Evacuation Routing Out of Tsunami Hazard Zones

Martin SCHUSTER[1] and Christopher GOMEZ[2]

[1]Carinthia Univiversity of Applied Sciences, Villach/Austria · martin.schuster@edu.fh-kaernten.ac.at
[2]Department of Geography, University of Canterbury, Christchurch/New Zealand

This contribution was double-blind reviewed as full paper.

Abstract

This paper describes the development and implementation of a location based service for tsunami evacuation routes. Based on the position of the mobile phone the system provides a real-time calculation for an escape route out of the hazard zone. A smartphone application visualizes the route on a map. Nowadays, tsunami hazard zones are well known and determined with complex simulation models. Evacuation routes in such hazard zones are mostly sign posted along roads and visualized in evacuation plans. People who are inside a tsunami hazard zone have to make themselves familiar with the evacuation plans and have to find their own way out of the zone. The developed concept shows the possibility of a location based service to determine an escape route and visualize the result on a map. In addition, the route is the shortest possible way out of the zone. The concept is implemented for a test area in Christchurch, New Zealand. The developed prototype consists of a three tier architecture. The mobile Android application (client) determines the position of the smartphone and is responsible for the visualization of the result. A calculation program on the server side calculates the shortest route out of the hazard zone. The necessary data for this calculation come from a spatial data base which is the third component in this architecture. The whole system allows the user to calculate an escape route out of the hazard zone at the touch of a button. Implemented in an existing tsunami warning application it would be a life-saving service.

1   Introduction

Disaster management is unique among Geographic Information System (GIS) applications because it deals directly with loss of human life and property damage. On December 26th 2004, an example of poor disaster management was recognized after a massive 9.0 earthquake in the Indian Ocean. A horrifying tsunami destroyed coastline areas of 11 countries and caused an unbelievable number of deaths. By that date, people from around the world realized the power of nature and how devastating hazards and loss can occur by underestimating its power. To create a comprehensive disaster management system, our society needs to rely on advanced geo-spatial technologies and services. Mobile GIS is one of the most vital technologies for the future development of disaster management systems. Mobile GIS and mobile Geographic Information Services extend the capability of traditional GIS to a higher level of portability, usability and flexibility (TSOU & SUN 2006).

A study in Bangladesh established that wireless mobile technologies can be used in disaster information management and the results showed that mobile technology may be used to disseminate pre-disaster warnings and post-disaster announcements (CHOWDHURY et al. 2005). Mobile devices, especially smartphones, have an essential status in our life. Almost everybody in industrial countries has a smartphone. A study of Gartner (GARTNER INC. 2010) predicts the sales of OS (Operating System) devices are expected to surpass the 1 billion mark by 2015. Furthermore, the OS Android will move to become the most popular OS worldwide and will build on its strength to account for 49 percent of the smartphone market by 2012 (GARTNER INC. 2011). Current tsunami evacuation methods are mostly signs to mark evacuation zones, evacuation routes and evacuation safe locations (DARIENZO 2003). Tsunami warnings are broadcast on local radio, television stations and prompt the people inside a zone to evacuate it as quick as possible. Furthermore, there exists e.g. a new tsunami awareness application for smartphones. The goal of this application is to get informed about the towns a tsunami could hit and where evacuation assembly zones are (FRIEDERICH 2012). This type of information is preventive, i.e. only accessible before the event in order to prepare the user (thus, not reactive, i.e. after the event). In turn, the main disadvantage of this application is that the users have to prepare their evacuation route by themselves. An application which provides an evacuation route depending on the position of the mobile phone (in other words "the current location of the user") would be a helpful decision-support for each user. As part of a project at the Department of Geography at the University of Canterbury in Christchurch, New Zealand a location based service application named "Tsunami Evacuation Routing and Information Service" (TRIS) was developed. This service provides the calculation and visualization of an escape route out of a tsunami hazard zone.

