MANAGING MULTIPLE POINT-BASED DATA INSTANCES ON A SINGLE COORDINATE IN MAPPING MASHUPS

Brian Tomaszewski
The Center for Multidisciplinary Studies
Rochester Institute of Technology
31 Lomb Memorial Drive
Rochester, NY 14623
bmtski@rit.edu

Abstract

The development of so-called "mapping mashups" has seen tremendous growth in recent years. The basic idea behind a mapping mashup is that data sources with a geographic component are dynamically combined with a web-based mapping client to create a new, unique application. An example of a mapping mashup would be combining location-based data such as real estate transactions with the Google Maps client. Data sources used in mapping mashups are often dynamic. When working with dynamic point-based data in particular, it is difficult to predict how many data instances may be referenced to a single latitude/longitude coordinate when a map is rendered. Managing the display of multiple data instances on a single coordinate is a classic cartographic design problem. Managing access to multiple data instances on a single point becomes an even more challenging problem when a map is dynamically created and has interactivity that effects map display such as panning and zooming - capabilities which are often found in a mapping mashups. This paper presents a mapping mashup development case study with a specific focus on how the problem of managing the display and access of multiple point-based data instances on a single coordinate was addressed. In particular, the mapping mashup presented in this paper was developed to present humanitarian job vacancy information for the United Nations Office for the Coordination of Humanitarian Affairs (OCHA) ReliefWeb site (http://www.reliefweb.int). Humanitarian job vacancies are often highly frequent at locations where humanitarian crises are occurring such as cities like Kabul, Afghanistan or regions like the Darfur region of the Sudan. Thus, the problem of multiple point-based data instances on a single coordinate is common to the application domain discussed in this paper. From a technological perspective, geographic and other data utilized by the mapping mashup is created by geo-coding text extracted from a job vacancy Real Simple Syndication (RSS) feed and then combined with a Google Maps client. Methods used in the mashup to address the data management and access problems outlined previously include (a) multi-scale symbol display management using the Google Maps marker manager API, and (b) custom JavaScript procedures for managing access to and display of multiple point-based data instances on a single coordinate such as dynamic circular symbol placement and hierarchical data arrays. Methods used in this work to address the problem of multiple data instances on a single coordinate have resulted in a production mapping mashup application that can be accessed on the ReliefWeb site at http://www.reliefweb.int/rw/hlp.nsf/db900ByKey/vac_map?OpenDocument. Preliminary feedback of the application indicate it is a useful visual device for accessing humanitarian job vacancy information and is an effective compliment to text-based job vacancy information on the ReliefWeb site, The significance of techniques presented in this paper are two-fold. First, this work presents technical guidelines that other cartographers engaged in building interactive, web

mapping applications can utilize and expand upon to address similar display and interaction problems. Second, the application can contribute to research in the visual analytics domain that focuses on the synthesis and presentation of visual information derived from text-based data sources.

1. Introduction

A "mapping mashup" refers to an application where data sources with a geographic component are combined with a web-based mapping client to create an original, new application. An example of a mapping mashup would be combining location-based data such as earthquake instances with the Google Maps client (see Google Maps Mania (2009) for numerous examples of mapping mashups that have been developed for a wide variety of application domains).

The map created in a mapping mashup application is typically constructed using data sources whose contents constantly change. For example, the HealthMap application displays news stories from around the world related to infectious disease (Freifeld et al., 2008). As new, relevant stories are published, items displayed on the main map of the HealthMap application change to reflect new content incorporated into the map. The dynamic nature of map content used in mapping mashup applications creates design and technical challenges for the cartographer developing the mapping mashup. One particular challenge is the age-old cartographic design problem of displaying multiple data instances on a single latitude/longitude coordinate. This challenge is exacerbated when the map offers interactivity such as panning and zooming that is common in any modern digital mapping technology such as a mapping mashup.

This paper discusses a mapping mashup development case study of how the issue of managing multiple point-based data instances on a single coordinate was addressed. The structure of the paper is as follows. First, the mapping mashup's application context is discussed to establish the end-user requirements the application was developed to meet, and in turn, used to shape the technical solution used for the application. Next, the application's overall architecture is outlined as a full discussion is beyond the scope of this paper. The cartographic design and technical solution used to address the issue of managing multiple point-based data instances on a single coordinate is then discussed. The paper concludes with a summary and ideas for future work.

