Daniel Henrie*

Ordnance Survey historic town plans of Scotland (1847-1895): Geo-referencing and web delivery with ArcIMS and OpenLayers

Summary:

This paper describes the digital 'curation' process and how it developed to make the Ordnance Survey Historic Town Plans of Scotland available on the National Library of Scotland's website. It focuses on the aims behind making these available digitally and studies the particular processes and presentation options implemented to improve delivery of and access to them. Beginning with a brief history of the Town Plans and aims behind their digitisation, it then goes on to examine the initial website presentation and related concerns. It then focuses on the reasons and aims for further development, the technology and processes used to bring about such improvements, and examines the issues and problems that arose. The stages looked at are: Geo-referencing the Town Plans, using ESRI's ArcIMS software to deliver the geo-referenced mapping in a customised interface, the limitations of the ArcIMS presentation and the exploration of OpenLayers open source software to create mash-ups. This final stage aims to outline the success achieved using OpenLayers to creating a website that is faster, more accessible and contemporary and benefits from using mash-ups to contextualise our historical mapping.

Introduction: Historical Background and Project Aims

Between 1847 and 1895, the Ordnance Survey mapped sixty-one Scottish towns at the most detailed scales in their entire history. The resulting maps are a comprehensive snapshot of urban Scotland and an incredibly rich information resource on the history and geography of all the larger populated places in Scotland. Initially, the scale chosen was 1:1,056, or five-feet to the mile, a scale ten times larger than the 6 inch to the mile mapping (1:10,560) which became the de facto standard for all rural areas. For the purposes of sanitary engineering, the five-foot scale was soon found to be too small, and a number of towns paid for surveys at the true ten-foot scale (1:528) at the instigation of the Board of Health. In 1855, following pressure from Ordnance Survey the Treasury approved the nation-wide mapping of towns at the metric ten-foot scale (1:500).[1]

The primary aim in digitising the OS Town Plans was to widen access. They are an incredibly popular resource, widely used in areas such as academic research, family history and surveying. Digitisation allows us to offer remote access that enables a very important collection of maps to be explored via our website.

A project such as the digitisation of the OS Town Plans is also an opportunity to create and store as much useful metadata alongside the digital maps as possible. Scanning the whole OS Town Plans series involved one thousand nine hundred and eighteen sheets to cover all editions of each town. A system to record information and digitally archive the sheets is a key element of such large-scale digitisation. This useful data forms good records and builds a strong base of information for searching and 'recalling' files as necessary in the future.

---

* Mapping Assistant, National Library of Scotland, Edinburgh [ml749dh@nls.uk]
[1] For more detailed further information please see http://www.nls.uk/maps/townplans/intro.html

Digitisation Process

The digitisation of the OS Town Plans followed the National Library of Scotland's standard process. A numbered database record was created for every sheet of the OS Town Plans series in our Digital Objects Database (DOD) and metadata was compiled for each record: title; sheet number; date and geographic location were all recorded. A unique image file number was assigned and used as the file name when saving the corresponding scan. The scans were made at a resolution of 400dpi and saved as full colour 24-bit tiff images (to enable background transparency). A directory path was added to each file's database record, making them easily located in our digital archives.
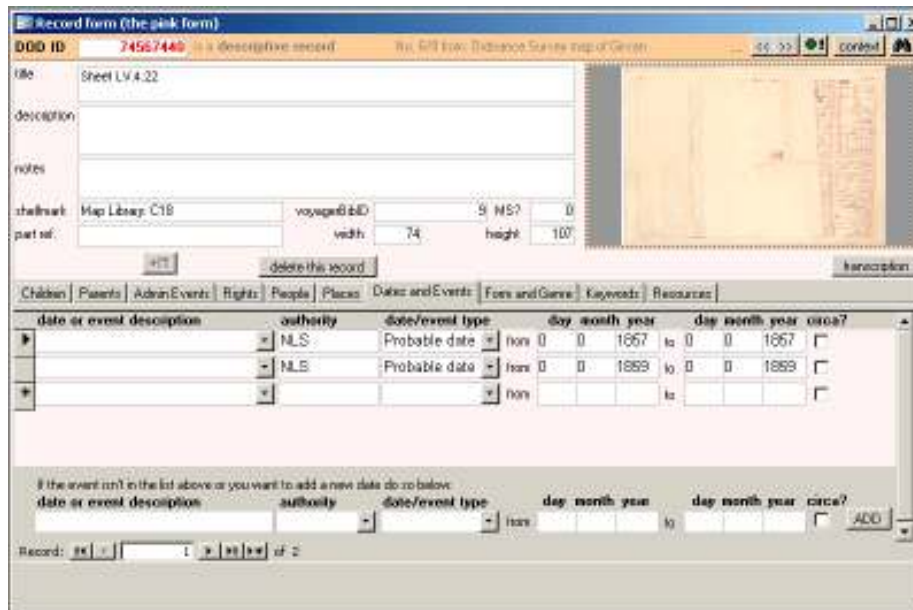

Figure 1 - National Library of Scotland's Digital Objects Database showing selected metadata for a scanned OS Town Plans sheet.

