Interactive Visual Exploration of a Large Spatio-Temporal Dataset: Reflections on a Geovisualization Mashup

Jo Wood, *Member, IEEE,* Jason Dykes, Aidan Slingsby, and Keith Clarke

Abstract — Exploratory visual analysis is useful for the preliminary investigation of large structured, multifaceted spatio-temporal datasets. This process requires the selection and aggregation of records by time, space and attribute, the ability to transform data and the flexibility to apply appropriate visual encodings and interactions. We propose an approach inspired by geographical 'mashups' in which freely-available functionality and data are loosely but flexibly combined using de facto exchange standards. Our case study combines MySQL, PHP and the LandSerf GIS to allow Google Earth to be used for visual synthesis and interaction with encodings described in KML. This approach is applied to the exploration of a log of 1.42 million requests made of a mobile directory service. Novel combinations of interaction and visual encoding are developed including spatial 'tag clouds', 'tag maps', 'data dials' and multi-scale density surfaces. Four aspects of the approach are informally evaluated: the visual encodings employed, their success in the visual exploration of the dataset, the specific tools used and the 'mashup' approach. Preliminary findings will be beneficial to others considering using mashups for visualization. The specific techniques developed may be more widely applied to offer insights into the structure of multifarious spatio-temporal data of the type explored here.

Index Terms — Large dataset visualization, text and document visualization, multiresolution visualization, geographic visualization, applications of infovis.

1 Introduction

Extracting structure, meaning and insight from large, multifaceted, spatio-temporal datasets is a challenging task and such data are consequently underused in many application domains [1]. This paper considers how the increasingly used 'mashup' approach to software and data integration can be applied to exploratory visualisation in order to address this underuse. In doing so, wider issues for the use of mashups in visualisation are identified.

Exploratory visual analysis that proceeds in an iterative fashion can be a highly effective means of preliminary investigation [2,3]. Such analyses require the means to select, inspect and modify visual encodings, with data transformed, filtered, sampled and aggregated according to time-, space- and attribute-based criteria. As the exploration proceeds, flexibility is needed to generate new samples, derivatives (such as alternate spatial aggregations), filterings and visual encodings in response to the insights and informal hypothesis gained and to integrate these with ancillary data. This process can be likened to Shneiderman's [4] "visual information-seeking mantra: overview first, zoom and filter, then details-on-demand".

Off-the-shelf geographic information systems (GIS) provide some of the functionality, particularly for spatial processing, but less flexibility for geovisualization. The alternative of developing tools from scratch with the required flexibility is difficult, time-consuming and requires skills not possessed by many engaged in geovisualization. Development effort may hinder progress and be too slow to allow iteration to be of sufficient benefit [5]. For this reason, toolkits have been developed that allow different interactive visualization components to be linked together using scripting or programming languages; e.g. GeoVISTA *Studio* [6], InfoVis Toolkit [7], and Improvise [8]. These frameworks can be used to build powerful visualization applications from specialist components.

- *Jo Wood (jwo@soi.city.ac.uk), Jason Dykes (jad7@soi.city.ac.uk), and Aidan Slingsby (a.slingsby@city.ac.uk) are based at the giCentre, Department of Information Science, City University, London.*
- *Keith Clarke (kclarke@geog.ucsb.edu) is based at Department of Geography, UC Santa Barbara.*

Manuscript received 31 March 2007; accepted 1 August 2007; posted online 27 October 2007. Published 14 September 2007.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

We propose an alternative approach for the exploration of data that has considerable potential. It is inspired by the geographical mashup, which takes a set of open and freely-available resources, and combines them using *de facto* standards often based on XML. The use of general purpose scripting offers flexibility in interactive visual encoding specification and the filtering and processing of data according to spatial, temporal and attribute criteria, potential for rapid prototyping, and opportunities for distribution so that others can be invited to explore data and integrate findings with other data and software.

We demonstrate this approach with a geovisualization mashup case study in which we use Google Earth [9] for interactive visual synthesis of encodings generated using a combination of MySQL [10] (for data storage and querying to select and aggregate), PHP (for linking the database and the server and generating output) and LandSerf (for surface processing, calculating spatial derivatives and output). KML [11] is used to describe visual encodings and define interactions. This combination allows us to store, query, process and produce abstract graphics. Google Earth is used as a means of interactively visually analysing and synthesising data, through its spatial and temporal navigation tools, its access to wider contextual data, its ability to stream data from servers in response to user interaction and its embedded HTML browser.

We use this combination of tools to visually explore a mobile directory service log file containing 1.42 million records with spatial, temporal and attribute components. We demonstrate the flexibility of the approach by designing a set of interactive abstract graphics: 'tag clouds', 'tag maps', 'data dials' and 'geo-mipmaps' which visually synthesise data and provide interactive means of filtering and aggregating by space, scale, time and attributes. These are used in conjunction with other ancillary data to explore the dataset.

The approach is informally evaluated at several levels. Firstly, we reflect on the success of the specific interactive visualizations which both synthesise data at multiple spatial, temporal and attribute aggregations, and act as bases for filtering of the data. Secondly, we focus on characteristics of our data that have been revealed and warrant further investigation. Thirdly, we contemplate the specific tools and languages we have used in the mashup. Finally, we consider the geographical mashup approach and discuss the value of its use for the initial stages of the exploratory process as well as the information visualization development process as a whole. We argue

that geographic space provides a powerful and intuitive shared framework in which to synthesise data for exploration.

2 Approach — Mashup technique

The approach follows the conventions of the web application hybrid or mashup [12] that has gained recent popularity. While the term mashup is loosely defined, it is considered here to involve the integration of widely available applications using web-based technologies [13] to create a new application tailored to a specific task. In particular, it exploits the increased use of XML to mark up data in a way that allows it to be reused in a variety of contexts [12]. Mashups commonly use some form of geographical representation to integrate applications and data sources and provide a visual interface to them [14]. Google Map technologies are described by [15] as variously accessible, agile, adaptable and data rich. They are seen as "quick, popular and probably – soon – ubiquitous" and thus being particularly 'mashable'. The re-use of existing functionality and data to create new applications tailored to specific tasks has the potential to meet the needs of those engaging in exploratory visual analysis.

