

Dynamic web-mapping using Scalable Vector Graphics (SVG)

Paul Clarke

Web-mapping is a growing and popular use of GIS but creating even a simple map is often a non-trivial and costly task. One potential solution to this problem is Scalable Vector Graphics (SVG), an XML based 2D graphics standard. SVG offers rich client-side interactivity and unrivalled graphical quality, without the need for an expensive mapping/applications server. We introduce SVG technology and provide an overview of its benefits for both producers and users of interactive maps. We then outline the drawbacks of SVG web-mapping and how it compares to existing approaches. Different methods for generating SVG maps are compared including hand-coding, exporting directly from ArcGIS9+ or using extensions to ArcView/ArcGIS which allow the user to automatically create dynamic maps from geospatial data. Finally, we explore some example applications that demonstrate the key advantages of SVG.

What are the issues with web-mapping?

Every day, people use online maps for a variety of uses, from planning road journeys to exploring unknown locations. Although web-mapping is the most simple and widely accessible use of Geographic Information Systems (GIS), and a great many useful sites exist, traditional web-mapping technology has several drawbacks. One of the main issues for those providing maps is the cost involved. Established GIS application servers, such as ArcIMS™, are usually required to handle map manipulation and generate map “snapshots” that are sent to end-users. Every time the user wants to change or interrogate the map the server has to update the view and send it to the user, which can create an impression of poor interactivity. Furthermore, setting up and maintaining a GIS application server is often not a cost-effective or viable option for small and medium sized companies.

One approach to this problem is a better way of handling map interactivity on the client side, removing the reliance on a GIS application server. Solutions to date include maps based on java-applets or Flash. However, their proprietary nature can make it difficult to generate mapping data. This paper investigates a new approach to web-mapping using Scalable Vector Graphics (SVG). We investigate what SVG is, looking at both benefits and drawbacks for mapping applications and explore the different ways that SVG maps can be created.

What are Scalable Vector Graphics?

SVG is a generic standard for 2D graphics on the web, based on the Extensible Markup Language (XML). With SVG it is possible to create simple graphics (see **figure 1-1**) in just a few lines of text (see **figure 1-2**). Adobe, amongst others, provides a free SVG plug-in for Internet Explorer and some browsers (e.g. Mozilla™, Firefox™) support it directly. It therefore doesn't cost

anything to start experimenting with this technology. SVG defines individual shapes as mathematical vectors, meaning that the quality of the image remains the same, regardless of scale. This can be a major advantage as it eliminates the pixelisation that occurs when you zoom in too far on normal bitmap images.

Figure 1-1: Simple example of SVG

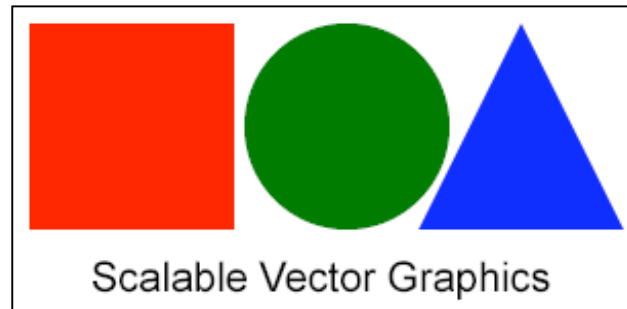


Figure 1-2: The code used to generate the above graphic

```
<?xml version="1.0" ?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20010904//EN"
"http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">
<svg>
  <rect x="10" y="10" width="100" height="100" fill="red"/>
  <circle cx="165" cy="60" r="50" fill="green"/>
  <path d="M200,110 l 100,0 l -50,-100 l -50 100 z" fill="blue"/>
  <text x="40" y="140" font-size="20">
    Scalable Vector Graphics
  </text>
</svg>
```

Are Scalable Vector Graphics suitable for web-mapping?

Although SVG was designed as a generic graphics language it has obvious applications for web-mapping. The following section looks at both the pros and cons of SVG.