Initial Website Presentation

Demand for this series meant we wanted to make them available online as quickly as possible. A section existed on the NLS Map Library website (http://www.nls.uk/maps/towns/index.html) for other historic Scottish town plans. As this was already a visible and well-used section of the website, it was decided to integrate the OS Town Plans within this. This website section is a simple html page divided into sets of links for different Scottish towns. Each set of links, laid out chronologically, can be clicked to open a new html page for the specific map selected. A new html link for each set of Town Plans was added to every relevant town's section. Upon clicking each link, a new webpage opens displaying a graphic index for the OS Town Plans for that town. A graphic index provides an overview of the whole town, featuring sheetlines marking the location of each individual sheet. Simple image mapping was used to generate html to make these graphic indexes 'clickable': an alt tag and an html link were created for each sheet, and when opened, the corresponding OS Town Plan opens in a new window.

This initial website was a simple navigation solution. By exploiting an already established page to deliver new links, we created a clear, concise interface in which to browse individual OS Town Plans sheets. This readymade solution was perfect for making the maps quickly available, but just adding new links within an existent large series of links perhaps did not give the OS Town Plans adequate online prominence.

Figure 2 - Image showing original access to OS Town Plans. The image shows the set of links for Town Plans of Edinburgh with link to OS Town Plan highlighted in purple.
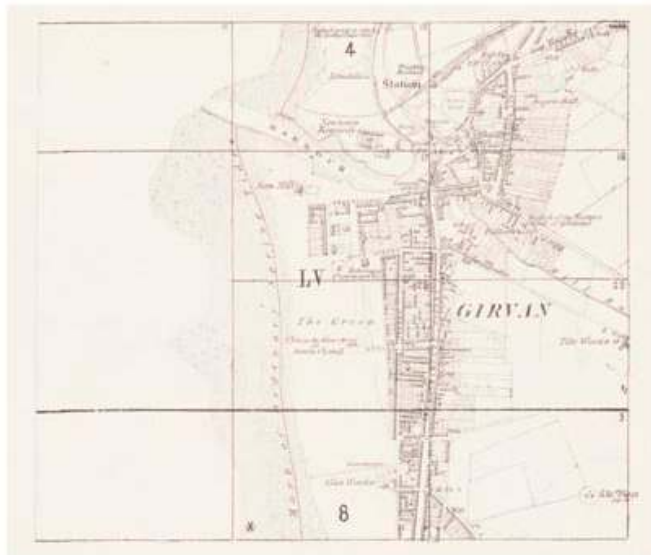
Figure 3 - Webpage showing a clickable graphic index. Clicking the sheet lines opens individual OS Town Plans sheets.
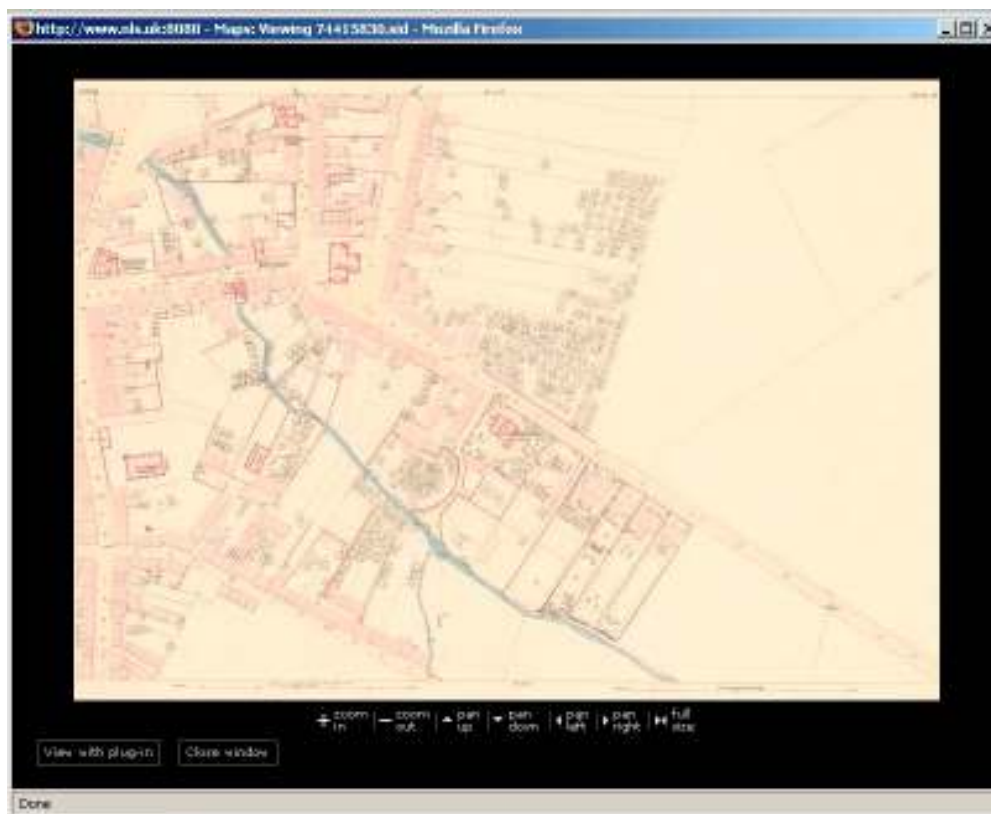
Figure 4 – Browser window showing OS Town Plan sheet in zoomable viewer.