Technologies used in mashups tend to be freely available with published APIs following *de facto* or *de jure* standards. They include technologies for specifying semantic information and associating styles (e.g. RSS, KML, HTML, CSS), server-side technologies for retrieving information and generating content dynamically (e.g. servlets, PHP, ASP, JSP) and client-side technologies for enabling user interaction (e.g. JavaScript, applets, browsers and geobrowser applications). Ajax ("Asynchronous JavaScript and XML") [16], which combines web based protocols with client-side JavaScript for interaction is used in many mashup applications. While we do not use Ajax directly, many of the design principles behind Ajax are utilised. Asynchronous communication between client and server is important so that user interaction is not interrupted by requests for and the delivery of data over a network. This is supported by the use of thicker clients and efficient data caching in order to reduce traffic between client and server. Equally important to these technologies is the use of standard tools and conventions for interaction that are widely available [16].

Sophisticated and customisable spatial processing functionality is required in a geovisualization mashup to transform data and perform spatial and statistical comparisons. An adaptable environment in which to investigate and develop new techniques for visual representation and interaction is also essential. This flexibility usually requires relatively low level programming (e.g. using languages such as C++ or Java) and considerable development time [17]. A potentially conflicting requirement is that development must be sufficiently rapid to allow prototyping and new techniques to be created *as part of* the visualization process.

We use the mashup approach to address this conflict by focusing effort not on the (demanding) development of visualization tools and GIS functionality, but rather on the specification of visual encoding and interaction using high level KML markup and developing loose couplings between KML and a GIS. This has the potential to reduce development time by allowing existing pre-written applications to implement the representation and interaction (primarily Google Earth in our case study) as well as specialised spatial processing (LandSerf in our case study). Using mashup technologies for geovisualization in this way has the added benefit of enabling collaboration and extension by other users of these and similar applications.

3 Case Study — Technologies and data used

The component technologies were configured to explore our dataset (Fig. 1). All are freely-available with open APIs and data exchange formats and each has capabilities that fulfil some of the requirements outlined above and enable us to address priorities identified through communication with the data owners. It should be noted that while the following technologies are all required to build an integrated system, most of the development surrounded the generation of KML and visual exploration using Google Earth.

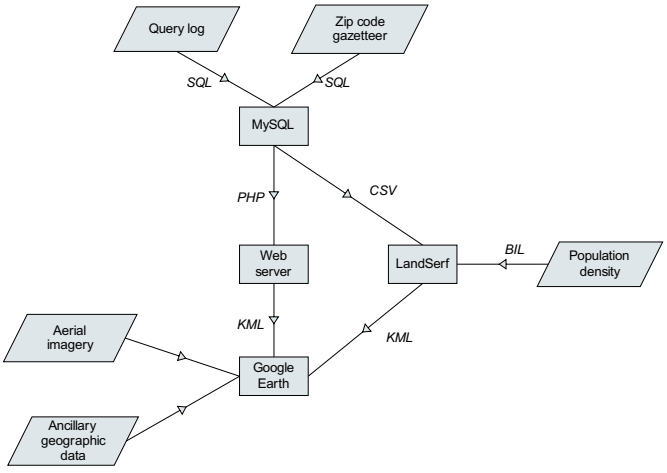

Fig. 1. Data, applications and communication technologies used.

3.1 Google Earth

Google Earth [9] (we used v4.0, free edition) is a one of a growing number of geobrowsers, widely used for the visual synthesis of spatial data and for interacting with these data at multiple scales. It provides an intuitive interface, including tools for zooming, panning and tilting, and it has good data generalisation capabilities, employing adaptive level of detail mapping, automatic text placement and text culling techniques for reducing clutter. Data are presented as layers, which may contain hierarchical sets of elements that can be turned off individually or by set. Ancillary data including high resolution aerial photography, gazetteers and boundaries are available in Google Earth. User-defined and third party data are specified in KML. Data can also be streamed from a server in response to changes in the visible area of the viewing window sent by Google Earth. Version 4.0 of the software also has timeline functionality allowing elements with temporal information to be encoded (KML v2.1) and then selected and filtered by the user. The timeline implementation in Google Earth is designed to support linear timelines with various temporal resolutions ranging from seconds to years. Cyclic timelines (for example by hour or by month) can be supported by arbitrarily defining dates. Ordinal data can also be mapped in a similar way through rescaling.

3.2 KML

Data are supplied to Google Earth in KML (we used v2.1), an XML markup language in which graphical encodings and interactions can be defined for interpretation [11]. Like all XML-based languages, it is based on a nested set of elements. A 'Document' or 'Folder' element is uppermost, which contains features and styles for cartography. Within these, various elements may be added, many of which can be associated with coordinate-based geometries and actions that generate further KML in response to interactions. These include: 'Placemark' (an element with a geometrical description), 'Overlay' (an area overlaid on the ground or the screen), 'Region' (geographic space that can be used to trigger events when entered), 'Container' (used to nest features hierarchically), and 'NetworkLink' (uses HTTP to stream content, either at regular intervals or in response to user action). Style elements allow icons, text and images to be selected and specified. Features can be labelled, be associated with descriptive text (which can include hyperlinks to other KML or HTML resources) and be associated with 'TimeSpan' elements for specifying the time period for which they are valid. Placemark elements contain one or more geometries, including points, lines, polygons and 3D models. Together these features provide considerable flexibility in the specification of interactive, abstract geographic graphics.

3.3 MySQL

MySQL (version v5.0.27-log is used here) [10] is a relational database management system for data storage, indexing and retrieval according to spatial, temporal and attribute-based criteria, using SQL. MySQL was selected as it represents a commonly used, free and widely available platform for the storage and retrieval of large datasets and offers good integration with web services. We use standard SQL so that the possibility of using an alternative relational dataset remains open. Nevertheless, we recognise that by employing spatial extensions (e.g. [18]) or a dedicated OpenGIS Web Feature Service ([19]) we could optimise the efficiency of the spatial query of the database while maintaining a degree of interoperability with other packages.

3.4 PHP

PHP (version v4.3.6 used here) [20] is a server-side scripting language for web delivery. It is used primarily in this context as a mechanism for connecting with and querying the MySQL database from HTTP requests made using the KML NetworkLink and for subsequently serving KML output. In common with the other technologies evaluated here, it was chosen as it represents an open widely available platform with a published API.

