*International Conference on Computer and Information Application（ICCIA 2010）*

# The Study and Implementation of Mobile GPS Navigation System Based on Google Maps

He Li[1], Lai Zhijian[2]

College of Information Engineering
North China University of Technology
Beijing, China
[1] yudianhl@tom.com
[2] laizhijian@ncut.edu.cn

***Abstract*—Google Maps API provides a number of utilities for adding individual content to the Google map and various web map applications can be explored based on Google Maps API. Assisted global positioning systems (A-GPS) is adopted by mobile operation companies as the highest accurate positioning way in mobile location based services. This paper proposes a solution of mobile navigation system that realizes such functions as Google map browse and query, bus lines search, the rapid local positioning on your mobile phone etc. In this paper, we discuss the system technical scheme and the key realization technologies. At last, the test result is displayed on Nokia N73. The results prove that the solution of mobile navigation system based on Google Maps is feasible and valid, and it can be installed in PDA, mobile phones and other portable information terminals.**

***Keywords-mobile navigation system; Assisted global positioning systems (A-GPS); Google Maps API***

## I. Introduction

The Google Company declared a free Web Map Service---Google Map in 2005. Google Maps can display map images, topographic maps and satellite images, and can achieve global location search, classified information access, traffic information query, driving directions lines and even street scene three-dimensional model and so on. Besides this, Google also provides API to users for secondary development.

Moreover, with the development of mobile communication and the popularity of mobile phone users, mobile phone has become one of the main means of obtaining information. This makes it possible to combine the mobile communication technology and Google map and GPS. Now almost all mobile phones support J2ME, and the developed programs based on JAVA has better portability and system compatibility, which greatly reduces the cost of mobile application development.

Therefore, this paper present a mobile navigation system solution based on Google Maps API which applies to the popular cell phones supporting Java. The solution provides a convenient and economical way of technology for users on the exchange, access, and share of information based on electronic map query, bus line search, the rapid location positioning for mobile phones.

## II. Technical Background

The research of mobile navigation system is along with the development of computer network, Assisted GPS positioning technology, and Google Maps API etc.

### *A. A-GPS Technology*

Assisted GPS, generally abbreviated as A-GPS, is a system which can, under certain conditions, improve the startup performance or TTFF (Time to First Fix) of a GPS satellite-based positioning system.

Assisted GPS describes a system where outside sources, such as an assistance server and reference network, help a GPS receiver perform the tasks required to make range measurements and position solutions. The assistance server has the ability to access information from the reference network and also has computing power far beyond that of the GPS receiver. The assistance server communicates with the GPS receiver via a wireless link. With assistance from the network, the receiver can operate more quickly and efficiently than it would unassisted, because a set of tasks that it would normally handle is shared with the assistance server. The resulting AGPS system, consisting of the integrated GPS receiver and network components, boosts performance beyond that of the same receiver in a stand-alone mode [1].

### *B. Google Maps API Technology*

Google Maps API is a JavaScript technology-based application programming interface (API) offered by Google. It provides a lot of processing maps service and adding content to maps service. The Google Maps API allows users to embed Google Maps in their own Web pages.

The GMap2 class is the core class of Google Maps API. Objects of this class define a single map on a web page. Google creates a new instance of this class by the JavaScript new operator

The Google Maps API is fully integrated with the Google AJAX API. This framework allows us to load one API key for all supported Google AJAX APIs (including Google Maps) and also provides a common namespace for each API, allowing different Google APIs to operate together [2].

Google Maps requires the first-time customers to register for a Google Maps API secret key through a Google user account [3].

978-1-4244-8598-7 /10/$26.00 © 2010 IEEE

## III. SYSTEM ANALYSIS AND DESIGN

### A. System function design

The objective of this mobile navigation system is to allow users to use cell phones to obtain constant, dynamic bus lines or location information. The whole system function work flow diagram is shown in Figure1.

The system contains the following modules: bus lines query module, map query, local orientation, map zoom in or out etc.

The client IDE programming language and environment is MyEclipse 5.5.1, and Web server is Tomcat5.5. The testing mobile is Nokia N73.