2. Application Context and Background

The organizations and people that operate at international scales and across international political boundaries to alleviate and mitigate human suffering can loosely be described as the "humanitarian community". Members of the humanitarian community include well known international organizations such as the United Nations (UN), non-governmental organizations (NGOs) like the Red Cross, humanitarian-related departments funded by the governments of countries such as the Humanitarian Information Unit (HIU) in the United States, and many more. At any given time, numerous employment opportunities diverse as the humanitarian community itself exist within the humanitarian community. These jobs range in geographic location and operational scope from positions such as a logistics manager needed to coordinate relief supply deliveries in the Darfur region of the Sudan to a high-level policy analyst in Washington, D.C.,

USA Many, if not all, of these jobs are essential to the continued effective operation of the organization needing to fill a position, and hence the ability of the organization to provide vital services to the people they serve.

2.1   The ReliefWeb Job Vacancies Service

The United Nations Office for the Coordination of Humanitarian Affairs (OCHA) ReliefWeb organization is the premiere central hub for collecting, maintaining, and disseminating information to the humanitarian community. In particular, ReliefWeb maintains a humanitarian job vacancy service. According to Wolz and Nam-ho Park (2006:6), *""Job Vacancies" is the most popular service on ReliefWeb, receiving 35% of all ReliefWeb site traffic and a large share of all email subscribers."* Furthermore, in 2007, 12, 937 vacancy postings were published on ReliefWeb, an increase of 30% from 2006 (ReliefWeb, 2007). With the job vacancy service being (a) the most popular service of the entire ReliefWeb website and (b) the general motivation to develop new, interactive and visual services for their end users, the ReliefWeb Map Centre team developed an interactive, map-based application that could expand the text-only viewing capabilities of the job vacancy service interface (ReliefWeb, 2009).

The map-based interface to job vacancy information was developed to meet the specific goal of fostering connections between humanitarian job seekers and employers through geographically visual (aka geovisual) representations of job vacancy information. The specific application developed to meet this objective is discussed in the next section.

3.   The Application

The application uses a server/client architecture. Major components of the application include (a) a vacancy Real Simple Syndication (RSS) feed derived from a vacancy database which serves as the primary data source for application, (b) server-side components which process the vacancy RSS feed, and (c) the web-map client that presents the server-side processed results to the end user. Figure 1 gives a visual overview of how each of these components fit into the overall application architecture.
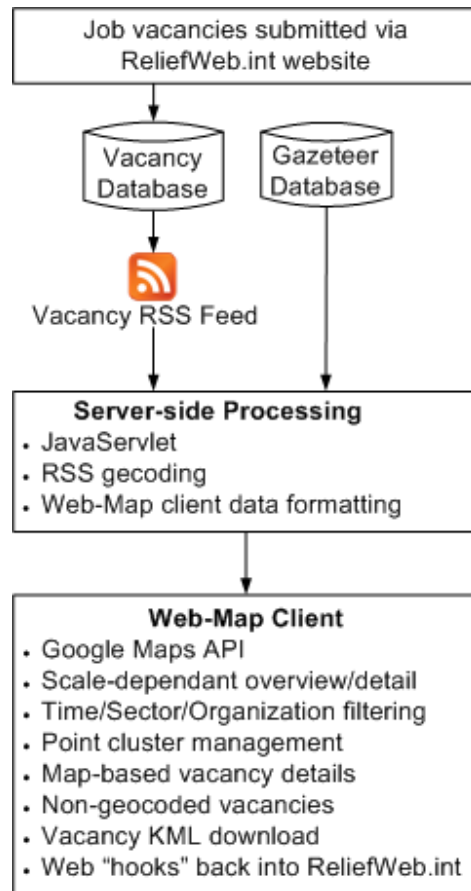
Figure 1: Application Overview

As discussed in section 1, full discussion of all aspects of the application, as outlined in Figure 1, are beyond the scope of this paper. In the following sections, details on how the cartographic design and interaction challenge of managing multi-point instances on single latitude/longitude coordinate is discussed.