3.5 LandSerf

LandSerf, (v.2.3 used here) [21] is a Geographic Information System for processing and visualizing spatial data. It represents the most specialised of the software platforms evaluated here. It contains spatial analytical functionality that is not currently available in any of the other software platforms, in particular, the generation of scale-specific surface measurements. It has a published API and can output both vector and surface data in KML format. It was chosen as part of this evaluation due to a need to generate scale-dependent surfaces as part of the visualization process but also to investigate how specialised software can be integrated in a mashup approach.

3.6 Data and context

The dataset explored here consists of 1.42 million requests made of a US-based mobile telephone service go2 [22] over a period of one month. Each record represents a search request made on a mobile device for a service selected from a fixed list. The result of the user's query would be a location-sensitive set of available businesses or services, possibly with directions from the user's location at the time of query. Each record in the database consists of the location at which the query was made, the time and date of the query and a further 16 attributes, including the service or business name ('tokenstring'), a zip code, 'languageID' and 'userID'.

The dataset was provided to explore opportunities for visualizing the data for internal analysis and external presentation by 'slicing and dicing' in a number of ways. Suggestions from the provider included the possibility of structuring by carrier, time, time of day and category and investigating the spatial distributions of these characteristics. Geographic relationships between call destination and query result location were considered as key to explaining user behaviour; for example, by investigating how users' locations correspond to the locations returned by the search, how this relates to what users are searching for and whether requests and needs vary over space. Such knowledge is valuable for explaining user behaviour and needs and can be converted into valuable user interface options and functionality that assist in the acquisition of relevant information. We were invited to segment the dataset and develop visual techniques to begin to investigate these issues.

Initial exploration of the dataset involved establishing and verifying the meaning of individual fields and selecting aspects of the data on which to focus in relation to the context provided by go2. Concentrating on space, time, language and business query provided a rich multi-dimensional space to explore that related closely to the interests of the data provider and allowed us to assess the technologies used in our mashup case study.

4 Design — Visual Encoding, Interaction and Synthesis

A range of example visual encodings and interactions are developed using these technologies to demonstrate the possibilities of the geovisualization mashup and the flexibility of the technologies used in our case study. The use of these techniques, the linkages between them and their combination with ancillary datasets are demonstrated in a Google Earth visual synthesis. We use our experience of this particular combination of technology, technique and dataset to informally evaluate the potential of the geovisualization mashup.

4.1 Placemarks and visual issues

Visual encodings can be generated in KML with some flexibility. Styles that specify the colour and transparency of the geometries and icons associated with Placemarks can be chosen according to any aspect of the data. For example, Placemarks were coloured according to LanguageID in order to explore their spatial and thematic associations. Two key issues are worth considering in detail.

4.1.1 Colour and symbology

High-resolution satellite imagery and aerial photography are the basis of the Google Earth geobrowser. This frequently obscures the detail of graphics described in KML and loaded as part of a mashup. One solution is to add partially opaque images to remove or reduce the visual impact of the base data using the GroundOverlay element. An complementary approach is to use bright saturated colours and bold symbols (large point symbols and thick line symbols) to visually emphasize KML graphics over the textured background.

Some of the well-established guidelines for the use of colour and symbology in cartographic representations (e.g. [23]) may be less valid in geographic 'mashups' that use high resolution colour imagery as a geographic base than is the case in other forms of cartography and visualization.

4.1.2 Overplotting, crowding and generalisation

Google Earth deals with crowded symbols and text by automatically reducing visual clutter in real-time as spatial data are browsed. This is achieved in a number of ways. Spatially coincident points are collapsed into a single symbol, which explodes to reveal the entire cluster of points when clicked. Labels around Placemarks are positioned or culled so as to not obscure each other.

These methods are employed automatically and with no indication to the user. For example, the only visual manifestation that a Placemark might be one of a set of spatially coincident Placemarks, is that there may be a couple of extra labels where space allows (Fig. 2, right). It is only by the user clicking on the symbol that it becomes clear that it is a collapsed composite of spatially coincident Placemarks, revealed as an explosion (Fig. 2, left). This automatic functionality for dealing with complexity in data illustrates Google Earth's suitability for visual synthesis. However, as illustrated, the lack of any visual feedback regarding this should be kept in mind when using Google Earth for visualization.

Spatial dithering and changes in symbology (e.g. colour, opacity, line thickness and size) can be used to reflect the existence of unseen or coincident data. These techniques are not currently automatically employed by Google Earth, but are both achievable through position adjustment or style specification of affected Placemarks at the KML generation stage (Fig. 3).

Text culling, label explosion and dithering all reduce the consistency of mapping between geographical location and graphic. Displacement, simplification and enhancement, which separate cartographic placement from geographic location, are commonly used in cartography. In a dynamic environment the effects and extent of overplotting, crowding and generalisation can be explored.

4.2 Plotting locations with labels and data synthesis

Subject to these visual issues, plotting the data as point Placemarks, with the 'tokenstring' as the label and the other data as a description provides an indication of the spatial distribution of cell phone

queries. The attribute values of particular records can be inspected and compared with ancillary datasets provided by Google Earth.

Early impressions of the locations of all queries, generated by mapping types of query as point Placemarks in KML, suggested a close spatial association with the main centres of population and indeed the population distribution of the United States. Whilst Google Earth provides a range of useful ancillary data one of the strengths of the approach used here is that exogenous data can be integrated into the mashup. We added US population density data from the *Gridded Population of the World GPW v3,* dataset [24]. This dataset contains population counts for 1995 at a 2.5 arc-second resolution - approximating population density at a 5km resolution. Data were masked by US shoreline and state boundaries in LandSerf so they could be compared directly with the go2 records.