2   TRIS System Concept

Considering that users of mobile devices need specific personalized information tailored to their current location, LBSs (Location Based Services) provide exactly such information. Within this chapter a development concept of a location-based service to show the user an escape route based on the current smartphone position is presented. Referring to RAPER (2007) most of the LBS solutions available use a tiered client-server approach. There are three basic elements in client-server architecture. The first element is the data storage in a Database Management System (DBMS). The second element is the service provider consisting of the communication handling and the data process. The third component is a graphical presentation through a user interface on a mobile client.

2.1   Data, Data Source and Data Schema

Geospatial data are an important component of each LBS. Based on the data requirements for LBS (WILMERSDORF 2007) the important data for this model are:

- Geodetic position of the user
- Tsunami hazard zone
- Evacuation target points
- Street network including distance and minimum travel speed (SCHEER, et al. 2011).

The user position is the key element in LBSs and will be discussed later. The data source for tsunami hazard zones could be either the result of inundation models or an already existing georeferenced, digitized evacuation map. Also the evacuation target points are a result of such inundation studies and can also be found in evacuation maps. Another data component is the street network. This network is the topological connection between the geodetic position of the user and the evacuation target points. The length or distance of a street segment as well as the minimum travel speed, are essential attributes to calculate the shortest or fastest path. The digital sources for street networks are for example TIGER Navtech as well as OpenStreetMap. For further determinations in the calculation model the street network must consist of street segments and street intersections (SCHILLER & VOISARD 2004). Table 1 shows a database schema for the essential data. The hazard zone is represented as a polygon. Each street segment of the street network is a single line with attributes of length and speed between two street intersections. The evacuation target points are a subset of the street intersections to allow the shortest path calculation to an evacuation point. This data are managed in a spatial database accessed by the service provider.

Table 1: Database schema

Data set name	Primary Key	Attributes	Geometry
hazard zone	zone_Id	name	Polygon
street segment	segment_Id	length, speed	Line
street intersection	intersection_Id		Point
evacuation target point	target_Id	name, description, intersection _Id	Point

2.2 Service Provider

The communication handling and data processing is part of the service provider. The service provider, including a calculation model to determine the next evacuation target point and a routing service call to calculate the route from the user position to the evacuation target point, is used to resolve the user query and provide the tailored response to the smartphone. The position of the smartphone, which is sent via Internet from the client, is used by the service provider. These coordinates are the input parameter for the calculation model. Apart from this calculation model, the service provider handles the database queries as well as the routing service request. The determination steps of the calculation model can be described as follows: The calculation model starts with the validation of the user location. If the location is outside the hazard zone, the model will be terminated. The fundament of this calculation model is the street network consisting of street segments and street intersection. If the user is inside the hazard zone the position in the street network has to be determined. Therefore the current location of the user has to be "snapped" to the next street intersection of the network. This step is essential to calculate distances inside the street network. After this step a graph of the street network is created. Each segment is an edge and each intersection is a node in the graph. Different routing algorithms could calculate the shortest path between the defined starting point and the evacuation target points. An example for a routing algorithm is Dijkstra's algorithm (PUNTAMBEKAR 2008) which calculates the shortest path to each node in the graph based on positive edges weight. Another example is the Floyd-Warshall or Bellman-Ford algo-

rithm (CORMEN et al. 2009) which is quite similar to Dijkstra's algorithm but it allows also negative edges weight. The coordinates of the next evacuation target point are the output of this calculation model. To facilitate the navigation out of the zone the route needs additional navigation information (e.g. turn left into Ilam Road). This information is a single instruction at each turn point. Several routing services exist which provide the calculation and additional information of the route such as Google Maps, MapQuest or CloudMade. These services need coordinate- or address data to calculate a route from a start to an end point. In this case the user position and the calculated evacuation target point are the parameters for this routing request. The response is a calculated route which is sent back to the smartphone with coordinates for each directional change.

2.3   Mobile Client

The mobile client application for smartphones handles several tasks but first the smartphone must have the following requirements.

