Archived:How to use Google Static Maps data in mobile applications

This article explains how to use Google Maps data in a mobile application. Google Maps offers REST services that allow accessing its data with simple HTTP requests, so they can be easily integrated into mobile applications.

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

13 Jul 2008

Since this article was written, alternative **Map Tile caching** solutions for Java ME have become available. **Map Tile caching** solutions such as the Nokia Maps API for Java ME have several advantages over **static mapping** solutions such as the static Google Maps API including:

- **Static** mapping services such as the Google Static Maps API or Nokia's RESTful Map API do not cache or tile the images when requested, therefore **each** request involves a round trip to the server. If the map on a mobile application needs to be refreshed at any time, using a caching library will result in a reduction in network traffic after around **three** maps have been displayed. An explanation of this can be found here
- As the name implies, Google's Static Maps API can only retrieve over http static images for a requested coordinate point, image size, image type and zoom level. Newer libraries offer additional functionality out of the box offering dynamic Map content and touch support, where the user can move around his/her current position, zoom in, zoom out, modify the view mode to satellite or translate an address to a coordinate point and show that on the map, among others. This abstraction of the underlying functionality is hidden from the developer,os much less coding is needed in order to achieve the same result .

Additionally the following points apply in favour of Nokia Maps when comparing to Google Maps:

- No legal restrictions of using the API outside a web browser application or need to provide a link to the native Google Maps App (if there is one), or to Google Maps (if there isn't one). See Terms of Service below.
- Higher free daily request limits. Nokia Maps API for Java ME supports up to 50,000 render requests per day and per unique IP address (as of January 2012), for Nokia Developer registered users (free of charge) while the limit for Google's Static Maps API is currently 1000 unique (different) image requests per viewer per day.

An article with the same functionality, written with Nokia Maps API for Java ME, that uses much less code can be found here

Sign up for a Google Maps API key

NOTE: Usage of this code with the free Google Maps API Key breaks Google's Terms and Conditions (section 10.8). You should purchase an Enterprise License if you wish to use the Google Maps API as shown in this example.

First you need to sign up on this page:
http://code.google.com/apis/maps/signup.html
Once you have signed up, you get a key (a simple string) that you can use for all your queries to Google Maps services.

Google Static Maps API no longer requires a Maps API key

For updated information on Google Static Maps API, please visit http://code.google.com/apis/maps/documentation/staticmaps/

Static maps

Standard Google Maps code is suitable for Web applications. However, it includes a lot of Ajax functionalities that are not really useful if you are building a mobile application. The solution is to use the static maps service that allows retrieving single images that can easily be used in mobile applications.

The static maps service supports different image formats (png32, GIF, JPG) and customizable image size, so you can get perfect images for all purposes. For example, if you want to retrieve the location at:

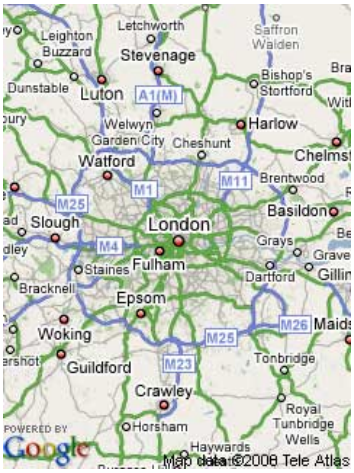
- latitude: 41.867878
- longitude: 12.471516

You can simply retrieve this URL with an HTTP GET request:

```
http://maps.google.com/staticmap?center=41.867878,12.471516&format=png32&zoom=8&size=240x320&key=<API_KEY>
```

This way you will get a PNG32 image with a width of 240 pixels and a height of 320 pixels, centered at point (41.867878,12.471516), and with a zoom level of 8 (the zoom range is from 0 to a maximum level of 19)

(C) Copyright Nokia 2013. All rights reserved.



Geocode an address

From Google Maps docs:
Geocoding is the process of converting addresses (such as "1600 Amphitheatre Parkway, Mountain View, CA") into geographic coordinates (like latitude 37.423021 and longitude -122.083739)

The following example describes building an application that displays the address typed by the end user. First you need to geocode its address into geographic coordinates.
To do this, Google Maps offers another REST service that can easily be accessed with simple HTTP requests.

If you want to geocode this address
Leicester Square, London
Request this URL from your code

```
http://maps.google.com/maps/geo?q=Leicester%20Square,%20London&output=csv&key=<API_KEY>
```

and you will get this output:

```
200,6,51.510605,-0.130728
```

Where:

- The first number is a code, which in this case (200) means that geocoding has been successfull (for a full list of status codes see: [1])
- The second number gives a measure of the geocoding accuracy (from 0 to 9 - maximum accuracy)
- The 3rd and 4th numbers represent latitude and longitude of the geocoded address, so these are the coordinates used to retrieve the map through the static map service.

As you can see, there is an 'output' parameter in the geocode request. This means that you can choose the output format you prefer. The supported formats are:

- xml
- kml (same as xml, but with different Content-Type)
- json (not really useful for mobile apps)
- csv (comma-separated values)

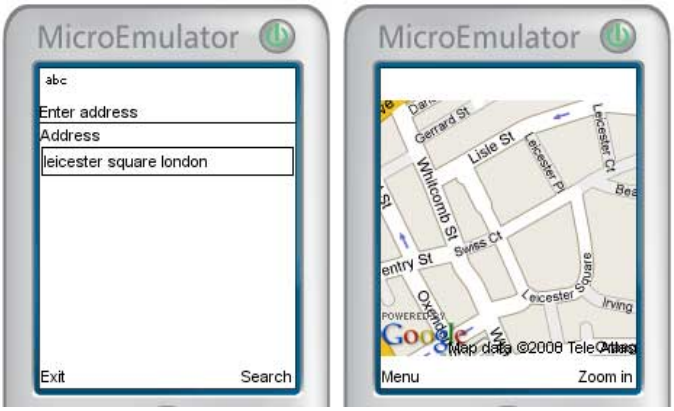
Proxy server, usage limits

Since your Google Maps API key is bound to a specific URL, in order to access map services **you need to setup a proxy server** that will receive HTTP requests from the mobile application and forward them to Google Maps REST URLs, returning Google responses to mobile clients. *(as pointed out in the Comment page, this is not a fully clear point yet)*

Also, be aware that there is a **limit to the number of requests**, both for static maps and geocode service, you can do each day. For personal uses they are more than enough, but you need to keep this issue in mind if you plan to develop commercial services.

Sample application

 (C) Copyright Nokia 2013. All rights reserved.



A sample J2ME application, using the approach described here, is available on this page: Google Maps J2ME Test

Google Maps J2ME API source code used in this example is also available here: Google Static Maps API in Java ME

Related content

- Google Static Maps API in Java ME
- Location API
- How to get Location Using Location API JSR 179

(C) Copyright Nokia 2013. All rights reserved.

 (C) Copyright Nokia 2013. All rights reserved.