It was decided that the importance of the OS Town Plans meant they deserved particular digital 'curation' and their own distinct development to enhance remote access. The two main improvements we chose to focus on were to create an enhanced interface in which to view the maps, and to implement new functionality in terms of navigation and search methods. To deliver this we aimed to explore and exploit modern technology such as geo-referencing, software such as ArcSDE and ArcIMS and the mashing-up of data using opens source software and Google Maps.

Geo-referencing the Historic OS Town Plans

The obvious next step was to geo-reference the individual scanned map sheets. We had successfully made other series of maps available this way and recognised it as the keystone to our further development aims.

The Geo-referencing process

Unlike previous geo-referencing projects where geographical inaccuracies of the historical mapping needed to be a primary consideration and to be dealt with sensitively, it was felt the large scale and detail of the OS Town plans would be best served by as accurate and precise geo-referencing as possible. We outsourced the geo-referencing in the hope that using professionals dedicated to this field would achieve results with a standard and accuracy that would best reflect the quality of the original mapping, and make future cartographic analysis or comparison with modern mapping possible.
'XYZ Digital Map Company' was awarded the project. They are a locally based company that had successfully geo-referenced Ordnance Survey mapping and the Roy Military Survey Map of Scotland for previous NLS projects. Their technical expertise and experience working with the idiosyncrasies of his-

toric mapping was an obvious attraction. Supporting this was the important local geographic knowledge they demonstrate from their previous project involvement and base in Scotland.

 NLS provided XYZ with the complete set of on thousand nine hundred and eighteen tiff images that make up the digitised version of the OS Town Plan series, a combined total of around 1TB of imagery. The project cost was approximately £7 per image, at a total cost of £14,000. Work commenced in November 2007 and was completed in March 2008. XYZ used the software ER Mapper to perform the work and it was decided that the geo-referencing would use the following process:

1. Crop the images (sheets) to remove borders and border information.

2. Seam the sets of cropped images (sheets) for each town together into composite images - a composite map of each 'whole' town was created.

3. Geo-reference the 'whole town' composites[2]

4. Export the geo-referenced 'whole town' composites as ECW (Enhanced Compression Wavelet) files.

5. Transform geo-referenced 'whole town' composites into sets of geo-referenced tiles for each town.
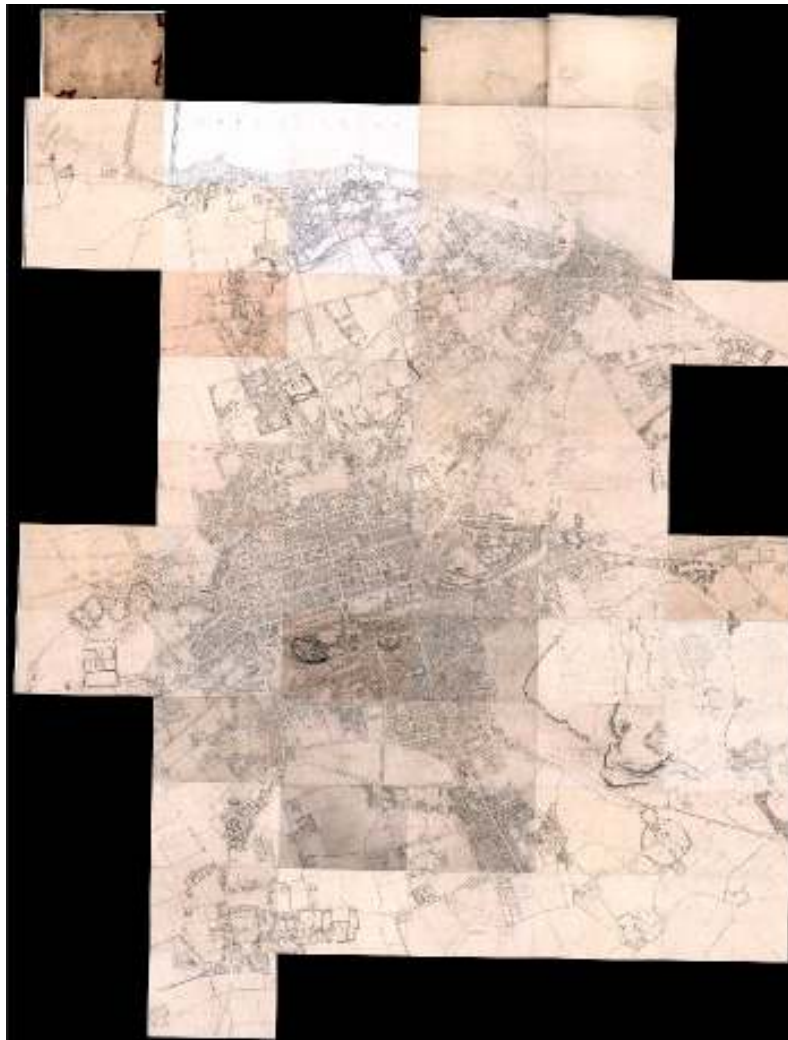