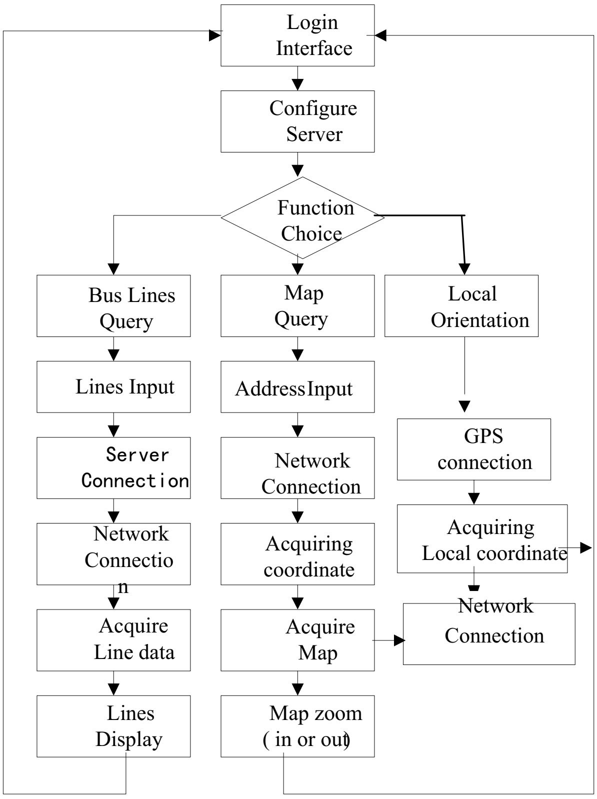


Figure 1. The system function work flow diagram

### B. Map query module design

The processing steps of map query module are as follows:

(1) Input one or two address name (Chinese or English), and select the initial maps size.

(2) Judge the query methods: single or two locations inquires.

(3) Convert the address name to the UTF8 format, and query its coordinates.

(4) Combine the map page URL

(5) Connect HTTP and acquire map page

(6) Convert map page to image and load image on cell phones

(7) Browse the map and zoom in or out the map.

The processing flow diagram of map query module is shown in Figure2.

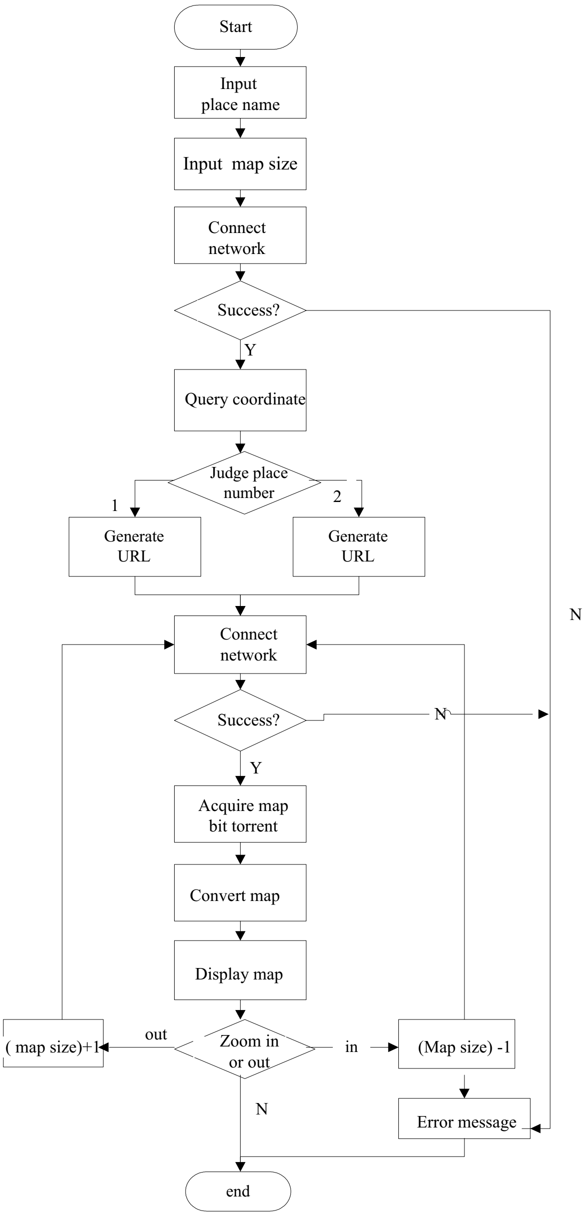


Figure 2. The processing flow diagram of map query module

### C. Bus lines query module design

The processing steps of bus lines query module are as follows:

(1) Select query methods according to bus lines or station name.