3.1 The Cartographic Challenge

The cartographic challenge faced in developing the application stemmed from the fact that humanitarian job vacancies are often highly frequent at locations where humanitarian crises are occurring such as cities like Kabul, Afghanistan or regions like the Darfur region of the Sudan. Furthermore, application users needed to be able to examine vacancies on the map from temporal and thematic perspectives relevant to the humanitarian community such as the relationship between vacancies and various humanitarian reform sectors and organizations (Office for the Coordination of Humanitarian Affairs (OCHA), 2006). For example, a person looking for a job with a UN organization in the agriculture sector in Africa might want to know specifically about jobs fitting these criteria that are closing within three days so as to not miss a potential opportunity. Thus, the map design and interactions needed to support filtering functionality to assist end-users in their job searching needs. The following section discuses the technical solution developed to address these challenges.

3.2   The Solution

3.2.1   Overview and General Features

The application's web-client uses the JavaScript version of the Google Maps API (GM-API) (Google, 2009). The JavaScript version of GM-API was used as the application technical specifications explicitly stated that web-browser plug-ins, such as the Flash plug-in, were not allowed. The reason for this is that the overall ReliefWeb.int website does not support plug-ins as plugs-ins can limit accessibility to ReliefWeb.int. Thus, the application had to be fully functional in standard web browsers such as Internet Explorer using only HTML and JavaScript.

The map supports two scale-dependant, overview/detail viewing levels. Clusters of data points on a single geographic coordinate and displayed at the detail level are managed using JavaScript arrays. Functionality is also provided that allows map users to filter map display contents. These items are discussed in the following sections.

3.2.2   Overview and Detail Display Management with the Google Maps Marker Manager

Overview and detail displays are managed in the application using the Google Maps Marker Manger API (GM-MM API) (Ricket and Majewski, 2007). The GM-MM API was used as it is a stable, open source code library specifically designed for developing multiple viewing levels in Google Maps-based mashups. The GM-MM API works by structuring data points within a multi-dimensional array. Top-level elements within the array include the zoom level range and places associated with the zoom level range. Sub-elements within the array include specific place-instance information. The following pseudo-code is an example of an array that could be used with the GM-MM API.

```
Array mapLayers =
//the first two top level array elements – these supply the overview
"zoom" – [0, 3] //1st element – defines drawing levels 0 to 3
"places" – //2nd element – sub-array of points drawn between levels 0 to 3
        "point name": "United States"
        "lat/long coordinate": [40, -97]

        "point name ": "Canada",
        "lat/long coordinate ": [58, -101]

//second two top level array elements – supplies the detail view
"zoom" –  [4, 6],  //1st element – defines drawing levels 4 to 6
"places"  – //2nd element – sub-array of points drawn between levels 4 to 6
        "point name": "Headquarters Vacancy",
        "lat/long coordinate": [37.423021, -122.083739]

        "point name": "New York Office Vacancy"
        "lat/long coordinate": [40.754606, -73.986794]
```

The array defines two top level elements that will draw between zoom levels 0 to 3, which can be used as overview levels in Google Maps applications. The "places" element in turn defines

two sub-arrays of data instances that will draw at these levels. Also defined are two top level elements that will draw between zoom levels 4 to 6, which can be used as details levels in Google Maps applications.