Fig. 2. Using the Google Earth aerial photography for context: These queries appear to have been made from a rest area, and may relate to a single session and information need. *Photographic imagery copyright 2007 TerraMetrics Inc.*

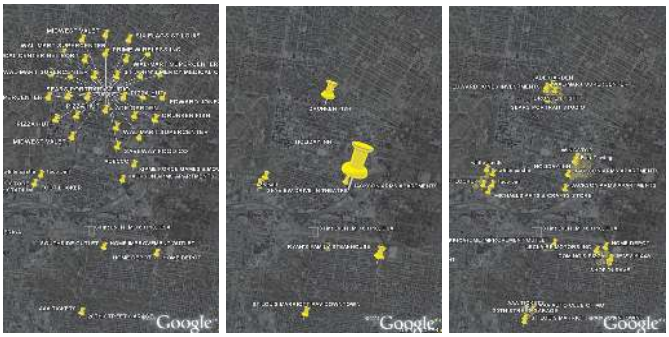

Fig. 3. Three plotting techniques: overplotting with user initiated 'explosion' (left), proportionally sized (middle) and dithering (right), using identical data and extents. The latter two techniques must have the positional and size adjustments explicitly specified in KML. *Photographic imagery copyright 2007 Sanborn.*

Many points in the labelled query data were overplotted. Visualizing and inspecting the timestamps of individual records drew attention to cases of multiple queries being made milliseconds apart, with the same userID and query. This context suggested that certain groups of records should be treated as a single event representing a single user 'session' [25].

Overplotting with different userIDs and different queries was identified at a wide variety of locations. This suggested two possibilities. One is that queries were made from a fixed position where multiple queries would be expected, such as a building of multiple occupancy, parking lot or rest area. An indication of this could be seen by inspecting the aerial photography, subject to positioning error (Fig. 2). The other possibility was that not all the positioning was obtained through GPS or triangulation of cell mast signals, and that positions were simply assigned to the location of the nearest cell mast (which, incidentally do not show up well on aerial photography unless shadows are visible).

The zip code field in the go2 dataset formed the basis of our geographic comparison of the location of a user performing a query and the location of the results returned. Our information was that this represented the location of queried businesses. In order to perform this comparison, an additional dataset was incorporated – a freely available zip code gazetteer [26]. This was imported into the RDBMS to convert zip code into latitude and longitude coordinates. The relationship between the origin of the query and the apparent destination zip code was explored initially using source-destination vectors generated in KML, using dithering to address the spatial coincidence problem.

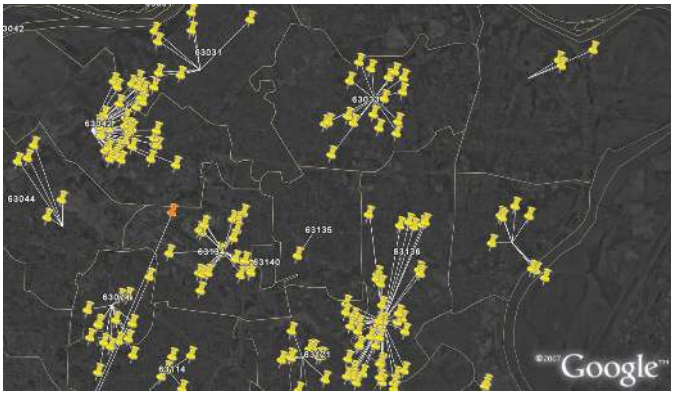

Fig. 4. Query- zip code vectors: The majority of lines are within a single zip code, but some of the zip codes in the zip code gazetteer appear to be wrongly positioned (see the orange Placemark). *Photographic imagery copyright 2007 Europa Technologies*.

This representation was visually compared with the zip code boundaries available in a Google Earth layer. The vectors only rarely crossed the zip code boundaries (Fig. 4). Where this did occur, lines were frequently sufficiently long to suggest a zip code georeferencing problem. This was confirmed by comparing these zip code anomalies with other independently derived zip code gazetteer services. The combination of our abstract graphics with ancillary data caused us to re-evaluate the properties of the destination data. Visual comparison of the source-destination vectors with the zip code boundaries suggested the destination zip codes were, contrary to expectations, simply the zip codes of the query location. The visual checking and synthesis with secondary data in the mashup enabled us to identify this important spatial relationship. In this case, this resulted in insights concerning the characteristics of the data and prevented continued unproductive analysis of one of the key relationships that we had hoped to explore.

4.3 Encoding Interaction

KML can be used to describe and activate interactions by using NetworkLinks to call PHP scripts that generate new content. The tree-like presentation of the KML hierarchy in Google Earth lends itself well to hierarchical filtering. For example, if Placemarks are grouped into sets by 'tokenstring' (business name), within which they are organised by 'languageID', it is possible to display the languageID for particular queries. Rapid prototyping allowed the data to be sliced in a variety of ways before placing them in these hierarchical sets. Sets at any level in the hierarchy can then have their visibility toggled in the geobrowser.

Whilst this approach has powerful exploratory potential, there are also some inherent limitations. The rigid hierarchical structure constrains the exploratory potential of data that do not lend themselves to hierarchical organisation. For example, it is impossible to show all tokenstrings for one languageID, because the latter is nested within the former. Alternative hierarchical structures can be

generated from the same dataset and loaded in Google Earth as layers that can be combined and compared. Alternatively, ordinal data can be mapped onto an ordered timeline as mentioned in section 3.1, allowing the user to interactively select content through Google Earth's timeline interface.

5 Flexibility and New Views

Labelled plots provided us with insights into the spatial structure of the dataset and enabled us to inspect individual records and make comparisons with ancillary data. The line plots and their combination with an ancillary dataset have informed us about the spatial relationships between locations associated with individual records.

The following visual encodings convey data at different spatial, temporal and attribute aggregations, and allow for data filtering in these terms. They were designed to focus on particular aspects of the data following our initial exploration and are indicative of the possibilities provided by the technologies used in our case study for developing new user-designed, data and context specific views.

5.1 Tag clouds

Tag clouds (Fig. 5, right) are a visualization technique for summarising the prominence of words. The size of each word in a collection signifies its frequency of use [27]. Words are usually ordered alphabetically. Tag clouds were conceived for the summarisation of 'tags', the free form text labels that are widely used for labelling digital content, such as photographs (e.g. Flickr), movie clips (e.g. YouTube) and Internet bookmarks (e.g. del.icio.us). They summarise patterns of use in the application of tags for labelling content – changes in use through time can be considered by constraining to particular time periods or by using more sophisticated temporal analyses (e.g. [28] and [29]). Tag clouds can be generated from any collection of text or piece of prose (e.g. tagcrowd.com) and are a widely used visualization technique that is applicable to the textual information in our dataset.

Interactive tag clouds were developed to summarise the relative frequency of queried businesses within a particular geographical area. Tag clouds generated as HTML using the NetworkLink functionality were displayed in Google Earth's integrated web browser and show the non-spatial word frequency within the geographical area selected for viewing in the geobrowser [30].