(2) Inputting inquiring conditions.

(3) Connect Web server according to the server address, port setting, and service page information.

(4) Obtain relevant line information through query string.

(5) Send line information back to cell phone.

(6) Display line information

This function module requires the correct server setting, such as the server information, and tomcat WEB server is in the start-up status.

The web server setting parameters are shown in Table I.

TABLE I. SERVER SETTING PARAMETERS

| Name | Default Value | Note |
|---|---|---|
| Server address | 192.168.1.100 | Server IP address |
| Server port | 8080 | HTTP Server port: 8080 |
| Server Page | /BQServlet/query .do | the servlet URL in The web.xml file |

### D. Local orientation module design

The processing steps of local orientation module are as follows:

(1) Select "local orientation" query function.

(2) System startup GPS (or A-GPS) function, and acquire highly, longitude, latitude of the location.

(3) Call the single address query function of the “address query” module, and form the map page URL.

(4) Connect HTTP and acquire map page

(5) Convert map page to image and load image on cell phones

(6) Browse the map and zoom in or out the map.

## IV. SYSTEM IMPLEMENT TECHNIQUES

The following is the implementation of map query module.

Firstly, we should register a Google Maps API secret key, in this mobile navigation system, the registered key is:

*ABQIAAAAli-bcTq8jGCP7KoQ2ANSThSUaLh9GPAk0u374U4nx-kcR8j1VhSxfbQujJrCkn4D-oEbseE0ejlWBw.*

Then, we create a "map" object by programming and make the map in the center based on a given geographic point and configure user interface.

Lastly, initialize the map object.

Thus, our system can display the Google Maps. However, the map cannot be zoomed in or out. Each map contains a number of attributes that may be inspected or set. To carry out these functions, map-controls are added to increase the ability to control maps. In this system, we added the following control options: GLargeMapControl, GOverviewMapControl and GScaleControl etc.

Now we take the map query module as an example and describe the key implementation technology.

### A. coordinates acquire technology according to the inputting place name

The following codes are specific solutions.

```
public double[] geocodeAddress(String address) throws Exception {
    byte[] res = loadHttpFile(getGeocodeUrl(address));
    String[] data = split(new String(res), ',');
    if (!data[0].equals("200"))
    {int errorCode = Integer.parseInt(data[0]);
    throw new Exception("Google Maps Exception: "
        + getGeocodeError(errorCode)); }
    return new double[] { Double.parseDouble(data[2]),
    Double.parseDouble(data[3]) };
    }
```

### B. Generating URL technology

The following codes are specific solutions.

```
private String getGeocodeUrl(String address) {
return "http://ditu.google.com/maps/geo?q="+
urlEncode(address) + "&output=csv&key=" + apiKey;
}
```

### C. Data conversion technology in the process of HTTP transmission

Due to the limit of mobile phones and J2ME setting, in many cases, using special data types and patterns of transmission is often needed. So we should realize the function of data conversion: (1) the conversion from input text characters to UTF8 format. (2) The conversion between characters streams and bit streams. (3) The conversion from bit streams to map images.

The followings are specific solutions.