Figure 5 - Geo-referenced ECW showing all the sheets of the Edinburgh OS Town Plans as a composite dataset.

---

[2] The geo-referencing used a first order affine transformation – basic rotation and shift in scale

The second stage of this process offered the most significant choice. We chose to have seamed mosaics of 'whole' towns created and then have these composites geo-referenced as this involved XYZ creating the fewest 'control points'. The downside of this was that it created large files and increased processing times when rendering files. The alternative, to have every sheet individually geo-referenced, can provide greater accuracy but involves numerous 'control points' being created for every sheet. As this is the main manual, time-consuming part of geo-referencing, it determines cost. The geographic accuracy of the Town Plans was deemed high enough that the extra cost to geo-reference the maps in this way would not yield worthwhile benefits.

With reference to stages 4 and 5 above, upon completion, we received the files in two forms. XYZ provided us with an ECW file for each composite town. The ECW files detailed each town as a 'whole' and contained embedded geo-referencing and map projection information. Also supplied were tiled versions of each ECW. XYZ created a programme to split each composite town into a set of equally sized, individually geo-referenced tiff images. Each tile was approximately 40-50 MB, with its geo-referencing information provided in an accompanying tfw (tiff world file). The purpose of this was to ensure the files were a manageable size and in a format that would be compatible with the ArcSDE software we planned to use.

Geo-referencing Improvements

o Composite datasets mean whole towns can appear on a single webpage.
o No need for multiple links so basic navigation is simplified.
o Sheets are seen in context with surrounding and adjoining sheets.
o With the correct software, composite towns can be overlaid on other geo-referenced historical or modern mapping for cartographic comparison.
o A simple interactive interface can be set up with zoom and pan tools.
   This lets us browse the maps in 'real' terms. Our general geographic knowledge translates into our browsing actions. If we know the location we want is west we pan left, if we to look more closely we zoom in.
o Metadata created by geo-referencing allows for the implementation of true geographic searching.
   By geo-referencing using a known projection system and datum, precise points can be navigated to via geographic co-ordinates. If we link such spatial co-ordinates with text-based information we know will be searched, we can create such a system.

Developing a new websites for the Geo-referenced OS Town Plans

We made the decision to develop two further websites using the geo-referenced OS Town Plans, one using commercial ESRI[3] 'ArcGIS Server' software and the other exploring the possibilities of open source software. From developing websites using geo-referenced historical mapping for previous NLS projects[4] we already had an established workflow for 'ArcGIS Server' software. Knowing this method would render a successful result, we felt we wanted to use this as a solid base to try to develop an alternative website interface in tandem with exploring open source software.

---

[3] http://www.esri.com/ - ESRI website
[4] http://geo.nls.uk – National Library of Scotland website of geo-referenced mapping applications

Developing a new website using ESRI ArcSDE and ArcIMS

Two pieces of ESRI ArcGIS Server software were used to create a new website to deliver the OS Town Plans. In the most basic terms, ArcSDE 9.2 was used to load, create and store our geo-referenced datasets and ArcIMS 9.2 governed our front end, accessing the stored imagery and controlling its delivery and establishing our interface.

Loading the Geo-referenced Town Plans with ArcSDE

The primary role of ArcSDE is to act as the database access engine to spatial data within a relational database management system. It lets us load and mosaic single geo-referenced raster images into large geo-referenced datasets and allows storage and access to mass geo-referenced imagery.
We had hoped that the ECW files of the composite towns could provide a shortcut and be loaded into ArcSDE, but this proved unstable. Despite support for ECWs in our version of ArcSDE, it frequently produced 'Invalid Raster Dataset' errors or tried to load indefinitely. Rather than persevere, we decided to revert to the exact process we had used on a previous project and commenced loading the individual tiles that XYZ had created from the composite towns. These tiles were created as we realised the ECWs posed a potential problem and that the manageable size and tiff format, and tfw geo-referencing information would be easily mosaiced by ArcSDE. Each town was split into hundreds of these individually geo-referenced tiles so loading them into SDE took time, essentially to create the same mosaics we already had as ECWs. Trial and error revealed that batches of only 10 –15 images could be loaded into these new composite datasets we were creating so it was a lengthy process. One benefit, however, was that the ArcSDE GUI allows for background transparency to be set. This meant that we could easily remove the black or white backgrounds that were visible around the edges of the ECW versions of the composite towns. After a few weeks of loading, we had a new raster dataset containing a composite of each whole town loaded into ArcSDE.