Clicking a word in the cloud results in the generation of KML that zooms in on the geographical extent of the viewable area to which the word applies. We can then generate a tag cloud thatis constrained to the new view, and so allow iterative interactive exploration of 'tag space' in Google Earth (Fig. 5).

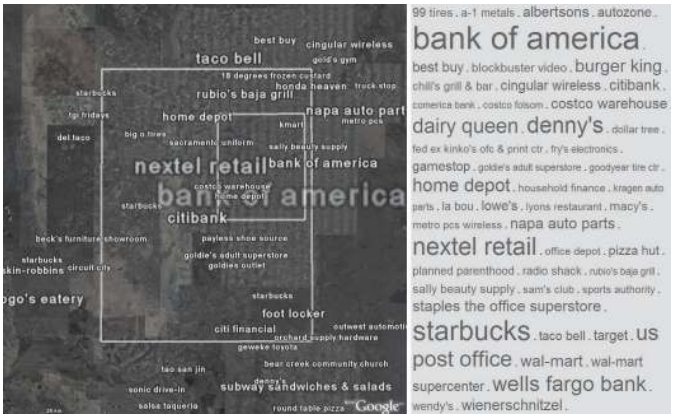

Fig. 5. Tag map (left) and tag cloud (right) of the 50 most popular business names in the selected geographic area. The smaller bounding box is 'bank of america'; the larger is 'starbucks' (which is more spatially diluted). *Aerial imagery copyright 2007 NASA.*

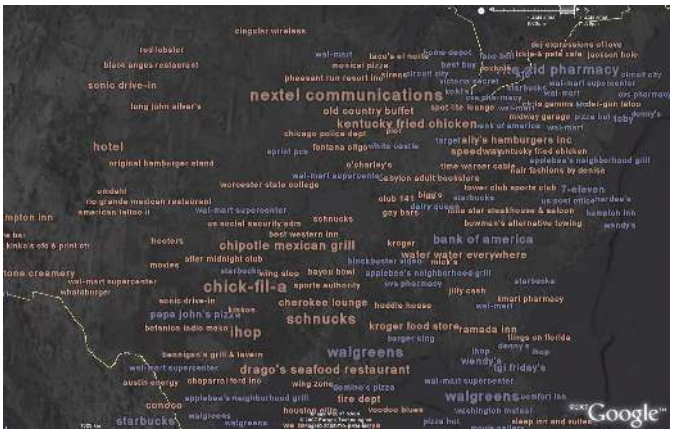

Fig. 6. The tag map of queries made between 20:00 and 0:00 is dominated by queries ffast food. Red indicates positive deviation from the expected norm; blue indicates negative deviation (mainly shops in this late-night example). *Aerial imagery copyright 2007 NASA, Europa Technologies and TerraMetrics Inc.*

Tag clouds summarise the frequency of textual attributes and allow us to aggregate records by textural attribute and constrain them to a geographical area. The interactivity that has been incorporated allows the geographical extents of specific words to be explored through filtering and selection by text and space. The interpretation of HTML in Google Earth's integrated browser results in a close coupling between these views of geographic and information spaces.

5.2 Tag maps

Tag maps [31,30] are spatial versions of tag clouds, in which words are placed on a map whose sizes correspond to the frequency at that position at a spatial scale appropriate for the spatial extent of the map (Fig. 5). Various arrangements and orders of tags in tag clouds have been reported; for example Hassan-Montero and Herrero-Solana [27] cluster tags according to semantic similarity and Kerr [32] surrounds each tag with those that share tag space, whose distance from the central tag reflects the level of association. Techniques that rely on the distance and placement of words often use spatial metaphors for conveying relations [33]. Here we use geographical space not as a metaphor but as a basis for a conventional mapping of georeferenced text into Cartesian coordinates. Tag maps can be considered geographically grounded tag clouds, where the spatial relationships between words correspond to real geography.

Tag maps spatially aggregate records at a scale appropriate to the viewable window selected by navigating with the geobrowser. As a new map is requested for a specific view, the tag map is dynamically generated from the database. In this way, the tag map offers a true multi-scale means to explore data at different spatial aggregations.

The combination of tag maps and tag clouds offers both a spatial and aspatial view of the frequency of words [30]. Tag clouds can draw our attention to high frequency words in a specific geographical area, and tag maps can show whether the word is localised (it would appear once in a large text size; e.g. 'bank of america' in Fig. 5) or evenly distributed (it would appear many times in small text; e.g. 'starbucks' in Fig. 5).

Additional symbolism can be used to convey information in tag maps. Colour can be specified in KML to encode information about word frequency and how this relates to global expectations. In Fig. 6 words are sized by frequency of occurrence but coloured using a binary scheme according to whether they are more or less frequent in an area than expected according to global (national) norms.

We have also used tag maps to explore differences in the occurrence of words at different times of the day, week and month through the Google Earth timeline functionality. KML can be generated to animate a tag map throughout the month, but this does not reveal any strong patterns. Daily and weekly temporal cycles are

more likely to reveal spatio-temporal trends and the timeline can be mapped to any temporal scale, using arbitrary dates. For the daily cycle shown in Fig. 6, we used a fixed date (1st January 2005) but varied the time component of this date to encode the time of day of all queries in the log. The timelines in Google Earth allow periods to be interactively selected and sequentially animated.

5.3 Data dials

MacEachren *et al*. [34] suggest that combinations of abstract and realistic symbolism may be useful in certain geovisualization applications. 'Data dials' group records by geographical location, time of day and attribute. They are multivariate abstract graphics designed to visually encode numeric and categorical information relating to the queries made at a particular location using features available in KML. Each corresponds to a geographical location and the queries made at particular times are conveyed as a line radiating from a point representing the location of the query. The angles of lines correspond to the time of the query and lengths to the number of queries made at any time. Colour can be used to encode other attributes such as language ID. A nested hierarchy can be used for interactive filtering by attribute in Google Earth (see section 4.3).