Pros

Quality - The graphical quality of SVG maps is outstanding at all scales as it uses vectors and anti-aliasing for a smooth appearance. A suite of graphical effects such as drop-shadows, transparency, gradient-fills and animation are supported. In addition, standard raster images (e.g. JPEGs and GIFs) can be seamlessly integrated into SVG maps.

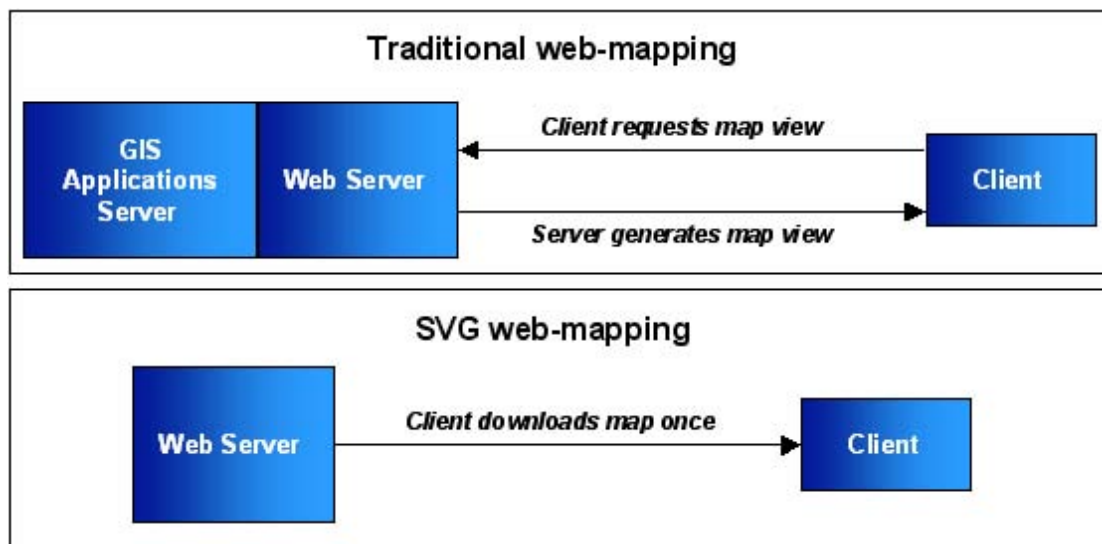
Manipulation – Every map feature is defined as an SVG vector object with associated methods and attributes that enables all manner of feature interactivity. The majority of GIS data involves a geometric shape linked to non-spatial data and the SVG language can model this with ease. Using

scripts embedded within an SVG file a whole range of events can be triggered when the user clicks on or hovers over a feature. For example a feature might display a useful tool-tip or launch a new web page or application when clicked on. Another key advantage is that SVG objects can change position dynamically on the client-side without the map having to be redrawn. This opens up the possibility of moving objects in real-time (e.g. tracking applications) or the user themselves being able to create or edit map features or annotations.

Cost - The major advantage for those who want to generate a web-map is cost. SVG can be very cost-efficient as it eliminates the need for traditional GIS application servers (see **figure 1-3**). An SVG map can be uploaded to a standard web server such as Apache (or CD-Rom, DVD etc.) and be treated just like any normal web page, image or animation. As the functionality for viewing and controlling the map is embedded as scripts within the SVG map, all manipulation is performed on the client side. This reduces subsequent client-server interaction and provides a responsive environment for the user. Standard GIS functions such as turning layers on and off, measuring and zooming now take immediate effect without the usual "Redrawing map" message.

Network communications - With a single SVG map, the data is downloaded onto the client in a single transaction. This means that although it may initially take longer to download, subsequent interaction can then be performed without further intervention from the server. This can lead to less delays and reduced network traffic.

Figure 1-3: Comparison of map delivery in traditional and SVG web-mapping



Cons

File size – One of SVG's advantages is also a disadvantage for complex maps as, like any large file, it can take a long time to download on low bandwidth connections. One solution to this problem, employed by some SVG mapping tools, involves splitting up the map into a grid of tiles which are loaded as and when they are needed. This means that all the map tiles are stored on the server but only those currently in view are accessed by the client. This has the advantage of a fast initial download but the disadvantage of increased network traffic as new data is periodically streamed to the client.

