

# 腾讯位置服务 JavaScript API 使用手册

14331400

庄建平

## 1. 使用前

要使用腾讯位置服务 API，首先需要申请开发密钥。这里提供一个我已经申请好的密钥：

MMSBZ-UYZW6-NECSR-MACZX-TEV4J-YSFV7

应用名称 / 描述	应用类型	key	操作
基于云服务的车辆监控系统 实训项目	浏览器	MMSBZ-UYZW6-NECSR-MACZX-TEV4J-YSFV7	<a href="#">设置</a>

## 2. 加载 API 服务

要使用腾讯位置服务 API，首先需要在页面的前端使用 **script** 标签加载 API 服务，格式如下：

```
<script charset="utf-8" src="http://map.qq.com/api/js?v=2.exp&key=YOUR_KEY"></script>
```

调用 js 的 API 没有次数限制，因此可以不使用 key（请无视上一步）。

```
<script charset="utf-8" src="http://map.qq.com/api/js?v=2.exp "></script>
```

JavaScript API 除了提供基本的功能库外，还提供了一些有用的附加库。加载方法是将 script 标签的 src 设置为：

```
http://map.qq.com/api/js?v=2.exp&key=YOUR_KEY&libraries=drawing,geometry,autocomplete,convertor
```

```
http://map.qq.com/api/js?v=2.exp&key=YOUR_KEY&libraries=drawing,geometry,autocomplete,convertor
```

其中 **libraries** 参数用来指明加载的附加库，可以指定多个附加库名称，名称之间用英文半角字符逗号","分隔。支持的附加库种类：

**drawing** 绘图工具库，用于在地图上方绘制折线、多边形、圆等几何图形；

**geometry** 几何运算库，用于计算地球表面的距离、面积等；

**autocomplete** 文本输入提示库，用于快速帮助用户完成检索关键词输入；

**convertor** 坐标转换库，将标准经纬度或其它地图 API 经纬度转换为腾讯经纬度坐标系。

## 3. 一些需要注意的地方

坐标转换：

如果原始数据是使用的标准经纬度，或者其它地图，比如 Google、Baidu 等地图坐标系，那在使用腾讯地图显示前，需要进行坐标转换，否则显示位置将会和实际位置有偏差。

目前支持转换的坐标类型有：

GPS 经纬度

搜狗经纬度

百度经纬度  
Mapbar 经纬度  
Google 经纬度  
搜狗墨卡托

若要使用转换接口，需加载附加库 `convertor`，详见“2.加载 API 服务”。

坐标格式：

JavaScript API 使用的坐标格式是{纬度, 经度}，即**纬度在前，经度在后**，如果在使用 API 过程中，发现地图无法显示，请先查看坐标格式是否正确。从其它几家 API 切换过来的用户尤其要注意这个区别。

地图级别范围：

JavaScript API 使用的二维地图和卫星影像图，级别范围都是[1,18]。

## 4. 使用方法

开发指南：[http://lbs.qq.com/javascript\\_v2/guide-start.html](http://lbs.qq.com/javascript_v2/guide-start.html)，在这里可以看到 API 的大致用法；

详细参考网站：[http://lbs.qq.com/javascript\\_v2/doc/index.html](http://lbs.qq.com/javascript_v2/doc/index.html)，在这里可以找到详细的 API 文档。

示例：[http://lbs.qq.com/javascript\\_v2/demo.html](http://lbs.qq.com/javascript_v2/demo.html)，在这里可以找到 API 的用法示例。

经纬度对象的创建：`new qq.maps.LatLng(float, float)`

a) 创建一个地图。

地图的创建，最基本的参数有两个：1，html 中的哪个地方用来放地图；2，地图的中心点的经纬度。

示例：[http://lbs.qq.com/javascript\\_v2/case-run.html#sample-map](http://lbs.qq.com/javascript_v2/case-run.html#sample-map)。

要使地图在网页上显示，必须为其在网页中留出一个位置。通常，我们通过创建名为 `div` 的元素并在浏览器的文档对象模型 (DOM) 中获取此元素的引用执行此操作。