Website delivery using ArcSDE and ArcIMS

Our ArcIMS OS Town Plans application (http://geo.nls.uk/ostowns/) is based around an easily customisable interface that uses a combination of Macromedia Coldfusion and JavaScript code. Of benefit to users is that it requires no downloads or plug-ins to run. It lets us set up layers of geo-referenced mapping, decide on the layer order that they overlay each other and determine their visibility at different scales. This is all done by editing simple html, xml and JavaScript code as well as an .axl file[5]. The layers of geo-referenced mapping are displayed on an html webpage inside a fixed frame that acts as a 'portal' for the mapping. Dynamic html allows the user to navigate inside this frame, with panning, standard zooming, and dragable 'box' based zooming actions. The ColdFusion connector translates requests from users' web browser clients for the ArcIMS image server to then generate a map on the html page. As well as layers featuring geo-referenced datasets we can include vector-based imagery. Such shape files can be used to set up geo-referenced series of polygons or points that may correspond to towns or gazetteer points. ColdFusion allows tools for selecting polygons or points and can also be used to query and return information from associated databases. This allows for the setup of the type of geographic searching we discussed earlier.
We used a combination of these elements to set up the OS Town Plans Website. The frame within which the OS Town Plans mapping will appear constitutes the main part of the page. Initially it shows a zoom-

---

[5] The .axl file is an XML-based file describing the core parameters and layers in the ArcIMS image server.

able map of the whole of Scotland and features a selection of navigation tools. Due to the very large scale of the town plans we set them to appear only when the user is zoomed close in. To facilitate navigation to the zoom-level at which the town plans are visible, we included layers of mapping of Scotland at different scales[6] and set their visibility dependent on zoom-level. As the user zooms in, the mapping within the frame becomes more and more detailed. As a further guide a shape file was added. It is always visible and uses a clear circle to mark each town and alert the user of their location. We also set our geo-referenced Ordnance Survey 6 inch to the miles mapping of Scotland as a base layer. Its visibility was set at the same zoom-level as the OS Town plans and coded to appear underneath them to provide a historical context of mapping from a similar period. We enhanced functionality of the webpage by adding a searchable gazetteer of streets. An Access database of modern Scottish streets[*] was edited to contain only the street names and associated geographic co-ordinates of the towns that form the OS Town Plans series. A text-based search box is clearly visible on the right of the webpage and any text can be entered and searched for. ColdFusion searches our gazetteer of street names database and matching results are generated on the html page. These results are clickable, and, when clicked, the co-ordinates from our database for the selected street are used to navigate the map straight to the corresponding boundary formed by these points.
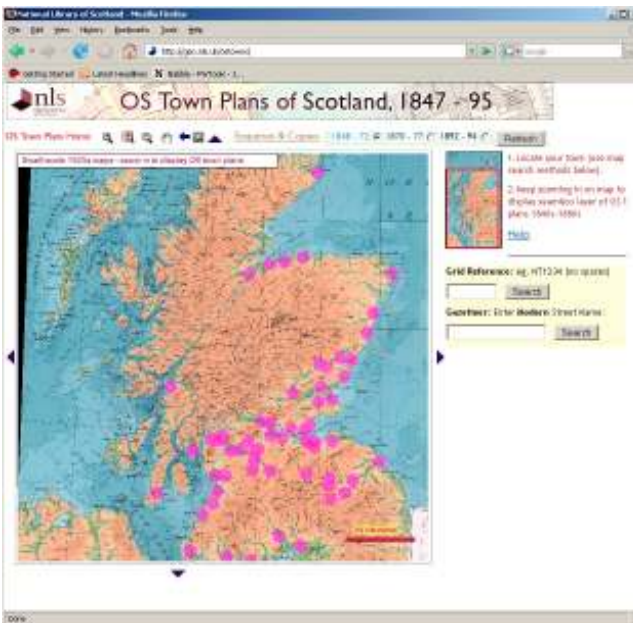

Figure 6 - ArcIMS interface. This shows the initial display of the mapping with shape file circles marking the location of OS Town Plans.

Figure 7 - ArcIMS interface. This shows the mapping zoomed in to display the OS Town Plan for Edinburgh in detail. On the right hand side you can see a gazetteer search was used to locate Charlotte Square.

This set-up definitely delivers the geo-referenced mapping in an enhanced interface with better navigation and improvements possible by geo-referencing are apparent. The successes delivered were:

o   Whole composites of towns are easily browsed with simple tools and shown comparative to other mapping.
o   Layers allow all the OS Town Plans to be contained in one location on one web page.
o   ArcIMS allowed us to introduce geographic searching.

