SVG Pottery Documentation

Release 1b

Stefano Costa

Sep 14, 2021

Contents

SVG Pottery is a set of best practices, how-tos and recommendations to ease the process of publishing pottery drawings on the Web.

In short we strongly recommend that:

1. you publish your drawings on the Web, and not just on paper, not just inside a PDF article, but on their own

2. you use SVG as the format for publishing your drawings

The rest of this documentation goes into the details of why and how, and explores some issues and possibilities with this model. You can just go ahead and start using SVG if you like.

Some software tools are also being developed to assist archaeologists with simple, quick ways to publish their drawings on the Web.

This documentation is maintained in a Git repository at Codeberg.

If you want to cite this guide in your academic work, you can refer to the following paper, that was presented at the Computer Applications in Archaeology conference in 2012 [COSTA2014]:

The SVG version described here is SVG 1.1, the current W3C recommendation.

Contents:

1 Why SVG?

Or, in other words, why not another vector graphics format, like DXF, DWG, PDF or Adobe Illustrator's?

- SVG is an **open standard** defined by the W3C
  - all modern browsers support directly visualising SVG
  - lots of vector graphics software support editing SVG
- SVG is XML, and this brings several advantages like
  - ability to do manual editing with a text editor
  - availability of advanced processing tools
  - ease of translation to other formats, both vector and raster
- SVG is part of official recommendations and guidelines:
  - it is one of the two *preferred* formats for vector graphics at the Archaeology Data Service (together with DWG)
  - it is the only recommended format for vector graphics in the MINERVA EC Guidelines for Digital Cultural Content Creation Programmes
  - unfortunately the Deutsches Archäologisches Institut still doesn't mention SVG in their IT-Leitfaden (recommendations)

In this chapter, we are first going to look at some other popular formats, with their limitations, and then we will examine why SVG is the best *available* choice for publishing pottery on the Web.

1.1 Other Formats: Why Not . . . ?

There are several possible digital formats for pottery drawings. Some are commonly used, some are not, but all have in common one thing: they are almost never used for publication (with the exception of PDF), where print or raster images are the current standard.

An example of raster images used to disseminate line drawings of archaeological pottery is Roman Amphorae: a digital resource (University of Southampton 2005), where a typological collection is presented and for every type of amphora one or more profiles are available as JPEG images. In this section we discuss why most formats are not suitable for Web publishing.

DWG

DWG is a proprietary format developed by Autodesk Inc., and it is the native format of the AutoCAD family of software programs. AutoCAD is often used for digitising pottery drawings, for several reasons. First, it is used for other common tasks of archaeological routine, such as site plans. Second, it has a comprehensive set of drawing tools. Third, AutoCAD guarantees good-quality printed graphics. But DWG is not a good choice. The first main problem with DWG is that most often it is used, but not chosen. Rather, it just happens to be the default format of a popular software. The second problem is that DWG is proprietary. Every version of AutoCAD introduces a new version of the format, incompatible with earlier versions. There is no open source software capable of reading and writing all DWG versions. No web browser has or is likely to have in the future support for DWG. As a consequence, DWG is not suitable for being published on the Web.

DXF

DXF is an interoperable version of DWG. Its specification is open (even though it remains property of Autodesk Inc.) and there are several open source libraries and programs capable of reading and writing DXF. However, DXF files tend to be very large in size compared to the corresponding DWG files, and have the same complex framework of incompatible versions.

PDF

PDF is an open standard, registered with ISO and part of lots of recommendations as format for archiving documents of many different types. PDF is a vector format, that is, vector graphics features (lines, polygons, text) can be scaled retaining their quality. However, objects contained in PDF are almost completely unstructured: if three lines form a triangle, this is not relevant in PDF, and therefore there is no way to extract the information "this file contains a triangle". For this kind of purpose, pottery drawings are no different than triangles. PDF is suitable for archiving because it is easy to create but generally not easy to edit and PDF documents are usually intended to stay as they are. Images can be included in PDF also as raster images (with JPEG compression), so using PDF for vector drawings is an ambiguous recommendation. Moreover, PDF on the Web is an ambiguous presence: despite being very widespread, it is not natively supported by web browsers (unless using additional plug-ins) and it is treated as a downloadable file. PDF files containing pottery drawings are for example the ones published by the FOLD&R journal.

EPS/AI

EPS and AI are two variants developed by Adobe Systems Inc. for their Illustrator software. They share almost all problems of PDF, and they are not native Web formats. To make things more complicated, recent versions of AI are proprietary and not interoperable.

GML

GML is an XML language and an open standard of the Open Geospatial Consortium. GML can be used to describe any spatial feature, as commonly seen in GIS. Because all vector spatial features are also geometric features, it is possible to represent a vector graphics object in GML. GML has two significant limitations. First, it is not a vector graphics format, and normal image viewing programs are not able to deal with it. Second, it is deeply rooted in the geospatial domain, and therefore all data is assumed to be in a geospatial coordinate reference system (CRS). If no CRS is specified, WGS84 geographic coordinates are assumed, and most importantly there is no straightforward way of marking a GML file as having a local, non-geographic coordinate system.