- GPS receiver and/or
- Wi-Fi receiver
- Permanent Internet connection

The GPS and the Wi-Fi receiver are important for the position determination and the permanent Internet connection is necessary for the map visualization as well as for the service provider request. The first task of the mobile client is to get the position of the smartphone. OS of smartphones offer an interface to get access to the location of the device. The second task is to send the coordinates of the position to the service provider and interpret the response.
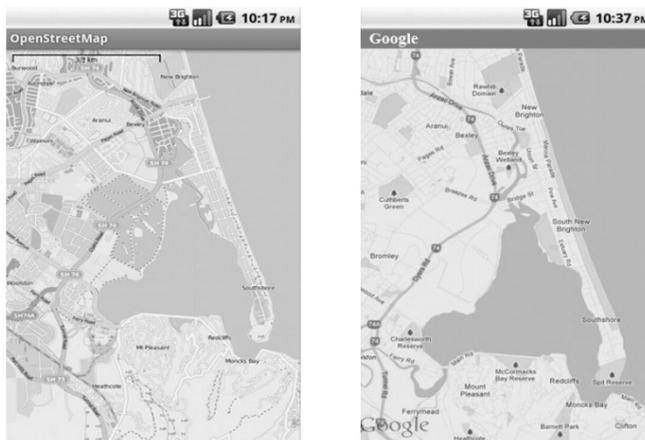
Fig. 1:   OpenStreetMap and Google Maps in Smartphone applications

The third task is the visualization of the route and its information with a map added for a better orientation. There are different providers for mobile maps such as Google maps and OpenStreetMap, which are very well known map services for Android and iPhone. An alternative to these two services are Navtec and Microsoft Bing Maps for Windows mobile.

These maps can be integrated into the mobile application by an external map library. Figure 1 shows and example of the two main maps for Android and iPhone smartphones. Once a map is integrated in a mobile application, the external map library allows the developer to add powerful mapping capabilities to the application. Such libraries contain classes which offer a variety of display options and controls (add overlays, zooming, etc.). The route and user position as well as the evacuation target point could be displayed as map overlays. Line segments between the coordinate points could be drawn on the map to display the route. For the evacuation target point and the user position a symbol could be used. With these overlays and the additional instructions at each turning point, the user should be able to navigate out of the tsunami hazard zone.

3   Implementation of TRIS

3.1   Data

For this project a study area in Christchurch, New Zealand is defined. It is located on the east coast of Christchurch and contains the suburbs New Brighton, South New Brighton and Southshore. Especially for this region the Christchurch City Council developed an evacuation map. This region is vulnerable for regional- and distant source tsunamis. After an unsuccessful data research for this test region all necessary data are manually digitized. The basis for the digitized hazard zone was the evacuation map on the website of the Christchurch City Council (CIVIL DEFENCE AND EMERGENCY MANAGEMENT 2012). Microsoft Bing Maps satellite images were the basis for the digitization of the street network which consists of single street segments (lines) and street intersections (points). The evacuation target points are a result of the intersection of the hazard zone boundary and the street segments. After digitization the data were imported into a PostGIS (POSTGIS 2012) database. PostGIS is a spatial extension of a PostgreSQL database like Oracle's Spatial Extension. PostGIS follows the Open GIS "Simple Feature Specification for SQL" and consists of special functions and data types. Quantum GIS offers an interface to import spatial data into a PostGIS database. In addition a table graph is created to store the relation between the streets and the street intersections (table 2). With this table the relations between the street segments and street intersections, the geometry and the length of the edge are stored as well. It should be mentioned that each edge has exact two incident nodes.

Table 2:   Attributes of the database table Graph

Graph				
id_edge	Length	id_node1	id_node2	edge_geometry

3.2   Service Provider

The service is implemented as a Java servlet on an apache tomcat webserver. The data communication protocol between the mobile application and the server is the Hypertext Transfer Protocol (HTTP). HTTP defines nine methods; one of them is the GET method. This method was created to send input data to a server. This input data is the position of the mobile phone in the form of two request parameter; latitude and longitude. In addition, this

method sends a response message back to the mobile application. This message contains an Extensible Markup Language (XML) document with a textual route description as well as the coordinate points of the whole route. The following sub section describes the implementation of the calculation model and the routing service request.