---

[6] Georeferenced Ordnance Survey Quarter Inch (1921-1923), Bartholomew Half-Inch (1926-1935), and Ordnance Survey Popular Edition – One Inch (1920-1930) were used.

o   A single, modern street name gazetteer could be included and any search returns results for every town across the OS Town Plans series.

o   The website is easily expandable. ArcIMS is flexible and we can easily remove or switch layers, shape files and linked databases to enable ongoing improvements.

There were, however, distinct limitations inherent in this presentation.

o   Slow performance delivering the mapping.

ArcIMS performs complex processing every time it translates a request from a web browser client to display the mapping. Reduced resolution pyramids are built using ArcSDE for the datasets to help speed up display at different zoom-levels, but performance times were still disheartening. Refreshing mapping when panning and zooming takes as long as 10-15 seconds, even on a fast internet connection.

o   Frame-based viewing of the mapping looked limited and outdated

Compared to full page viewing available in Google Maps style interfaces it seemed outdated. Frame size was enlarged to allow a greater area of map to be viewable but was done knowing slower internet connections would struggle.

o   Complex Interface

The interface itself, while useful, is not exactly simple. Its appearance is application-based and may confuse or deter some users. Combine this with poor access for slow internet connections and it does not make for a universally accessible website.

o   Bugs in ArcIMS services

Bugs in refreshing ArcIMS services regularly led to downtime, network changes at NLS often meant reconfiguring ArcIMS settings to get it to run properly and the difficult software upgrades also led to settings having to be changed.

Developing new webpages using Open Source Software, OpenLayers and Mash-Ups

Open source is software that is distributed with the source code open to be seen, adapted or added to by the user. It is often free or distributed with the hope of donations or promotion in return for use. The geographic sector has seen large growth in this area, particularly in the form of mash-ups. A mash-up, in this sense, combines spatial information and different mapping together and delivers it through a single customisable interface. While sounding technical, such applications are incredibly familiar and constantly expanding. Google Maps and Google Earth, Microsoft Virtual Earth, OpenStreetMap, OpenLayers and Yahoo Maps are obvious examples of mash-ups that are readily used and have content added by many users. Our intention was to use such resources to create mash-ups of our geo-referenced Historical OS Town Plans and modern satellite imagery.

Initially, NLS had avoided open source software due to a perceived lack of stability and support. As librarians and not true technicians or developers, we were reticent about the 'language' of open source. We felt uncertain about the open source model leading to the need to alter chunks of complex code to make things 'do what we need', and the use of non-GUI (Graphic User Interface) command-line and programming-based working methods.

The limitations of the ArcIMS website and a number of factors combined to encourage us to look beyond the ESRI ArcGIS model and try open source software.

The sheer popularity of Web 2.0 and open source applications such as Google Earth, and Google Maps was hard to ignore and the use of open source software such as OpenLayers to create web mash-ups compatible with these resources was becoming commonplace. To stay relevant our own exploration felt imperative. Mashing-up our OS Historical Town Plans with modern satellite imagery from Google Maps or Microsoft's Virtual Earth, was not just contemporary though, it seemed a perfect way to dis-

play this mapping. The incredible detail of our OS Town Plans makes them perfect for cartographic comparison and such a mash-up would produce this in a ready-made populist interface.

The perceived support we associated with commercial software was not wholly apparent with the ESRI ArcGIS products. Even though we had established a workflow, it was a result of much sweat and toil. There was often a difference between supporting literature and the reality of the ESRI products, and, when struggling to establish our working processes, the first place to turn for assistance was primarily forums and help pages. We no longer felt it always equalled a distinctly superior service. The increased popularity of open source software meant that the communities involved were growing, and with its focus on development based on users' needs and discoveries, we felt a similar forum-based support network was available should we need it. Vitally, NLS had made a strong contact within the open source community. Petr Pridal was working to develop exactly the type of software and processes that would enable us to present the OS Town Plans as the open source mash-up we had in mind.

Using OpenLayers

We decided to create a mash-up and a new webpage for each town, and chose to use OpenLayers software to deliver our open source mash-up. OpenLayers provides the simple, familiar interface and full-screen display of a Google Maps style interface, but can pull mapping from multiple resources (Google, Yahoo, Virtual Earth, OpenStreet etc) into its interface. The OpenLayers interface presents each company's imagery in the form of layers that the user can switch off and on. Being open source software, we can edit the html to determine which base layer is displayed by default, or edit further to remove these layer options. As the satellite imagery varies greatly between the likes of Google, Yahoo and Virtual Earth, this was of particular relevance. Using the best imagery available was of prime importance to provide good cartographic comparison for the OS Town Plans at the detailed zoom-levels that befits their detail.