For the vacancy mapping application, overview and detail views, as managed and implemented using the GM-MM API, appear as shown in Figures 2 and 3.
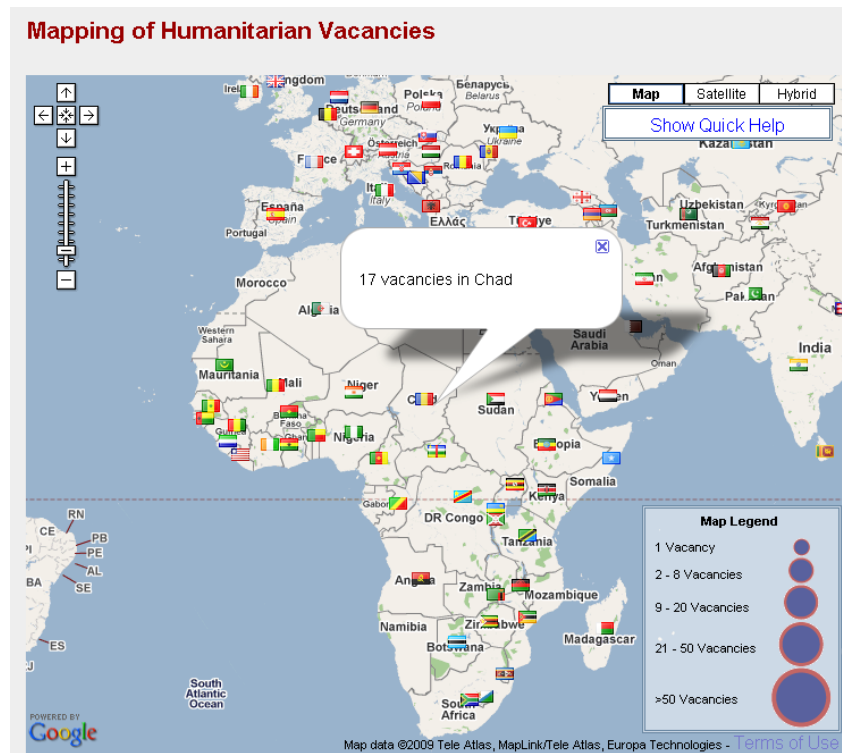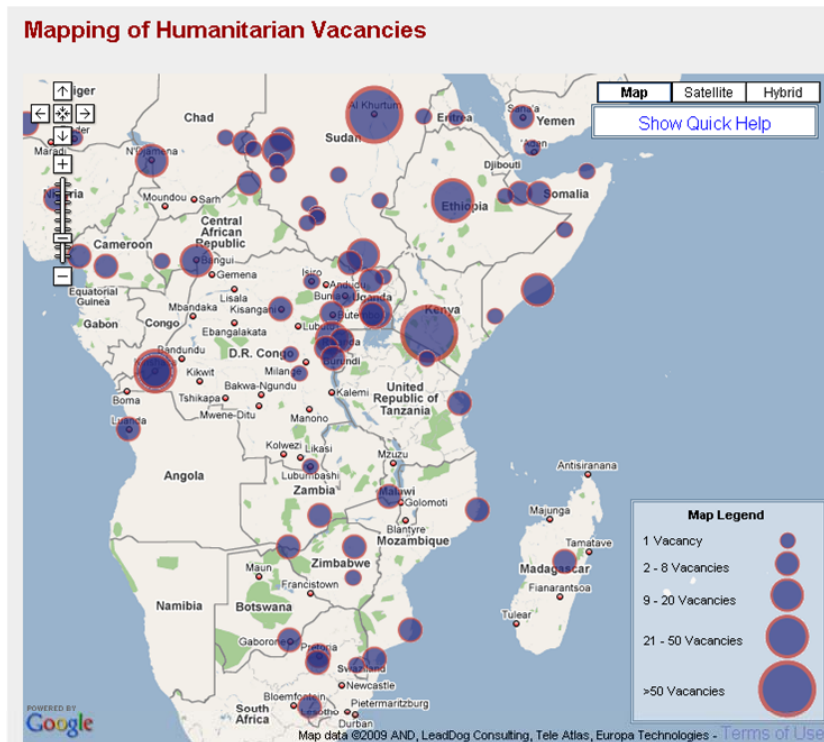

Figure 2: Application overview view.

Figure 3: Application detail view.

Figure 2 (the overview), displays between viewing levels 0 to 3. In this viewing level, flag icons are used to represent countries where job vacancies are located. As seen in Figure 2, users can mouse over a flag to see an information bubble indicating the number of vacancies contained in the country of interest. Upon zooming in further, the user will then see the detail view (Figure 3). In this viewing level, users can examine the map for specific job vacancy information. It is at the detail level where the point cluster management functions of the application are most critical.

3.2.3 Point Cluster Management

As seen in Figure 3, the detail view uses graduated point symbols for representing job vacancies. Point symbol sizes are based on a five-class, natural breaks data distribution. Points are assigned to a data class based on the number of job vacancies found at a given latitude/longitude coordinate. The clustering of multiple job vacancies on a single latitude/longitude coordinate was the key design and interaction challenge for developing the application. Managing the clustering of data on single coordinates is composed of two parts – a data processing component and a display/interaction component.