Non-binary – SVG is an XML language which is text-based. Therefore feature-rich maps can result in large files. To address this the language itself has been designed to be as compact as possible (e.g. you can define relative as opposed to absolute co-ordinates). The Adobe SVG plug-in can also automatically decompress SVG data compressed using GZIP. This approach typically reduces file size by up to 80%.

Data update – A key advantage of most traditional web-mapping systems is that they allow data to be updated centrally and these changes immediately filter down to the clients. With SVG, once a map has been converted from source data, updating the source data will have no effect on the SVG map. As described in the previous section it is possible to update SVG features dynamically but this is not well suited to changing large volumes of data. Therefore SVG is best suited to maps with background data that will change little over time. An alternative approach is to generate SVG on-the-fly on the server, although this is not really practical for large maps.

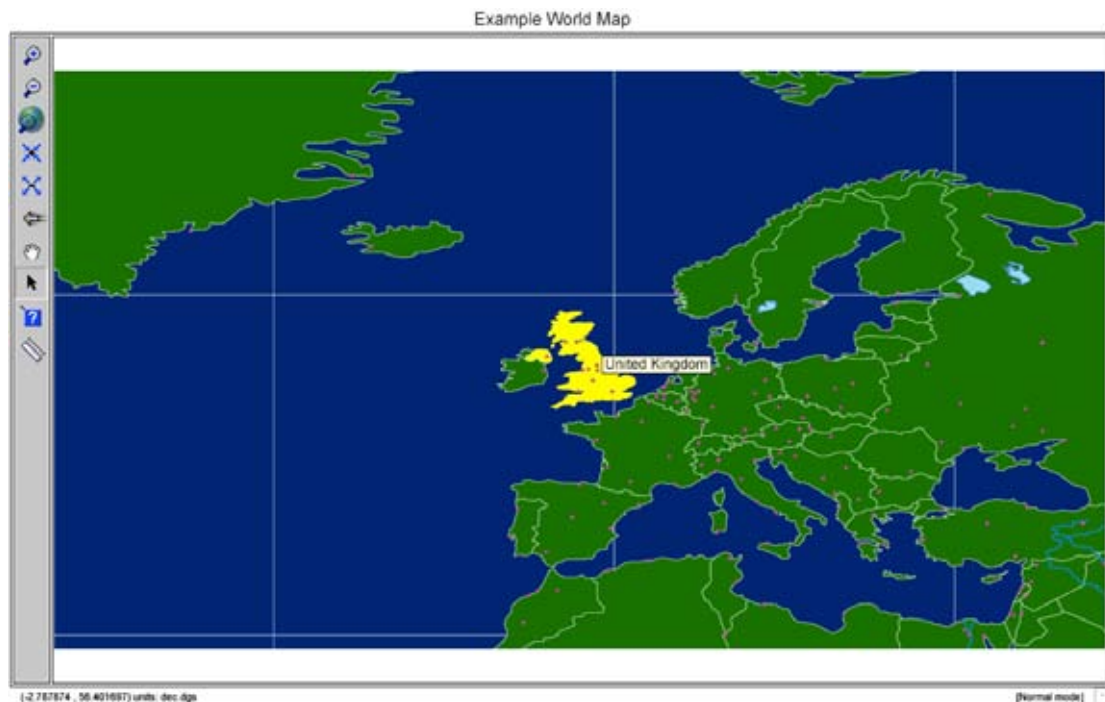
Data exposure – Traditional web-mapping servers generate a snapshot of the map data so users are not exposed to the raw data. With SVG mapping each map feature is a vector object complete with geographical co-ordinates. Given time it could be possible for someone to download an SVG map and reverse engineer the data to obtain the original source data. Although this can be mitigated against it is a risk that must be considered, especially for sensitive or copyrighted data.

How can you create Scalable Vector Graphics maps?

Several methods can be used to generate a map using SVG and **Figure 1-4** shows a typical example. A number of these methods are examined below.

Manual coding – All you need to generate SVG is a text editor and it is possible to hand code SVG just like any other language. However, this is unsuitable for anything more than a few simple shapes, and certainly doesn't make sense for a map with hundreds of features as this could take days or even months to produce!