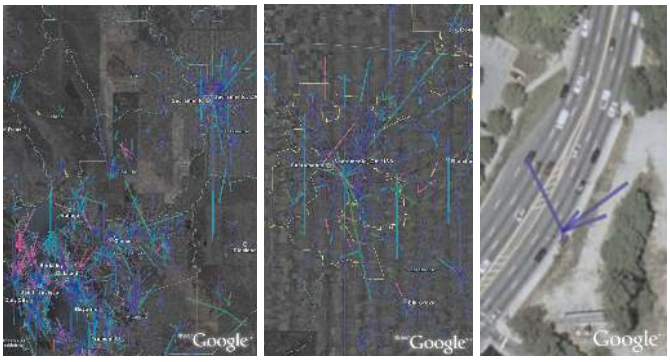

Fig. 7. Data dials at different zoom level showing the time of day (as 24 hour dial), number of queries (line length) and the languageIDs (color hue), sorted by day from the centre. The image on the right is the data dial for the apparent roadside queries in Fig. 2. In this case the queries were made at five different times of day leading us to reject the postulated 'single session' theory. *Photographic imagery copyright 2007 Europa Technologies and TerraMetrics Inc.*

Data dials are indicative of the kinds of abstract symbolism that can be generated using the mashup approach and demonstrate the interesting synergies that can result from combinations of contemporary high resolution data and abstract cartography. They can be streamed to Google Earth in real time and calculated for discrete points or as summaries of regions at resolutions appropriate to the data and current spatial extent. They can be scaled according to the extent of the same view. The graphics are suitable for data measured on a cyclical scale and can represent records aggregated at different temporal resolutions (e.g. by hour of day, day of week, day of month). Fig. 7 uses the metaphor of a 24 hour clock to show query numbers and locations by time of day and languageID.

6 CUSTOMISABLE SPATIAL PROCESSING FUNCTIONALITY

The primary purpose of the geobrowser is to aid the interactive display and integration of data rather than to perform spatial analysis. This provides a potentially significant barrier to the investigation of spatial patterns where the ability to perform analytical and statistical processing is often paramount. We integrated specialised software and data using the mashup approach by performing analysis using existing GIS software and transforming the results into KML for exploration within Google Earth.

The spatial pattern of any widespread human activity is likely to reflect in part the distribution of population in which that activity takes place. This was the case for the pattern of mobile phone queries that broadly matched the concentration of people in urban settlement. To investigate our approach further the population data [24] were compared with the go2 query data to draw attention to areas where queries of any given type were over or under represented in comparison to that expected given the local population. The LandSerf software [21] and its associated high level scripting environment LandScript [35] were used in this case study to process gridded raster population data.

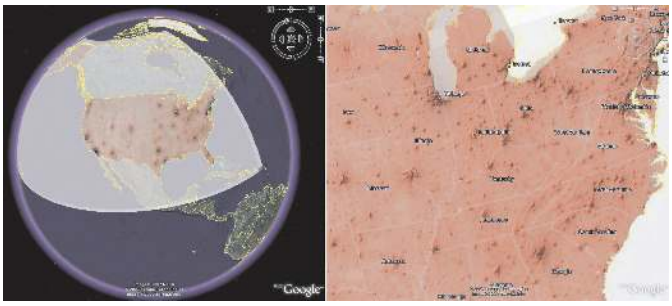

Fig. 8. Population density surface generated in LandSerf displayed in Google Earth using geo-mipmapping. The surfaces displayed at any viewpoint relate to resolutions appropriate for the distance of view. *Aerial imagery copyright 2007 NASA, Europa Technologies and TerraMetrics Inc.*

Generating and visualizing population density presents particular challenges due to the tendency towards a highly positively skewed frequency distribution – the vast majority of areas have very low density and few areas have very high density [36]. It is also a highly scale-dependent measurement as density at a given location varies greatly if calculated over different spatial extents.

Several techniques can be used to overcome the former problem including transformation to linear density [37,36] and smoothing using quadratic regression [36]. The scale dependency of population density is more problematic as it is this very characteristic that may be of interest during the visual exploration process. LandSerf was used to generate population density surfaces over a range of scales through quadratic smoothing. Each surface was associated with a unique *level of detail* (LOD) in KML so that when viewed in Google Earth densities calculated at increasingly finer resolutions are triggered as the user zooms in to any part of the region of interest (see Fig. 8). This is an example of geo-mipmapping [38] that uses spatial navigation to control the selection of scale-based data.

Chi expectation surfaces [39] were generated to compare query densities in cells with those expected given underlying populations.

$$\chi = \frac{obs - \exp}{\sqrt{\exp}} \quad (1)$$

Fig. 9. Chi-expectation surfaces. Darker red indicates greater numbers of business queries than expected according to population, darker blue indicates fewer than expected queries. *Aerial imagery copyright 2007 NASA, Europa Technologies and TerraMetrics Inc.*

where *obs* is the number of observed queries in any grid cell and *exp* is the number expected assuming it to be proportional to the population in that cell. The resulting values are normalised with respect to absolute numbers of observations and scaled around 0. This spatial processing was achieved by using LandScript to encode the transformation as a focal map algebra operation [40]:

```
# Script to calculate the chi expectation surface for mobile go2 queries.
# Expected values based on 2005 gridded population data.
 Version(1.0);
basedir = "c:\research\go2\data\";

# Convert query point values to density surface
queryPoints = open(basedir&"allBusiness.vec");
queryDensity = density(queryPoints,1,0.04166667,0.04166667);

population = open(basedir&"popDensity.srf");
popMean = info(population,"mean");

queryDensity = open(basedir&"allBusinessDensity.srf");
queryMean = info(queryDensity,"mean");

chi = new(queryDensity);
chi = ifelse(population >0,((queryDensity/queryMean)-(population/popMean))
                          / sqrt(population/popMean), 0);

#Save expectation surface as mipmapped KMZ file.
save(chi,basedir&"chiQuery.kmz");
```

The chi-expectation surfaces and population density surfaces were added to the mashup. The various encoding techniques and interactions described here all share the same geographic space and in combination provide a basis for comparison and ideation. By creating surfaces of the chi expectation value and using a divergent colour scheme, areas that with greater or fewer than expected numbers of business queries can be identified over a range of scales (Fig. 9). As might be expected, the dominant spatial distribution is of blue underrepresented use of the mobile query service. The service is used more than expected (indicated in red) in selected parts of urban settlement and these can be explored in the context of the mashup using the ancillary data and visual encodings described above.