The data processing component starts when the vacancy application's webpage is loaded and the raw, geo-referenced vacancy data that was created by the server-side processing mentioned in section 3 in opened by the web client. Figure 4 shows what the raw, geo-referenced vacancy data looks like.

```xml
− <markers>
    <timestamp time="Mon, 27 Jul 2009 10:57:15 EDT"/>
    <no_vacancy_count value="66"/>
  − <marker lat="5.3411111" long="-4.0280556" country="CI" expire_soon="0" sectors="Coordination and Support Services, Food, Health" orgs="Non-governmental Organization" icon="single.png"
    ccentlat="8" ccentlong="-5" cname="C?te d'Ivoire">
      <p><b>Vacancy title:</b> <a href="http://www.reliefweb.int/rw/res.nsf/db900SID/OCHA-7UCEEE?OpenDocument&RSS20=07-P" target="_blank"> Uu expert en nutrition</a></p><p>
      <b>Source:</b> Action Contre la Faim</p><p class="BasicInfo"><b>Vacancy Posted On:</b> Mon, 27 Jul 2009 13:51:42 GMT<br>Vacancy is approximately 0 days old<br><b>Vacancy Closing
      Date:</b> 27 Sep 2009<br>Vacancy will close in 62 days<br><b>Sector(s): </b>Coordination and Support Services, Food, Health<br><b>Org(s): </b>Non-governmental Organization<br>
      <b>Location:</b> C?te d'Ivoire (Abidjan)<i><br>Geocoded Location: Abidjan</i><br><span class="CloseDate"> This information was auto-generated on: Mon, 27 Jul 2009 10:57:15 EDT</span>
      </p></div class="line"></div><p><img src="http://www.ungiwg.org/RWVacancyGeoView/img/rw_transparent.png"> information for C?te d'Ivoire<br><a href="http://www.reliefweb.int/rw/rwb.nsf
      /doc106?OpenForm&rc=1&cc=civ" target="_blank">Latest Updates</a><br><a href="http://www.reliefweb.int/rw/dbc.nsf/doc117?OpenForm&rc=1&cc=civ" target="_blank">Updates By
      Sector</a><br><a href="http://www.reliefweb.int/rw/rwb.nsf/doc404?OpenForm&rc=1&cc=civ" target="_blank">Maps</a>
    </marker>
  − <marker lat="-11.6666667" long="27.4666667" country="CD" expire_soon="0" sectors="None" orgs="Non-governmental Organization" icon="single.png" ccentlat="0" ccentlong="25"
    cname="Democratic Republic of the Congo (the)">
      <p><b>Vacancy title:</b> <a href="http://www.reliefweb.int/rw/res.nsf/db900SID/OCHA-7UCEES?OpenDocument&RSS20=07-P" target="_blank"> Resident Senior Program Manager</a>
      </p><p><b>Source:</b> National Democratic Institute for International Affairs</p><p class="BasicInfo"><b>Vacancy Posted On:</b> Mon, 27 Jul 2009 13:47:55 GMT<br>Vacancy is
      approximately 0 days old<br><b>Vacancy Closing Date:</b> 25 Sep 2009<br>Vacancy will close in 60 days<br><b>Sector(s): </b>None<br><b>Org(s): </b>Non-governmental Organization<br>
      <b>Location:</b> Democratic Republic of the Congo (the) (Katanga, Lubumbashi)<i><br>Geocoded Location: Lubumbashi</i><br><span class="CloseDate"> This information was auto-generated
      on: Mon, 27 Jul 2009 10:57:15 EDT</span></p><div class="line"></div><p><img src="http://www.ungiwg.org/RWVacancyGeoView/img/rw_transparent.png"> information for Democratic Republic
      of the Congo (the)<br><a href="http://www.reliefweb.int/rw/rwb.nsf/doc106?OpenForm&rc=1&cc=cod" target="_blank">Latest Updates</a><br><a href="http://www.reliefweb.int/rw/dbc.nsf
      /doc117?OpenForm&rc=1&cc=cod" target="_blank">Updates By Sector</a><br><a href="http://www.reliefweb.int/rw/rwb.nsf/doc404?OpenForm&rc=1&cc=cod" target="_blank">Maps</a>
    </marker>
```

Figure 4: Excerpt of raw geo-referenced vacancy data created by server side-processing and read when the application web page loads. The figure shows data for two job vacancies, each of which are in a <marker> tag.