Figure 1-4: Simple example SVG map with toolbar



Copyright © QinetiQ Ltd 2005

Programming – As SVG is XML it is easy to generate programmatically (or using a transformation style sheet). It is relatively simple to create a program in Java or C/C++ etc. that could access a spatial database and output an SVG map dynamically. Any program or programming language that enables the creation of text files could potentially be used to generate SVG maps including server-side scripting languages such as ASP, JSP and PHP. A customised programming solution ensures that the output SVG maps exactly meet the user requirements. Conversely the major drawbacks are that this approach requires time to set up and fairly advanced programming skills.

Export from drawing packages – A number of standard graphics packages such as CorelDraw™ and Adobe Illustrator™ have an option to export images as SVG. A number of SVG dedicated graphics packages are also available including JASC WebDraw™ and Sketsa™. Generic SVG tools can produce simple maps quite effectively but are not capable of generating maps with more complex GIS type functionality.

Export from ArcGIS9+ - ArcGIS (version 9 and above) includes an SVG option as part of its “Export map” functionality. Like the output from the drawing packages these maps are extremely simple, are not well suited for zooming in/out etc and do not support traditional GIS operations.

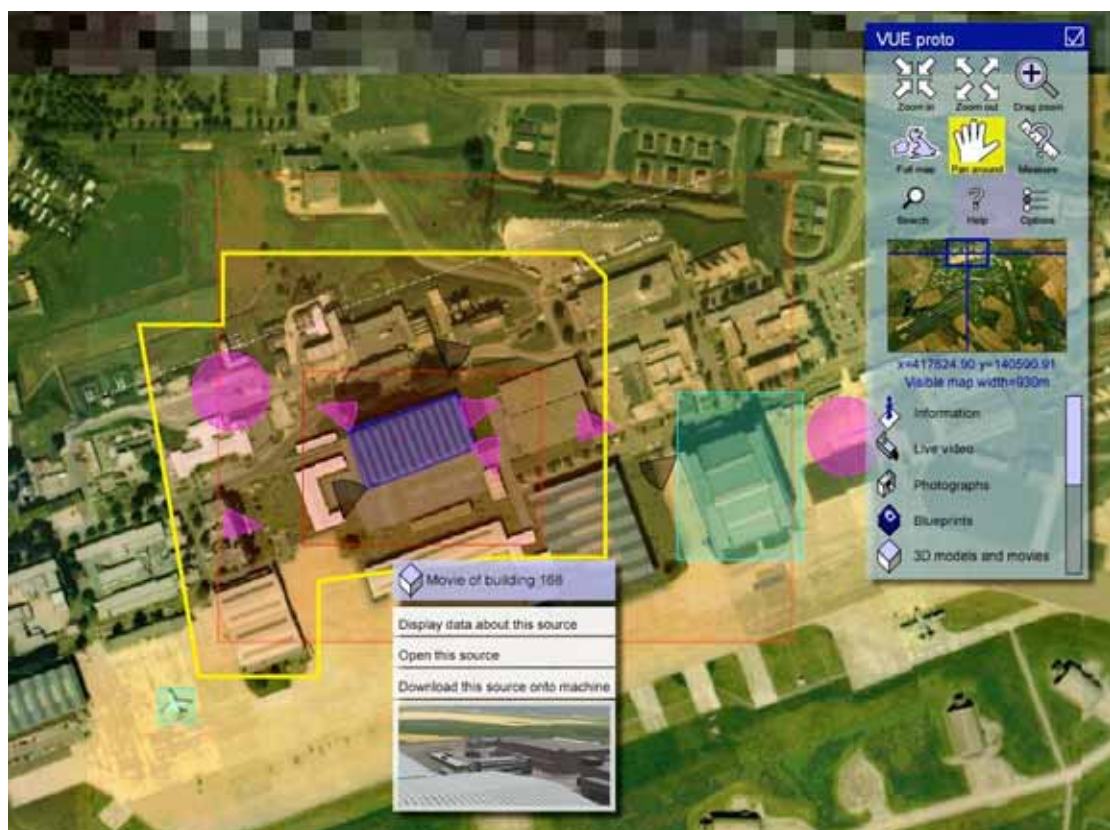
Dedicated GIS publishing tools – A number of GIS extensions are becoming available which allow a user to produce SVG maps without the need to manually code SVG or be a programming expert. These tools publish SVG maps of GIS data whilst combining typical web-mapping functionality (zooming, panning, feature information, searching, measuring etc.).