例如，在 html 中定义这样一个 `div`：

```
<div id="container" style="width:500px; height:300px"></div>
```

然后，在 js 中这样构造一个 map：

```
var map = new qq.maps.Map(document.getElementById('container'));
```

如果这样构造，那么在使用这个 `map` 之前，还需要执行一个操作，即将其初始化（设置中心点）。初始化通过地图的 `panTo()` 方法完成。`panTo()` 方法需要 `LatLng` 经纬度坐标，并且调用此方法必须在对地图执行任何其它操作（包括设置地图本身的任何其它属性）之前。

当然也可以在构造函数中传入中心点参数：

```
var map = new qq.maps.Map(document.getElementById("container"), {
    center: new qq.maps.LatLng(39.916527, 116.397128), // 地图的中心地理坐标。
    zoom: 8
});
```

b) 获取鼠标点击处的经纬度，可用于设置起点终点

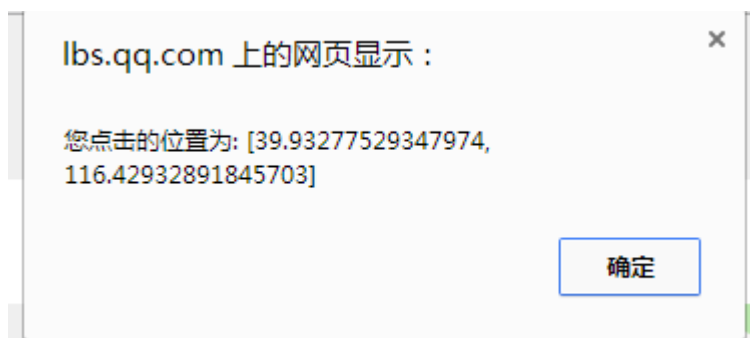
一个事件需要的参数有：1，监听的对象（在本次示例中，监听的是一整个地图，即 map 对象）；2，事件的类型（如单击、双击）；3，回调函数 callback。

示例：[http://lbs.qq.com/javascript\\_v2/case-run.html#sample-event-click-argument](http://lbs.qq.com/javascript_v2/case-run.html#sample-event-click-argument)。

典型的一段代码如下：先创建地图对象 map，然后调用 addListener 函数。addListener 函数的参数代表：这个函数作用于这个 map 对象，事件为单击，回调函数为 alert 点击位置的经纬度。

```
var map = new qq.maps.Map(document.getElementById("container"), {
    center: new qq.maps.LatLng(39.916527, 116.397128),
    zoom: 13
});
//绑定单击事件添加参数
qq.maps.event.addListener(map, 'click', function(event) {
    alert('您点击的位置为: [' + event.latLng.getLat() + ', ' +
        event.latLng.getLng() + ']');
});
```

运行结果：



c) 创建标记，可用于表示车辆位置

一个标记需要的参数有：1，地图对象的引用；2，标记的位置（经纬度）。

i. 添加标记：

[http://lbs.qq.com/javascript\\_v2/guide-overlay.html](http://lbs.qq.com/javascript_v2/guide-overlay.html)。

```

1  var center = new qq.maps.LatLng(39.914850, 116.403765);
2  var map = new qq.maps.Map(
3      document.getElementById("container"),
4      {
5          center: center,
6          zoom: 13
7      }
8  );
9  var marker = new qq.maps.Marker({
10     position: center,
11     map: map
12 });

```

如上，制定 center 的经纬度为(39.914850, 116.403765)，创建地图，然后在 center 的位置创建了一个 marker，结果如下：



- ii. 除了创建  这样一个 marker，也可以创建文本标注 Label:

[http://lbs.qq.com/javascript\\_v2/case-run.html#sample-overlay-addlabel](http://lbs.qq.com/javascript_v2/case-run.html#sample-overlay-addlabel).

```

var center = new qq.maps.LatLng(39.916527, 116.397128);
var map = new qq.maps.Map(document.getElementById('container'), {
    center: center,
    zoom: 13
});
var label = new qq.maps.Label({
    position: center,
    map: map,
    content: '示例：文本标注'
});