7 Conclusions

Conclusions can be drawn from this work at several levels. We were able to *design* and combine specific encodings and interactions to address the interests of the data owners by interactively 'slicing and dicing' according to time, geography and attribute. These techniques may be more widely applicable.

Some understanding was achieved along with *insights* that warrant further attention. We have identified patterns of mobile business query in time and space through tag maps, tag clouds and data dials and will continue working on the issues raised.

In terms of the *technologies* used in this particular mashup, server side processing was fast and dealt with the data volumes efficiently - MySQL works very well with datasets of this size. KML described data, visual representation and interactions effectively. Google Earth is impressively optimized to stream and display data quickly, producing smooth graphics rapidly to support the thought process. It also provides a rich set of tools for data selection including collapsible tree diagrams, timelines, spatial panning and zooming. None of these tools required direct development as part of the mashup as they were provided by the Google Earth platform. Rather, development was concentrated on the appropriate markup of data in KML to forge a connection between the data under analysis and the interactive tools available to explore the data. This is an important lesson in the use of mashups, since it implies that despite the large number of technologies involved, rapid development can be carried out concentrating on a single data-focussed technology.

The visual integration of ancillary data available through Google Earth was extremely useful – particularly the aerial photography and zip code boundaries for checking the integrity of data. Other layers such as the photos and hypertext available through the Geographic Web may also be of use in providing valuable contextual information.

Specialized GI functionality was introduced into the application to transform and aggregate data, and to calculate and smooth density surfaces. The loose coupling of LandSerf using KML as the binding meant that additional data and alternative graphical approaches could be rapidly exploited as part of the iterative visualization process

Limitations included the hierarchical organisation of data encouraged by XML and the data selection interface through which Google Earth provides access to the data. Organising data according to space, time or attribute through multiple alternative hierarchies is a somewhat inelegant solution, but one that worked for us and accords with the mashup philosophy. Other data may not be so amenable to hierarchical organisation.

Time is not dealt with by Google Earth and KML as effectively as space because the temporal extent selected through the interactive timeline is not currently communicated to the server. Whilst the client-side filtering by time is useful functionality, achieved with impressive speeds for geovisualization, communicating the temporal extent through a NetworkLink would allow spatio-temporal as well as spatial aggregations and selections to be made server-side. The mashup approach may provide a solution – there is no reason why these or other technologies should not evolve to perform this function. But this highlights one of the distinctive characteristics of using mashups for geovisualization. Unlike lower-level programming approaches, solutions that meet visualisation objectives are dependent on APIs that have a degree of volatility over time. The very same rapid development cycles facilitated by the mashup, coupled with a wide user-developer base who share solutions over the web mean that there is no stable or established route to finding a particular visualisation solution. While there is a risk that specific mashup technologies may change, our experience suggests that changes in technologies over the period of development (particularly in the Google Earth platform and the KML specification) tended to be backward compatible and resulted in increased functionality and flexibility over time. This suggests that the mashup may well be better suited to the *exploratory* visualisation process where specific visualisation goals evolve over time.

Indeed, the mashup philosophy, of loosely coupled functionality and data that is integrated through XML, complements the kind of iterative development needed in exploratory work. In comparison with lower level programming, development of visual interaction was rapid. This was especially advantageous in the early iterations of visual exploration where we were able to explore data associations and develop new visual tools on a scale of daily iterations.

Markup gives us the power to integrate data from various sources allowing us to describe visual encodings and add styles of interaction to spatially integrate and visually explore diverse datasets with existing tools – some of which were widely used, others very specialist. Impressively, it allowed us to deal effectively with large spatio-temporal datasets in a number of ways: through symbolism, interaction and spatial processing – including techniques such as sampling, symbol culling, symbol explosion, data streaming, aggregation, transformation (into density surfaces in our example) and geo-mipmapping. The results were accessible to others (both physically and conceptually), and a history of KML queries is retained in the Google Earth 'My Places' folder and can be used to log activity and thus describe and share the exploratory process. These features were a considerable help to our team when participating in this work.

There is certainly considerable scope for geovisualization mashups. They deserve further consideration in visualization case studies that integrate other technologies and alternative data. Our use of HTML hyperlinks in tag clouds to trigger calls to the online database that generate new KML is one example of a number of possibilities that might be explored: SVG could be used to produce alternative maps and graphics that add data to the mashup according to user interactions and queries. Other geobrowsers and data may provide alternative possibilities and insights.

We conclude by emphasizing the analogy between mashing technologies to produce an application and mashing views to explore data. Both involve flexible synthesis: one of data and functionality, the other of visual encodings of data through interactions. This correspondence, our positive experience and the evolving technologies suggest a bright future for geovisualization mashups and scope for developing specific techniques to address particular data visualization challenges.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support of go2 Directory Systems (www.go2.com) in allowing us access to samples of their spatially referenced query data, the Leverhulme trust, which funded Clarke's sabbatical at City University and the helpful remarks and observations of reviewers and colleagues in the giCentre. Figures are screen shots taken from Google Earth (v4.0.2737, free version), incorporating its ancillary data and aerial imagery from NASA, Sanborn, TerraMetrics Inc., DigitalGlobe, New York GIS and Europa Technologies whose permission to reproduce their imagery is gratefully acknowledged.

REFERENCES

[1] D. Guo, J. Chen, A.M. MacEachren and K. Liao, "A Visualization System for Space-Time and Multivariate Patterns (VIS-STAMP)", IEEE transactions on visualization and computer graphics 12 (6), pp. 1461-1474, 2006.

[2] A.M. MacEachren, M. Wachowicz, R. Edsall, D. Haug "Constructing knowledge from multivariate spatiotemporal data: integrating geographical visualization with knowledge discovery in database methods", International Journal of Geographical Information Science 13 (4), pp. 311-334, 1999.

[3] M. Gahegan. Chapter 4: "Beyond Tools: Visual Support for the Entire Process of GIScience. Exploring Geovisualization", J. Dykes, A. M. MacEachren and M.-J. Kraak. Elsevier Ltd, 2005.

[4] B. Shneiderman. "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations", Visual languages, Boulder; CO, IEEE Computer Society Press, 1996.

[5] J.A. Dykes, "Exploring spatial data representation with dynamic graphics", Computers and Geosciences 23 (4), pp. 345-370, 1997.