The raw geo-referenced vacancy data is processed by the applications JavaScript web client in the following manner. First, the data file is opened as an XML stream. After this, every <marker> tag contained in the file is read to extract various pieces of data about a vacancy. In particular, each marker tag is inspected for (a) country information about the vacancy (for use in the overview display), and (b) information about the specific vacancy such as its latitude/longitude coordinate, title, and related sectors and organizations. Next, country and vacancy information is aggregated in order to populate JavaScript arrays used for display and interaction functions. Aggregating in this context means that countries and vacancies are aggregated by their respective latitude/longitude coordinates to prevent duplicates points from being generated in displays and to generate summary counts (as was seen for countries in Figure X). Vacancy information is also sorted by vacancy closing date to ensure that vacancies with closing dates within three days of when the data was processed on the server appear at the top of overflow map display boxes (which are discussed below). Following the aggregation and sorting of vacancies and their respective countries, map marker objects are then created for countries and vacancies respectively. Map markers are JavaScript objects that are instantiated on the map as graphic symbols. Map markers also contain attribute information about the object being mapped. For example, country map markers contain the name of the country. Vacancy map marker objects store information such as the closing date of the vacancy, the sector associated with the vacancy and organization type associated with the vacancy. This information is queried when the map filtering tools (discussed in section 3.2.4) are used. Furthermore, vacancies are processed to determine whether a latitude/longitude coordinate contains one or more vacancies and thus should use either a default Google Maps bubble or an overflow box for displaying vacancy information.

After processing the raw geo-referenced vacancy data, two arrays consisting of countries and vacancies are then added as sub-elements to a map layer array that is used with the GM-MM API for controlling the zoom-level specific views that were discussed in section 3.2.2. A third array containing all the map marker objects is also created for use with the filtering tools that are discussed in section 3.2.4.

The generation of the map marker and layer arrays is the end point of the data processing component for managing the clustering of data on single latitude/longitude coordinate and the beginning of the display/interaction component.

The display/interaction component with specific job vacancies is based on a simple rule that if a latitude/longitude coordinate has more than one vacancy associated with it, an overflow box is used to display information associated with the coordinate. Otherwise, a default Google Maps bubble is used. Figures 5a and 5b visually demonstrate this distinction.
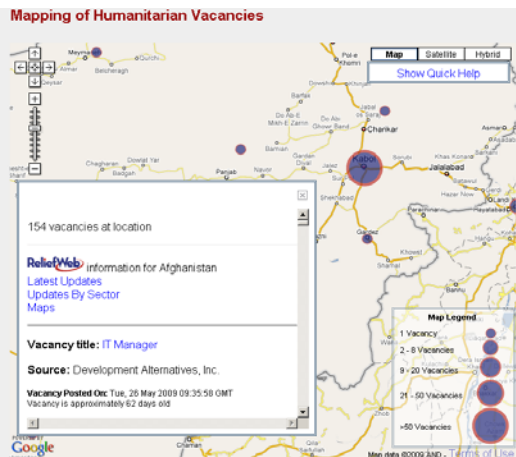
Figure 5a: Overflow box example for a coordinate with numerous vacancies. In this screen shot, Kabul, Afghanistan has been selected. Note how 154 vacancies are associated with this location, as shown in the top of the overflow box.
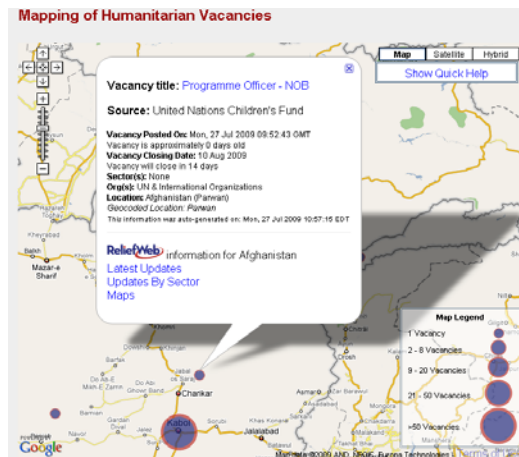
Figure 5b: Default Google Maps bubble example for a coordinate with one vacancy. In this screen shot, Parwan, Afghanistan has been selected.