OpenLayers also provided a solution to the slow performance issues surrounding our ArcIMS delivery. It avoids the heavy software processing ArcIMS undergoes every time it delivers imagery by displaying mapping built of rendered cartographic tiles at fixed scales. To do this, OpenLayers uses the Tile Map Service (TMS) developed by OSGeo[7]. Tiles are pre-created with a specific projection and datum and at a set of zoom-levels and then stored in corresponding directories. As Google Maps, Microsoft Virtual Earth and other geo-data Application Interfaces (APIs) use the same projection[8] and tiling profile, any mapping we pre-create tiles from is universally compatible. The main issue for the user is the projection and datum stored for their geo-referenced imagery and ensuring this is correctly translated when performing the tiling process itself. OpenLayers does not require any specific server technology; the only requirement is the server space to store the rendered tiles. OpenLayers directly accesses the tiles using URL requests from the html of the actual web page you create and JavaScript commands ensure the correct tiles are pulled into the browser window.

Tiling the OS Town Plans for OpenLayers with Gdal2tiles

We used a combination of command-line-based processes to tile our maps. We used Petr Pridal's gdal2tiles[9] programme to render tiles from the composite geo-referenced versions of each of our OS

---

[7] http://www.osgeo.org/
[8] The co-ordinates presented to the user are latitude/longitude on the WGS84 datum. Internally the applications use the Spherical Mercator projection, with coordinates in metres, and treat the earth as a sphere rather than the WGS84 ellipsoid.
[9] http://www.klokan.cz/projects/gdal2tiles/

Town Plans. Gdal2tiles also creates an OpenLayers.html file for the tiled mapping. This sets up the page and contains all the OpenLayers code that will pull our tiles and the different base layer mapping into the browser window and create our mash-up.

Gdal2tiles operates using command line instructions/functions and was still in the development phase so this stage was completed with great support from Petr Pridal. Unfortunately, gdal2tiles did not support tiling of ECW files so we had to pre-process our composite geo-referenced datasets.[10] The FWTools[11] command shell was used to create compatible tiff images from our ECW files. This process is outlined below:

1. A nearblack[12] command was run on each ECW file using FWTools.
   This uncompressed the ECWs and created 'giant' geo-referenced tiff files and converted near to black or white background colour into a solid black or solid white.
2. A gdal_translate[13] command was run from FWTools to translate the geo-referencing embedded in the 'giant' tiff images into the correct projection for accurate overlay when mashed-up with satellite imagery.
3. A further parameter was added to the gdal_translate command to render the background colour of each tiff as a transparent alpha layer.
   The nearblack command had rendered each image's background as entirely black (pixel value 0,0,0) or entirely white (255,255,255), this gdal_translate command can convert a colour with a single pixel value (e.g. 0 or 255) as transparent.
4. Gdal2tiles[14] was run via the OSGeo command shell on each 'giant' tiff to create the tiles for use with OpenLayers.
   This rendered the tiles, created them in the required directory structure, and as stated, produced an OpenLayers.html page set up with the correct JavaScript to access and display the tiles.

The file size of the 'giant' tiff images was problematic. We did not have the local hard drive capacity to store this amount of raster imagery; some of the tiff images created were over 100Gb, and, in the case of some very large towns, we did not have available hard disk space to create these 'giant' tiffs. We tried creating these'giant' tiffs directly on our server where space was available, but network security prevented this working. We therefore had to run the steps above on each ECW one by one, deleting the 'giant' tiff created for each after successful tiling with gdal2tiles.

Upon completion of a set of tiles for each town, all that was required were some simple html alterations to set up each OS Town's web page perfectly and to copy these pages and all the folders of tiles to our server ready for display online with OpenLayers.

Assessing our new OpenLayers OS Town Plans webpages

We delivered our OpenLayers OS Town Plans (http://geo.nls.uk/maps) using links from our geo.nls.uk/maps webpage. It is a basic html page with a growing set of links to NLS projects using overlays and mash-ups. A series of links, one for each Town Plan's OpenLayers mash-up, were added to this

---

[10] http://www.maptiler.org/ - Petr Pridal has now developed a simple to use GUI based application called Maptiler for tiling raster images. A future version of this may be developed to support different file formats, including ECWs.
[11] http://fwtools.maptools.org/
[12] Example nearblack command: nearblack –o C:\edinburgh_export.tif C:\edinburgh_input.ecw
For more info http://www.gdal.org/nearblack.html
[13] Example of gdal_translate command: gdal_translate –of vrt –a_nodata 0 –a_srs EPSG:27700 C:\edinburgh_export.tif C:\Edinburgh_ready2tile.vrt
This creates a .vrt (virtual raster file) for your 'giant' tiff. You specify this file when running the gdal2tiles command to tile the tiff images. More info: http://www.gdal.org/gdal_translate.html
[14] Example gdal2tiles command: gdal2tiles C:\edinburgh_ready2tile.vrt

