A User interaction case study on employing map technologies in Genes Reunited

Ali Habibzadeh - 2nd November 2009

The idea of developing a map application for our genealogy solution has been an idea which started to sound like a real plan at the beginning of 2008 when the marketing team in Genes Reunited started to evaluate and plan the budget which might be required for such project.

While the developers team started to analyse the ramifications of this project in terms of data translation from members address data fields into geographical coordinates and the back-end implementation of how this data can be refined and able to grow as our number of members grow.

Also data consistency and accuracy were important factors in the decision making of which platform should be chosen for this project.

The purpose of this article however is not to cover all the marketing or back-end development aspect of this project but to mainly focus on the user interaction and design process. But in brief the suggested platforms to be analysed for employment were Google Maps and Bing Maps (Virtual Earth at the time). These were proven dominant forces in terms of API facilities, Geographical data available and of course supplied documentation.

In terms of UK search results google maps proved to be using various data sets which produced more inconsistency comparing to Bing maps. Also google maps relies partially on Post Office data as one of it's sets which is not always up to date.

“Microsoft Mappoint Web Services ”, The commercially available geo-coding service from Microsoft proved to not produce ancient or postal data for the UK and the results were more consistent and accurate.

My involvement in this project has been mainly on the GUI side and client-side platform analysis. My thoughts were mainly focused on identifying user goals and creating strategies so the user can perform them as easily as possible without having to study the interface a lot.

We have various sets of Census data available which were already in use for several years by wide of users from experienced genealogist to only curious about family history first time users. These users searched these sets by providing either all or some of fields about the person they want to find like forename, surname, year of birth, its approximation and name of the place.

The results of the search were presented to the user using tabular presentation with sortable fields according to any of the ancestors details. Having this search mechanism already in place and learnt by our users I have decided not to present the map application as a separate route user can take to retrieve the results geographically, but in order to increase flexibility introduce it as an alternative method for the user to view their tabular results. As it would be much easier for them to think of the new feature just as an alternative representation of their tabular data.

Also for increased availability a search criteria modification tool was introduced both for the tabular and map presentation which contained a list of different census years and the search fields so the user can at any time modify their search details without having to return to the parent search landing page.

The interface of the map itself consist of three main sections.

1. Controls and navigation tools
2. Map area
3. Positioned data

The navigation tools were introduced based on what is already in the real world. These would convey the idea of moving towards different coordinates; a compass. The arrows on the compass respectively represent up for north, down for south and left and right for east and west. And the centre of the compass upon click would restore the initial view of the map when the user first made their search.

Various methods of movement were analysed and taken into account. We compared different acceleration and movement equations either more linear moves or swing likes. They were also tested for different durations and distant which travel towards when a particular coordinate control was clicked.

The optimum method of movement proved to be the one used from an equation introduced in 2008 by George McGinley Smith.

CHANGE IN VALUE * (CURRENT TIME /= DURATION) * CURRENT TIME + BEGINNING VALUE

Where change in value for our case was 100px and duration was 500miliseconds and beginning value was respectively the top and left position of the map tiles.

The zoom level used for the initial rendering was the sixth level of zoom Microsoft API offers in which the equator resolution is 2445.98 meters/pixels, which would allow to show the United Kingdom and Ireland in the first view.

The widely know + and – signs were used to control the zoom levels as well as implementation of the mouse scroll wheel events which makes the user be able to zoom on a location without having to move the mouse cursor away from the location there are on and move it to one of the zoom controls.

This initially created a dilemma for me as this implementation was against the default behaviour of the browsers. The mouse-wheel is used for scrolling the page. But the contextual analysis during the user sessions showed most users were more comfortable with being able to use the wheels for zoom level control and only a few thought of scrolling the page when they were on the map. Specially that I sized the whole application in a way that most of it would fit into most of today's monitor sizes.