As can be seen in Figures 5a and 5b, individual vacancies contain various detail information about the vacancy such as the vacancy title, the source of the opportunity, when the vacancy was posted on ReliefWeb, when the vacancy closes, what sector the vacancy is related to, organizations the vacancy is related to, and where the vacancy is located. Information that was automatically generated by the applications server-side processing is also included such as the location that was geocoded and when the information contained on the map was generated. Furthermore, the server-side processing also creates "hooks" back to the main ReliefWeb site by automatically constructing links to various forms information about the country where the vacancy is located (as seen in the bottom of the bubble shown in Figure 5b). The variety of variables that any given vacancy has such as closing date, sector, and organization, coupled with the fact that over 1000 vacancies are present on the map at any time required that filtering mechanisms be included so that users could find relevant information quickly and easily.

3.2.4 Filtering Mechanisms

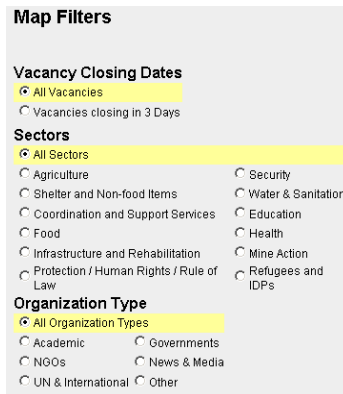Users can filter the map contents using the interface shown in Figure 6.

Figure 6: Detail of the Map Filter interface. This interface is located below and to the left of the main map.

Users can select one option button within each of the three filter categories (Closing Dates, Sectors, or Organization type), thus filtering data points within a filter category using the equivalent of OR in a WHERE clause of an SQL statement. For example, a user can select vacancies from the Agriculture sector OR the Security but not both. Figure 7a and 7b visually demonstrate the filter tools.
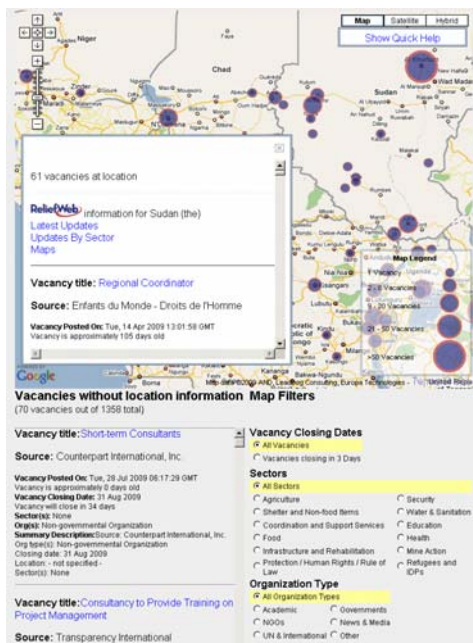

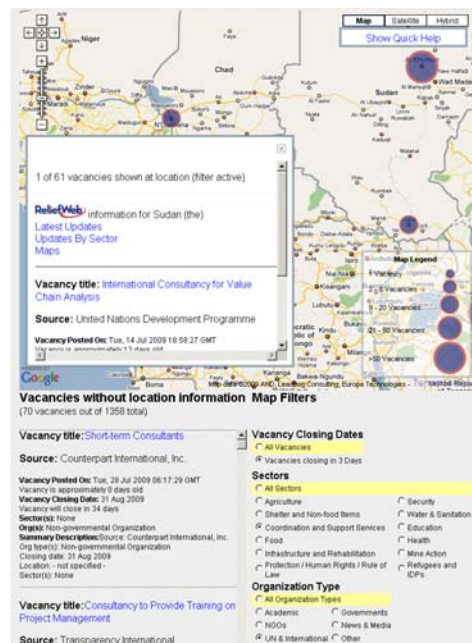**Figure 7a:** The map with no filters active. **Figure 7b:** The map with three filters active.

In Figure 7a, the map has no filters active and vacancies located at Khartoum, Sudan have been selected and are shown in the overflow box. Note in Figure 7a how there are sixty one vacancies at this location as indicated in the top overflow box and that 10-12 vacancy points can be seen along the western border of Sudan and various other spots in Africa. In Figure 7b, three filters –

vacancies closing in 3 days, vacancies from the Coordination and Support Services sector, and vacancies from UN & International organizations have been selected. Note that with these filters active, the number of points shown on the map in Figure 7b has been greatly reduced due to numerous vacancies not meeting the filter criteria and that the overflow box is now indicating that only one vacancy is at the location due to filters being active. In its current implementation, the application does not change map symbol sizes when locations with more than one vacancy have been filtered.