Calculation Model

The calculation model to determine the next evacuation target point uses different database functions as well as Dijkstra's routing algorithm. Referring to the determination steps of the calculation model in figure 3 the first steps are provided by PostGIS spatial relationship and measurement tools (POSTGIS 2012). To check if the position is inside the hazard zone the function "ST_Within" is used. This function validates if any geometry A is completely inside a geometry B or not. As already mentioned the model terminates if the position is outside the hazard zone and the response message contains corresponding information for the user. The next step determines the user position within the network. This means that the start position for the calculation of the shortest path is not the real position of the user but rather the next intersection in the street network. To store the relevant information with the correct relation the ID and the geometry of street segments are selected and inserted into the table graph. The relations of the street intersections and segments are automatically generated by using a "before insert trigger" (KLINE et al. 2009) for the table graph. This trigger fills the columns id_node1, id_node2 and the length of the edge. After this step the data can be stored in a Java data structure for the routing algorithm.

```
function Dijkstra (graph, start node,)
        for each vertex v in graph
                d[v] := infinity;
                previous [v] := undefined;
        end for
        d[start node] := 0
        Q := V
        while Q is not empty:
                u := v in Q with smallest distance;
                remove u from Q;
                for each neighbor v of u:
                        alt := d[u] +distance between (u,v);
                        if alt < d[v]
                                d[v] := alt;
                                previous [v] := u;
                        end if;
                end for;
        end while
        return d[];
end Dijkstra
```

Fig. 2: Pseudo code notation of Dijkstra's routing algorithm adapted from MAHLMANN & SCHINDELHAUER (2007)

The used algorithm in this calculation model is Dijkstra's algorithm. Dijkstra uses a defined start node and calculates the shortest path to every other node in the graph. The algorithm is shown in figure 2 and works as follows: the graph in the Java data structure and the start node are the input parameters for this function. The first loop sets the distance of the nodes

and the previous node to *infinity* and *undefined*. The distance from the start node is set to zero and the priority queue Q contains all node elements V. The following statements are executed unless the priority queue is empty. The node u is defined as the v in Q with the smallest distance (in the first iteration the shortest distance is zero – the start node is used) and is removed from the priority queue. The distance to each neighbour is calculated and the node with the shortest distance is chosen. The actual start node is set as previous node of the chosen node. The algorithm terminates if the priority queue is empty. After Dijkstra's routing algorithm the distances between the evacuation target points are compared. The coordinates of the target point with the shortest distance is the output of this calculation model. The next step is the request to the routing service. Figure 3 shows an overview about the whole calculation model.

```
input <location>
    if <location is inside hazard zone> then
            <find next street intersection>
            <create graph from street intersections and street segments>
            <declare start node>
            <Dijkstra (graph, start node)>*
            <get evacuation target point from path with shortest distance>
            <get coordinates from target point>
            output <inside, target point>
    else
            output <outside>
```

Fig. 3:   Determination steps in the calculation model (* see Fig. 2)

Routing service

To be able to display an exact course of the route with additional navigation information (e.g. turn left at …) a routing service has to be called. As mentioned in the previous chapter different routing services can be found. OSM provides a HTTP-request to calculate a route between two points but the result was inacceptable because the route did not exactly follow the street on the map. In some places the route was not visualized along the road and in others it led across buildings. Therefore a Google Maps request is used. The most important parameters of this request (MAPKI 2012) are defined as follows:

- saddr: source address
- daddr: destination address
- dirflg: route type (driving directions, walking directions)
- output: output file

For the source and destination address the latitude and longitude of the location separated by a comma are used. Figure 4 shows an example request.

```
https://maps.google.com/maps?saddr=-43.523659,172.584689&daddr=-43.527479+172.573691&dirflg=w&output=kml
```

Fig. 4:   Example of Google Map request

The whole response is sent back to the mobile application and is used to visualize the escape route out of the tsunami hazard zone.