page. When clicked, the OS Town Plan we have selected appears in the browser using OpenLayers interface. It is displayed as a semi-transparent layer overlaid against whichever satellite or mapping we have specified in the page's html. The interface is very simple; a slider bar appears on the left-hand side for zooming, and layer options are selectable on the right via simple radio buttons.
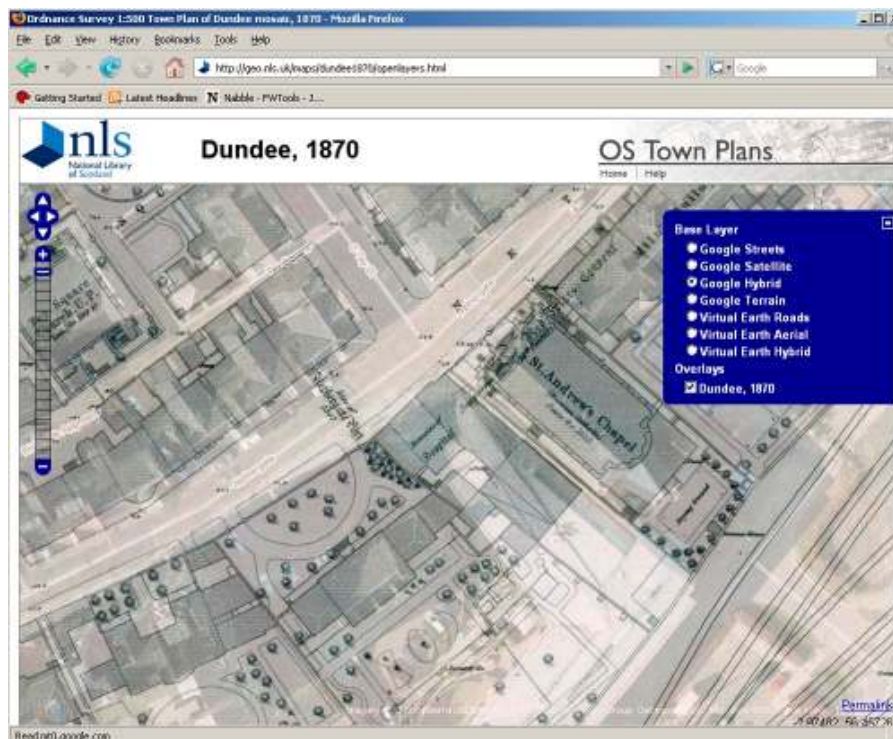

Figure 8 - OpenLayers interface. This shows OS Town Plan for Dundee (1870) mashed-up with Virtual Earth Aerial imagery.

The simple interface was a definite success, either instantly familiar due to the popularity of similar interfaces or easy to understand thanks to basic, easily interpretable functions. The maps are clear and the base imagery provides a strong cartographic comparison for our historic plans. Certain towns do not have good satellite coverage from Google, Virtual Earth, or any of the other major mapping providers, but the ability to alter OpenLayers' html and switch base layers meant we could select the most suitable coverage or choose to pull in standard street mapping instead. The geo-referencing was also successful, adequate at even the highest zoom-levels. As better satellite coverage becomes available, we can easily switch our base layer again to mash-up our historical mapping and accommodate this.

OpenLayers, as hoped, fulfilled our aim to deliver our geo-referenced mapping with greater speed and stability. This has been a definite success. The pre-created tiles take only a few seconds to mosaic on screen between zooms and pans, compared to the 10-15 seconds with ArcIMS, and direct storage and access of these from our server means this system is as stable as it is.

Overall, contemporising part of our collection in this way has had a strong impact. Presenting the OS Town Plans in such a familiar context, mashed-up with satellite imagery and modern technology that is now so populist, has created significant interest and appeal. Future development might even involve other institutions creating mash-ups by directly accessing our tiles imagery and pulling them from our server into their own applications.

Conclusion

Using OpenLayers mash-ups as a digital 'curation' option for the OS Town Plans has been a distinct success and we have already used it to present other geo-referenced mapping. Presently, however, it re-

mains a supplementary option and whether we will solely develop and present a project using OpenLayers and mash-ups is debatable. The lack of a geographically indexed gazetteer means ArcIMS still has at least one very distinct technological advantage. It is likely that such additions are available or could be developed, but knowing the support we received from Petr Pridal just to tile and produce the OS Town Plans as overlays, we still approach the use of open source software with a little caution, aware of our own technical limitations. With workflows established, we can now pick and choose the most suitable delivery method or, when beneficial, develop a project on three fronts: standard html webpage delivery of single map sheets; ArcIMS delivery of geo-referenced mapping; and OpenLayers mash-ups of geo-referenced mapping. This comprehensive approach to the provision of resources to access our digital collections seems to be the most universally inclusive; different delivery options with varying functionality recognise the different technological needs, abilities and access our users have.

References

[See the relevant footnotes within text]