In technical terms, the filtering mechanism works as follows. When a user selects a filter option from the filter interface, all of the map markers in the map marker array are inspected to see if their attributes match the selected filter(s). If a map marker does match the selected the selected filter(s), then the map marker remains visible on the map, otherwise it is made invisible. As markers are inspected for matching selected filters(s), items such the summary counts contained in the overflow boxes are updated to reflect the effects of the filter, as was seen in Figure 7b.

4. Summary, Conclusions and Future Work

The application represents an important advancement in the use of geographic visualization approaches for humanitarian information management. In particular, by visually presenting job vacancies on a map, the application can help create advocacy for vacancies that might not normally receive much attention or awareness through standard, text-based representations. Maps in general are becoming more widely recognized as advocacy devices for humanitarian causes (O'Connor, 2008, Tomaszewski and Czárán, 2009) Thus, the application can be situated as a specific example of the use of mapping for humanitarian advocacy. Furthermore, the visual and interactive aspects of the application align the application with calls from the humanitarian information practitioner community for the increased use of visualization approaches to represent complex data and information (Office for the Coordination of Humanitarian Affairs (OCHA), 2008). The application also contributes to research on transforming and synthesizing text derived from dynamic web-based data sources to geographically visual formats such as maps (Wick and Becker, 2007).

The technical solution developed to meet end-user requirements can easily be repurposed for future applications. For example, the ReliefWeb site contains thousands of place-based information artifacts such as news stories, published maps, and country profiles. Such information could be searched and accessed using map-based interfaces like the one presented in this paper. The display and interaction functionality presented in this paper for managing multiple data instances on a single latitude/longitude coordinate, ideally, could allow map users to find relevant information more quickly and easier, thus ultimately leading to improved evidence-based decision making for humanitarian operations.

5. Acknowledgements

6. References

Freifeld, C., Mandl, K., Reis, B. and Brownstein, J. (2008) 'HealthMap: Global Infectious Disease Monitoring through Automated Classification and Visualization of Internet Media Reports', **Journal of the American Medical Informatics Association**, 15, pp. 150-157.

Google (2009) 'Google Maps API', <http://code.google.com/apis/maps/>, Last Accessed: 29 July 2009.

Google Maps Mania (2009) 'Google Maps Mania Blog', <http://googlemapsmania.blogspot.com/>, Last Accessed: 8 July 2009.

O'Connor, S. (2008) **Maps for advocacy: An Introduction to Geographical Mapping Techniques Maps**, Tactical Technology Collective.

Office for the Coordination of Humanitarian Affairs (OCHA) (2006) 'United Nations Office for the Coordination of Humanitarian Affairs
UPDATE ON HUMANITARIAN REFORM' in **OCHA Reports**.

--- (2008) 'Global Symposium +5 on Information for Humanitarian Action: Final Report' in **OCHA Reports**.

ReliefWeb (2007) '2007 ANNUAL REPORT: Statistics', <http://www.reliefweb.int/rw/hlp.nsf/db900ByKey/Stats2007?OpenDocument>, Last Accessed: 20 May 2008.

--- (2009) 'Vacancies', <http://www.reliefweb.int/rw/res.nsf/doc212?OpenForm>, Last Accessed: 29 July 2009.

Ricket, D. and Majewski, B. (2007) 'Marker Manager', <http://googlemapsapi.blogspot.com/2006/11/marker-manager.html>, Last Accessed: 28 July 2009.

Tomaszewski, B. and Czárán, L. (2009) 'Geographically Visualizing Consolidated Appeal Process (CAP) Information', **Proceedings of the 6th International ISCRAM Conference,** pp. 1-6.

Wick, M. and Becker, T. (2007) 'Enhancing RSS Feeds with Extracted Geospatial Information for Further Processing and Visualization' In **The Geospatial Web-How Geobrowsers, Social Software and the Web**, ed. by Scharl, A. and Tochtermann, K., pp. 105-115. Springer, London.

Wolz, C. and Nam-ho Park (2006), 'Evaluation of ReliefWeb', Office for the Coordination of Humanitarian Affairs, U. N., Forum One Communications.