3.3   Mobile Client

The client of this three tier architecture is implemented as an application on an Android OS smartphone. Besides the route visualization which is explained later in this section the mobile application has to determine the position of the smartphone. The coordinates of the current position are obtained from the device's internal GPS receiver. To accomplish this, the Android Location Manager was applied that represents an interface to the internal GPS receiver of the mobile device. It is part of the core Android application framework, meaning that all user applications have access to the devices location (EDPEARSON 2007). With the device location the server application calculates the escape route out of the hazard zone. Therefore the position is sent via a HTTP request to the server application. Figure 5 shows an example of the request to the server.

```
http://www.servername/applicationname?type=position&lat=-43.523659&lng=172.584689
```

Fig. 5:   Example request to the server application

The response of the server is interpreted and the information is used to visualize the route. The Android framework APIs provides a set of 2D drawing APIs (GOOGLE INC. 2012) that allow developers to render a custom graphics onto a canvas. This Canvas class is used to draw a path onto the top of a Google map. This Google map is imported into the Android application from an external library (GOOGLE INC. 2012). The application shows two different views. On one hand a map to visualize the route as a path on a map and on the other hand a list to visualize the route information as text.
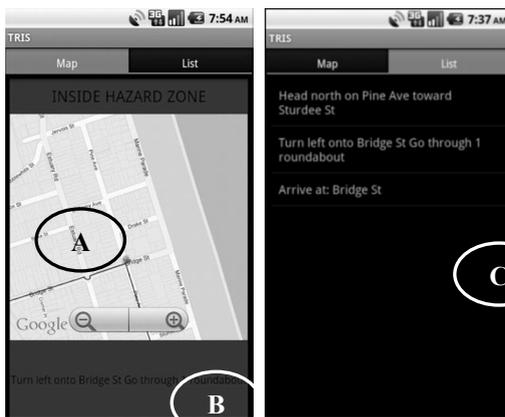
Fig. 6:   Map view of the mobile application

For a better understanding figure 6 shows the implemented application with a map (A), a description of the next change in direction (B) and another view with a list of the whole route description (C). In addition the navigation information such as turn left at Ilam Road, are added as proximity alerts (Google Inc. 2012) in the application. Proximity alerts are alerts at a specific location. For this case the alert is used to change the description of the route in the map view. At each turn point on the route the user gets textual information

about the further course. Further, a list view is implemented with an overview about the whole change of direction. If the user is outside the hazard zone the system changes the panel from red to green and signals the arriving at the destination.

4   Summary and Future Outlook

The implemented prototype shows that with the developed concept of TRIS, an escape route out of a tsunami hazard zone can be calculated and visualized. This mobile prototype allows the user at the touch of a button to send the position of the mobile phone to the server application. The calculation model determines based on the street network and Dijkstra's routing algorithm the next evacuation target point. With a HTTP-request to the Google Maps server the route is calculated and the result is sent back to the mobile application. For this first prototype only walking directions are used.

The developed and implemented LBS model determines an evacuation route based on the street network. This street network is not evaluated for emergency situations. Not every street or route is designed for emergency situations. Evacuation roads should have a minimum capacity for a fleeing crowd. This point should be considered in the data model in a future research. Also the transportation mode is important. It depends on the time frame between the point when the tsunami is triggered and the point when it hits the shore. Furthermore, the emergency shelters inside a hazard zone (horizontal and vertical) could be considered in the future. The developed model offers only a determination to an evacuation point outside the zone. These shelters could be admitted in the data model. Another point is to visualize the hazard zone in the mobile application so that the user can see the size of the zone. Additional information is necessary for emergency applications. A further point is that the current application could be integrated into an existing tsunami warning application. These warning applications offer most detailed information about a tsunami, tsunami zone as well as a triggered tsunami.

References

CHOWDHURY, G. H., CHOWDHURY, M. & KUSHCHU, I. (2005), Prospects of Using m-Technologies for Disaster Information Management in Bangladesh and other LDCs. EURO mGOV 2005, United Kindom, 234-253.