[6] M. Takatsuka and M. Gahegan, "GeoVISTA Studio: a codeless visual programming environment for geoscientific data analysis and visualization", Computers and Geosciences 28(10). pp. 1131-1144, 2002.

[7] J.D. Fekete, "The InfoVis Toolkit. Information visualization"; Proceedings of the IEEE Symposium on Information Visualization 2004, Austin, TX, IEEE Computer Society, 2004.

[8] C. Weaver, "Building Highly Coordinated Visualizations in Improvise", Proceedings of the IEEE Symposium on Information Visualization 2004, Austin TX, 2004.

[9] Google Inc., Google Earth website, http://earth.google.com/, 2007

[10] MySQL AB, MySQL website, http://www.mysql.com/, 2007

[11] Google Inc., "KML 2.1 Reference", http://earth.google.com/kml/kml_tags_21.html, 2007.

[12] E. Wilde, "Knowledge Organization Mashups", Technical Report TIK Report No. 245, Computer Engineering and Networks Laboratory (TIK), ETH Zürich, http://dret.net/netdret/publications #wil06f, 2006.

[13] J.M. Maness, "Library 2.0 Theory: Web 2.0 and Its Implications for Libraries", Webology 3 (2), http://www.webology.ir/2006/v3n2/a25.html, 2006.

[14] R. Lerner, "At the forge: Creating mashups", Linux Journal 147, pp. 10, http://www.linuxjournal.com/article/8984, 2006.

[15] C.C. Miller, "A Beast in the Field: The Google Maps Mashup as GIS/2", Cartographic, 41 (3), pp. 187-199, 2006.

[16] J. Garrett, "Ajax: A new Approach to Web Applications, Adaptive Path Essay Archive", http://www.adaptivepath.com/publications /essays/archives/000385.php, 2005.

[17] J. Dykes, Chapter 13: "Facilitating Interaction for Geovisualization. Exploring Geovisualization". J. Dykes, A. M. MacEachren and M.-J. Kraak, Elsevier Ltd., 2005.

[18] A. Karlsson, "GIS and Spatial Extensions with MySQL", MySQL Developer Zone, http://dev.mysql.com/tech-resources/articles/4.1/gis-with-mysql.html, 2007.

[19] Open Geospatial Consortium, "OpenGIS Web Feature Service Implementation Specification", http://www.opengeospatial.org /standards/wfs, 2007.

[20] The PHP group, PHP website, http://www.php.net/, 2007.

[21] J. Wood, LandSerf 2.3, http://www.landserf.org, 2007.

[22] go2 Directory Systems, www.go2.com, 2007.

[23] C.A. Brewer, Chapter 7: "Color Use Guidelines for Mapping and Visualization", in A.M. MacEachren and D.R.F. Taylor (eds), Visualization in Modern Cartography, , Elsevier Science, Tarrytown, NY., pp. 123-147, 1994.

[24] Socioeconomic data and applications center (SEDAC), "Gridded population of the world", v.3, http://sedac.ciesin.columbia.edu/gpw/, 2005.

[25] A. Goker, H. Myrhaug, M. Yakici, R, Bierig, "A context-sensitive information system for mobile users", 27th Annual International ACM SIGIR Conference, Workshop on Information Retrieval in Context, Sheffield, UK, 2004.

[26] CFDynamics, Zip code database v.10, http://www.cfdynamics.com/ cfdynamics/zipbase/index.cfm, 2001.

[27] Y. Hassan-Montero and V. Herrero-Solana., "Improving Tag-Clouds as Visual Information Retrieval Interfaces", International Conference on Multidisciplinary Information Sciences and Technologies, InScit2006: Mérida, Spain, 2006.

[28] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan and A. Tomkins, Visualizing Tags over Time, WWW 2006: Edinburgh, 2006.

[29] S. Havre, E. Hetzler, P. Whitney, and L. Nowell, "ThemeRiver: Visualizing thematic changes in large document collections", IEEE Transactions on Visualization and Computer Graphics 8, pp. 9-20, 2002.

[30] A. Slingsby, J. Dykes, J. Wood and K. Clarke. "Interactive Tag Maps and Tag Clouds for the Multiscale Exploration of Large spatio-temporal Datasets", 11th International Conference on Information Visualisation. Zurich, Switzerland, July 2007, pp. 497-504.

[31] A. Jaffe, M. Naaman, T. Tassa and M. Davis, "Generating Summaries and Visualization for Large Collections of Geo-Referenced Photographs", MIR 2006 8th ACM SIGMM International Workshop on Multimedia Information Retrieval: Santa Barbara, CA, ACM, 2006.

[32] B. Kerr, "Sketches: worth a thousand words: TagOrbitals: a tag index visualization", ACM SIGGRAPH 2006 Sketches SIGGRAPH '06: Boston, Massachusetts, 2006.

[33] J.A. Wise, J.J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur and V. Crow, "Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Documents", in S.K. Card, J.D. Mackinlay and B. Shneiderman (eds), Readings in Information Visualization: Using Vision to Think: San Francisco, Morgan Kaufmann, pp. 442-45. 1999.

[34] A.M. MacEachren, M-J. Kraak and E. Verbree, "Cartographic issues in the design and application of geospatial virtual environments", 19th International Cartographic Conference, Ottawa, Canada, pp. 657-665, 1999.

[35] J. Wood, "LandScript – Controlling Landserf by Scripting", http://www.soi.city.ac.uk/~jwo//landserf/landserf230/doc/landscript

[36] J. Wood, P. Fisher, J. Dykes, D. Unwin, and K. Stynes, "The use of the landscape metaphor in understanding population data", Environment and Planning B: Planning and Design 26, pp. 281-295, 1999.

[37] P.J. Clark and F.C. "Distance to nearest neighbour as a measure of spatial relationships in populations", Ecology 35, pp. 445-453, 1954.

[38] J. Wood, "Multim im parvo - many things in a small place", in J. Dykes, A. MacEachren and M-J. Kraak (eds.) Exploring Geovisualization, London: Elsevier pp. 313-324. 2005.

[39] Census Research Unit, "People in Britain: a census Atlas". London: HMSO, 1980.

[40] C.D. Tomlin, "Geographic Information Systems and Cartographic Modelling" Englewood Cliffs, HJ: Prentice-Hall, 1990.