So after this passing an event parameter in the scroll function and using these jquery methods which respectively prevents the browser from executing the default action and prevents any parent handlers from being notified of the event which consequently if placed inside the zoom function and is called when the user cursor is on the map and using the scroll wheel which prevents the browser from updating the page scroll value at this time.

```
event.preventDefault();
```

```
event.stopPropagation();
```

I also provided a map usage guide which the user can look at in case the arrow representations and + and – wouldn't immediately make sense.

Loading time was also a factor which required a form a visual cue to provide feedback for the user when their mapped data is actually available for them to browse which the simplest way was to show a decreased opacity overlay with a loading image while the ajax call is being made and fade the overlay out upon the success of the function.

At this point we had a semi-working prototype. The Data was Geocoded and the controls on the map were working so we introduced a series of pins to plot the relations found from the database on their geocoded lat-long. The pins were initially two transparent PNG's both consisting on a person silhouette places on a arrow like plate which it's sharp point was pointing to a location on the map. In addition one has a more pink and bright action colour and the other was a lower opacity version to differentiate the visited location from unvisited ones and interaction with each pin would reveal a balloon DIV which holds the relation data of the pin.

Two issues were clear at this point which was going to be a challenge to get fully right. One was what sort of even handlers should be employed for interaction between the pins and the data they hold and the high concentration of pins found in a certain area when results matches were plenty and caused huge overlapping which almost made some of the pins hidden from the user.

There was wide range of choices for approaching the even handling in terms of what action would reveal the balloon of each pin, how many can be opened at once, how each balloon will be opened in terms of movement and how they would be closed.

The proposed goal here is that the user should be able to reveal numerous geo positioned relation data as fast and easily as possible which would result in higher clicks on the company's selling point of relation data transcriptions and original census record images.

I found considerably faster user response to mouse-over and mouse out handlers. For the mouse-over of the pin to reveal the data balloon and for the mouse-out of the data balloon to hide the balloon again. However this introduced a lack of precision in two ways. One that this created the possibility of balloon being revealed unintentionally which decreased the attention of the user to the data they were seeing and second in the case of too closely positioned pins sometimes the user viewed the neighbouring data balloon while thinking they are looking at the data of the pin they thought they were hovering on.

However in the alternative implementation of this where the balloons were revealed upon the user clicking on the pin produced much more precise results but users were getting frustrated with the amount of clicks they had to do to browse all the data.

This made me wonder that the answer to the problem is not to alter the way event handing has proven to be more useful and easy to navigate for the user but the issue should be tackled by changing how the pins were distributed on the map.

The overlapping of the neighbouring pins could be greatly reduced if the user had to automatically zoom into the area where neighbouring pins where overlapping.

So the map clusters were introduced using the

`VEShapeLayerSetClusteringConfiguration(`*`type`*`, `*`options`*`)`

where type is a `VEClusteringType` Enumeration specifying which shapes to cluster and options is A `VEClusteringOptions` Class object specifying how the pushpin cluster is displayed.

This has a grid based clustering algorithm which divides the map into equally sized squares. If there is more than one VEShape pushpins in a those squares, they will be combined into one cluster. This allows to see more clusters in zoomed-out levels and they will break into pins which has initially made them as you zoom-in.

By default, push-pin clusters will have a special cluster push-pin icon which was overridden to match the rest of the icon sets I created for the application. Each cluster will also have a default pop-up balloon with information about the number of relations it was containing.

This allowed for the map results not to seem cluttered and confusing and it also solved the issue of mouse over events precision. A slight amount of randomization was used in order to convey a more natural placement for the clusters without compromising the exact placement of the pins they hold.

There were also cases where two people with the same name lived at the same address during different years which was presented as a list inside the data balloons.

All throughout the interface high contrast colours have been used between text and controls and their backgrounds. More green hues were given to informational elements and warmer hues like purple and pink were given to elements resembling actions which the user is expected to take.

Further information on the business needs and its impact on the subscription business model of Genes Reunited can be found on 2 page Case Study by Microsoft.

This application can be found on the records section of Genes Reunited website upon registration.