CIVIL DEFENCE AND EMERGENCY MANAGEMENT. (2012, 03 23), Retrieved 04 23, 2012, from Christchurch City Council:
http://www.ccc.govt.nz/homeliving/civildefence/informationondisastershazards/tsunami.aspx.

CORMEN, T. H., LEIERSON, C. E., RIVEST, R. & STEIN, C. (2009), Introduction to Algorithms. Massachusetts, The MIT Press, 588 pp.

DARIENZO, M. (2003), Tsunami Sign Placement Guidelines. Open File Report OFR 03-06. Retrieved 06 06, 2012, from Department of Geology and Mineral Industries, Oregon: http://www.oregongeology.com/sub/earthquakes/Coastal/OFR0306Signs.pdf.

EDPEARSON. (2007), Retrieved 06 06, 2012, from edpearson:
http://www.edparsons.com/2007/11/android-and-lbs-in-the-stack-at-last/.

FRIEDERICH, S. (2012), Retrieved 30 04, 2012, from The Daily World: http://thedailyworld.com/sections/news/local/tsunami-warning-info-now-available-online-and-smart-phones.html.

GARTNER, INC. (2010), Retrieved 03 22, 2012, from Statista: http://de.statista.com/statistik/daten/studie/12865/umfrage/prognose-zum-absatz-von-smartphones-weltweit/.

GARTNER, INC. (2011), Retrieved 03 22, 2012, from Gartner: http://www.gartner.com/it/page.jsp?id=1622614.

GOOGLE INC. (2012), Retrieved 04 04, 2012, from Android Developers Website: http://developer.android.com/index.html.

KLINE, K. E., HUNT, B. & KLINE, D. (2009), SQL in a Nutshell. USA, O'Reilly Media Inc., 193 p.

MAHLMANN, P. & SCHINDELHAUER, C. (2007), Peer-to-Peer Netzwerke. Heidelberg, Springer.

MAPKI (2012, 04 12), Retrieved 04 23, 2012, from Google Map Parameters: http://mapki.com/wiki/Google_Map_Parameters.

ORACLE (2012), Retrieved 06 01, 2012, from: http://www.oracle.com/technetwork/database/options/spatial/index.html.

POSTGIS (2012), Retrieved 05 14, 2012, from PostGIS: http://postgis.refractions.net/.

PUNTAMBEKAR, A. A. (2008), Analysis and Design of Algorithms. Pune, Technical Publications Pune.

RAPER, J. (2012), Design constraints an operational LBS. In: GARTNER, G., CARTWRIGHT, W. & PETERSON, M. P. (Eds.), Location Based TeleCartography and Services, Berlin/Heidelberg, Springer, 13-23.

SCHEER, S., GARDI, A., GUILLANDE, R., EFTICHIDIS, G., VARELA, V. & DE VANNSAY, B. (2011), Retrieved 12 15, 2011, from Joint Reseach Center: http://publications.jrc.ec.europa.eu/repository/bitstream/111111111/15978/1/lbna24707enc.pdf.

SCHILLER, J. & VOISARD, A. (2004), Location-Based Services. San Francisco, Elsevier, 198 p.

TSOU, M.-H. & SUN, C.-H. (2006), Mobile GIServices Applied to Disaster Management. In: DRUMMOND, J. (Ed.), Dynamic and Mobile GIS: Investigating Changes in Space and Time. Boca Raton, CRC press, 213-224.

U.S. CENSUS BUREAU. (2012), Retrieved 05 29, 2012, from Topological Integrated Geographic Encoding and Referencing system: http://www.census.gov/geo/www/tiger/.

WILMERSDORF, E. (2007), Providing an Information Infrastructure for Map Based LBS – The Approach of the City of Vienna. In: GARTNER, G., CARTWRIGHT, W. & PETERSON, M. P. (Eds.), Location Based TeleCartography and Services, Berlin/Heidelberg, Springer, 479-492.