- Conversion from text characters to UTF8 format

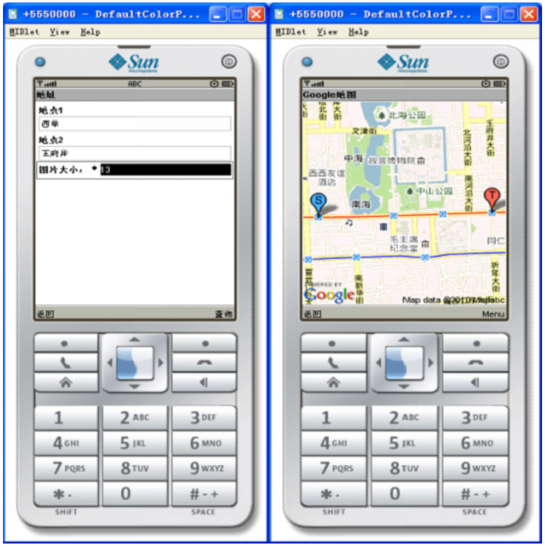
The realization codes are as follows:

```
private static String urlEncode(String str)
 { StringBuffer buf = new StringBuffer();
      byte[] bytes = null;
      try
         {ByteArrayOutputStream bos = new
ByteArrayOutputStream();
      DataOutputStream dos = new
DataOutputStream(bos);
      dos.writeUTF(str);
      bytes = bos.toByteArray();         }
      catch (IOException e) {}
      for (int i = 2; i < bytes.length; i++)
      {byte b = bytes[i];
      if (URL_UNRESERVED.indexOf(b) >= 0) {
      buf.append((char) b); }
      else {    buf.append('%').append(HEX[(b >> 4)
& 0x0f]).append(    HEX[b & 0x0f]); }}
```

- The conversion from bit streams to map images

```
/* one address query */
      public Image retrieveStaticImage(int width, int
height, double lat,
double lng, int zoom, String format) throws IOException
 {/* lat variable denoting longitude，the lng variable
denoting latitude */
byte[] imageData = null;
imageData = loadHttpFile(getMapUrl(width, height, lng,
lat, zoom, format));
return Image.createImage(imageData, 0,
imageData.length);      }
/* two addresses query */
      public Image retrieveStaticImage(int width, int
height, double lat1,
        double lng1, double lat2, double lng2, int zoom,
String format)
        throws IOException {
        byte[] imageData = null;
        imageData = loadHttpFile(getMapUrl(width,
height, lng1, lat1, lng2, lat2, zoom, format));
        return Image.createImage(imageData, 0,
imageData.length);
    }
```

A case of map query module is shown as Figure 3.



地址

地点1

西单

地点2

王府井

图片大小： 13

Figure 3. A case of map query module on Nokia N73

After inputting the first place and the second place, and choosing the display map size, a path marked by the orange line is displayed in the map page center.

## V. CONCLUSION

By use of Google Maps API, this paper develops a mobile navigation system realizing map query and local orientation. This system can apply to the popular cell phones supporting Java. The test results on Nokia N73 show the navigation system solution is valid and feasible. However, there are many issues to be further researched deeply, such as the more efficient spatial query algorithm and the more accurate map match etc.

## ACKNOWLEDGMENT

This paper is supported by Beijing Student Research and Innovation Program (1018).

## REFERENCES

[1] http://www.gpsworld.com/gps/assisted-gps-a-low-infrastructure-approach-734

[2] Google Corporation, Jan. 2009; “Map Basics,” http://code.google.com/intl/en/apis/maps/documentation/

[3] Google Corporation, Jan. 2009; “Google Maps API,” http://code.google.com/intl/en/apis/maps/.

[4] Meiyu Zhang , JingHua Mei , Design and Implementation for Mobile-based GIS Urban Information System, ISISE 2008:464-467

[5] He Li, Fan Jin-hong. Research and Implementation of Mobile Navigation System Based on GPS and GIS, ICISE2009:507-510

[6] Thurston, J., Poiker, T.K. and J. Patrick Moore. (2003). Integrated, Geospatial Technologies: A Guide to GPS, GIS, and Data Logging. Hoboken, New Jersey: Wiley

[7] http://baike.baidu.com/view/7125.htm?fr=ala0_1_1 , 2010-06-09

[8] http://wiki.forum.nokia.com/index.php/Google_Maps_API_in_Java_ME#Get_your_own_Google_Maps_API_Key , 2010-3-30