```

运行结果：



iii. 此外，还可以给标记添加点击事件：

[http://lbs.qq.com/javascript\\_v2/guide-overlay.html](http://lbs.qq.com/javascript_v2/guide-overlay.html).

```
1 qq.maps.event.addListener(marker, 'click', function() {
2     info.open();
3     info.setContent('<div style="text-align:center;white-space:nowrap;" +
4     'margin:10px;">单击标记</div>');
5     info.setPosition(center);
6 });
```

与 c 中提到的事件一样的原理，不一样的是，这里监听的对象是 `marker`，而不是 `map`。

iv. 添加折线，可用于表示路径：

[http://lbs.qq.com/javascript\\_v2/guide-overlay.html](http://lbs.qq.com/javascript_v2/guide-overlay.html).

```
1 var polyline = new qq.maps.Polyline({
2     path: [
3         new qq.maps.LatLng(39.910, 116.399),
4         new qq.maps.LatLng(39.920, 116.399),
5         new qq.maps.LatLng(39.930, 116.399)
6     ],
7     strokeColor: '#000000',
8     strokeWeight: 10,
9     map
10 });
```

运行结果：



d) 路径规划:

[http://lbs.qq.com/javascript\\_v2/doc/drivingservice.html](http://lbs.qq.com/javascript_v2/doc/drivingservice.html)。

（网站上的代码运行会后提示出错，并且不绘制规划路线。问题可能出现在 `select` 标签上，经过检查发现其无法获得 `start` 的 `value`。其他部分的代码是没有问题的。）

路径规划需要调用 `drivingService` 类，`drivingService` 类需要 `map` 对象来构造。

用 `drivingService.setComplete` 来设置回调函数。这里回调函数的意义是：如果查询到的起点、终点不唯一，就重新调用 `search` 来 `search` 第一组起点、终点。（为什么查询到的起点、终点可能是不唯一的？因为起点、终点的输入类型可以为 `string`，比如汉字地名，这样的地名可能有多种结果）

`drivingService.search` 执行完毕后，就会执行上述回调函数，最终在地图上显示规划路线。

```

var center = new qq.maps.LatLng(39.916527, 116.397128);
var map = new qq.maps.Map(document.getElementById("container"), {
    center: center
});
//设置获取驾车线路方案的服务
var drivingService = new qq.maps.DrivingService({
    map: map,
});
//设置搜索地点信息、驾车方案等属性
function search() {
    var start = new qq.maps.LatLng(23.253964, 113.134460);
    var end = new qq.maps.LatLng(23.142886, 113.435211);
    //设置驾车的区域范围
    drivingService.setLocation("北京");
    //设置回调函数
    drivingService.setComplete(function(result) {
        if (result.type == qq.maps.ServiceResultType.MULTI_DESTINATION) {
            //alert("起终点不唯一");
            var d = result.detail;
            drivingService.search(d.start[0], d.end[0]);
        }
    });
    //设置检索失败回调函数
    drivingService.setError(function(data) {
        alert(data);
    });
    //设置驾驶路线的起点和终点
    drivingService.search(start, end);
}
window.onload = search;

```

运行结果如下:



e) 坐标转换:

[http://lbs.qq.com/javascript\\_v2/case-run.html#sample-converor-library](http://lbs.qq.com/javascript_v2/case-run.html#sample-converor-library)。

要使用坐标转换，首先得加载 `converter` 类，具体请参照“2.加载 API 服务”。

```

//转换百度坐标为腾讯坐标
qq.maps.converter.translate(new qq.maps.LatLng(39.911082, 116.396135), 3, function(res) {
    latlng = res[0];
    var marker = new qq.maps.Marker({
        map: map,
        position: latlng
    });
});

```

函数具体参数为：1，要被转换的非腾讯坐标；2，要被转换的坐标的类型，1 代表 gps 经纬度，2 代表搜狗经纬度，3 代表百度经纬度，4 代表 mapbar 经纬度，5 代表 google 经纬度，6 代表搜狗墨卡托；3，回调函数，上图中的回调函数的参数 `res` 即为转换后得

到的坐标。