Extensions such as EasySVG™ and SVGMapper™ are available for ArcView 3.x™ and ArcGIS™.

The key advantage of such tools over other solutions is how quickly maps can be generated. The basic process involves loading data into the GIS, setting up the map output options, generating the SVG files and uploading them onto a standard server for people to view. This compares favourably with traditional approaches that generally require a lot of training, set-up and customisation before maps can be produced. For any individual or organisation that would like to gain the benefits of interactive web-mapping but are currently prohibited by cost, SVG mapping extensions could be the ideal solution.

What can be these maps be used for?

Web-mapping has numerous applications which can be achieved with SVG. The following two examples highlight the ability of SVG to support dynamic information updating.



Copyright © QinetiQ Ltd 2005

Figure 1-5: Advanced geospatial information portal with real-time updates

Information portal – A sophisticated information portal has been developed which allows users to easily access geo-referenced information (photographs, blueprints, reports, animations, 3D models etc). The simple and attractive interface was developed using SVG in just 20Kb of code (see **figure 1-5**). The

interface allows users to search for relevant information sources which are then dynamically displayed on the map. As new information sources become available they are automatically integrated into the map, and the user informed. Accessing any of the data sources involves a simple one-click operation. A number of tools (such as interactive image viewers) are available and also written in SVG. The portal can display both vector and raster maps and can access Web Mapping Services based on the Open Geospatial Consortium (OGC) standards.

Person locator - Within the Distributed Technology Group (DTG) at QinetiQ, people are spread over a wide area and a system was required which could show people's location on a map, provide contact details and indicate whether they were using their PC (see **figure 1-6**). A solution was found by combining a collaboration tool, enAble-Me (which monitors keyboard/mouse activity) and the EasySVG for ArcGIS™ extension.

Figure 1-6: Example real-time SVG mapping application depicting office locations and whether people are currently at their desk



Copyright © QinetiQ Ltd 2005

Blueprints of the buildings were loaded as CAD files into ArcGIS and converted into maps using EasySVG. The online 'presence' of each person within DTG is monitored and displayed on the map (e.g. green indicates a person is at their PC) and further information can be accessed by clicking on a name. The system has since been extended to monitor the diary system so if someone is out of the office his or her name is highlighted in blue on the map. Also, using a network of Wireless access points, the exact position of

people with PDAs can be tracked on the map in real-time. The map is also available to visitors who can use it to find a room and whether their point of contact is available.

Conclusion

SVG mapping offers a new alternative for interactive maps based on open standards. The first key advantage over traditional methods is that no GIS application server is required which reduces the cost and time it takes to distribute maps. The second key advantage is that the map manipulation is performed on the client side, reducing network traffic and providing a richer experience for the user. Although it is possible to generate SVG code manually by far the easiest method is to use a dedicated SVG publishing extension available for ArcGIS™. These automatically convert geospatial data into SVG and some support map customisation and powerful GIS functionality. Despite being relatively unknown at present, SVG has the power to revolutionise the creation, distribution and viewing of web-maps.

References

SVG specification standard: <http://www.w3.org/Graphics/SVG/>
Adobe SVG viewer plug-in: <http://www.adobe.com/svg/>
SVG wiki: <http://www.protocol7.com/svg-wiki/>
Introduction to SVG web-mapping:
http://www.carto.net/papers/svg/index_e.shtml

Copyright © QinetiQ Ltd 2005. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies.

Author information

Paul Clarke
Research Scientist
QinetiQ
AT010, Malvern Technology Centre, WR14 3PS, United Kingdom
+44 (0) 1684 896015
pdclarke@qinetiq.com