlay(new GMarker(points[i], icon));
}
map.addOverlay(new GPolyline(points));
```

## 6. Соединение точек в линию

Точки можно соединить линией, если это необходимо, при этом соединяться они будут в том же порядке, в котором они идут в исходном XML файле:

```
map.addOverlay(new GPolyline(points));
```

## 7. Результат

[Посмотреть пример карты](#) созданной с помощью данной технологии, но с некоторыми дополнениями. В исходном коде этой страницы можно посмотреть подробности реализации. Результирующий код программы

отображения подобной карты, должен выглядеть примерно вот так:

```
function load() {
  if (GBrowserIsCompatible()) {
    var map = new GMap2(document.getElementById("map"));
    map.addControl(new GLargeMapControl());
    map.addControl(new GMapTypeControl());
    map.addControl(new GScaleControl());

    var icon = new GIcon();
    icon.image = "http://labs.google.com/ridefinder/images/mm_20_red.png";
    icon.shadow = "http://labs.google.com/ridefinder/images/mm_20_shadow.png";
    icon.iconSize = new GSize(12, 20);
    icon.shadowSize = new GSize(22, 20);
    icon.iconAnchor = new GPoint(6, 20);
    icon.infoWindowAnchor = new GPoint(5, 1);

    GDownloadUrl("/test.xml", function(data) {
      var xml = GXml.parse(data);
      var markers = xml.documentElement.getElementsByTagName("marker");
      var numpoints = markers.length - 1;
      var points = ;
      for (var i = 0; i < markers.length; i++) {
        var point = new GLatLng(parseFloat(markers[i].getAttribute("lat")),
                                parseFloat(markers[i].getAttribute("lng")));
        points.push(point);
      }
      var points2 = points.slice();
      points2.sort(function(p1, p2) {
        return p1.lng() - p2.lng();
      });
      var west = points2[0].x;
      var east = points2[numpoints].x;
      points2.sort(function(p1, p2) {
        return p1.lat() - p2.lat();
      });
      var north = points2[numpoints].y;
      var south = points2[0].y;
      var sw = new GLatLng(south, west);
      var ne = new GLatLng(north, east);
      var bounds = new GLatLngBounds(sw, ne);
      var centerpoint = new GLatLng((north + south)/2, (east + west)/2);
      var zoomlevel = map.getBoundsZoomLevel(bounds);
      map.setCenter(centerpoint, zoomlevel, G_SATELLITE_MAP);
      for (var i = 0; i < (numpoints + 1); i++) {
        map.addOverlay(new GMarker(points[i], icon));
      }
      map.addOverlay(new GPolyline(points));
    });
  }
}
//>
```

Все остальные части кода, необходимые для того, чтобы установить Google Maps на свой сайт описываются [вот в этой статье](#).

Если у Вас есть замечания или предложения, мы будем рады добавить недостающую информацию к этой статье. Вы можете от свои комментарии используя форму справа или по одному из контактов.

[Обсудить в форуме](#) Комментариев — 16

## Ссылки по теме

- [Картографический сервис Google Maps. Общая информация](#)
- [Размещение Google Maps на своей интернет странице](#)

Последнее обновление: December 29 2010

Дата создания: 19.02.2007

Автор(ы): [Максим Дубинин](#)