SVG: strengths and limitations

Among the reasons why SVG is important in the specific case of pottery drawings, some are significant for the purpose of sustainability.

First of all, SVG can be used for disseminating high-quality vector graphics on the Web instead of raster images: in other words, this means that drawings will have a quality equal or better than a 1:1 print, and it will be also possible to perform manual or automated editing (e.g. based on personal taste, or requirements from publishers, colours and hatchings can be easily changed accordingly).

Secondly, SVG works across most vector graphics programs, and has a rich community focused around the format and not the software. This gives a higher sustainability to archives of SVG drawings: should one program become obsolete or unmaintained, the migration path would be comparatively easy, because no conversion of format would be needed. Finally, dissemination on the Web is a good way to preserve data, thanks to the existing mirroring services, and to the inevitable downloads that will happen.

SVG is not without problems though, and it should be clear that SVG is far from perfect as a native format to create and archive pottery drawings, while remaining the format of choice for Web publishing.

The three main issues we could identify testing an archive of several hundreds of drawings are:

- a lack of native support for units (i.e. expressing the vector data in real units and not in pixels) can be worked around only with complex procedures, based on the `<viewPort>` element. To date, there is no straightforward way to deal with this issue, even though it could be addressed developing one or more plug-ins for the most used software programs;
- all programs have specific ways of dealing with image layers, that are not part of the native SVG 1.1 specification, but generally this is done through the `<g>` (*group*) element. This is not a severe limitation, in that single objects can be assigned an id attribute, that is unique in each file. This possibility is discussed below in the advanced workflows section;
- except for basic elements such as `<title>`, metadata can be expressed in different ways. While Inkscape uses Dublin Core and Creative Commons vocabularies, Adobe Illustrator follows the XMP standard. Both ways are acceptable and "standard" in their own way, but this discrepancy introduces another layer of complexity, making it sub- optimal to use different programs onto the same archive.

Finally, even though it is not strictly a limitation, SVG is a file format, so the expected structure of an archive is one file per drawing, as with other image formats. Collating more than one drawing in the same file is discouraged, even though this is common when preparing illustrations for print. In theory, nothing prevents saving serialised SVG data in a database [JYY2004], to retain the relational structure often used in archaeological information systems.

2 Suggested workflow

Briefly:

- draw your potsherd using Inkscape, or another *SVG-capable software*
- give your drawing an `svg:id` attribute
- wrap the drawing in a container with `svg:transform`
- add a `svg:preserveAspectRatio` to your `svg` document

2.1 Drawing: a basic workflow

The most basic workflow is to use SVG from the start to save digitised drawings, rather than other formats available in the vector graphic software used.

Inkscape is the best solution to start digitising your drawings. Inkscape supports a large part of the SVG 1.1 standard, but it also has some extra features defined in the *Inkscape SVG* format, that is used by default when saving your work.

> **Warning:** We suggest here **not** to use *Inkscape SVG*, because for publishing it's important to use the standard SVG.

If you have legacy drawings in other formats, convert them to SVG. This is currently outside the scope of this document.

Almost all pottery drawings are not born digital. As a consequence, digitisation through tracing is the rule, not the exception. You need to import an existing drawing scanned from paper, and digitise it. This is straightforward in Inkscape. The only thing you should do is to have the scanned image on a separate *layer* from the digitisation.

Inkscape layers are just a shorthand for named SVG groups (the `svg:g` element). You can give a name to a layer, if you want, such as *original* and *digitised*.

Scale

It is good practice to give a metric scale together with your drawing. We think adding a "1:3" label in one corner of the printed drawing is a poor alternative that should be avoided: pottery drawings get copied very often both on paper and digitally. Adding a simple scale should not be very difficult.

If you do, it would be very good to use this scale also in the SVG drawing. As we will see, there are some issues with units and scale in SVG, so having a simple scale with a human-readable measure will be very useful. We could suggest to adopt a standard scale of a fixed measure, but for the moment being it will suffice to insert a clear indication, either by additional text or by using a black-and-white scale with 1 cm ticks.

Another attempt at defining scale is by scaling the drawing so that it matches the page size as if it was drawn on paper. Inkscape by default starts a blank document with an A4 page.

Human- *vs* machine-readable

A brief interlude about human- *vs* machine-readability will be helpful at this point to explain why we are paranoid about some apparently minor details.

Drawings on paper, or digitised as raster images, only have a proper meaning to us humans, in the sense that a software program could not extract any information from them, such as the size and shape of the pot (except by means of complicated procedures).

Using a vector format is a way to get machines to be able to read our hard work done by hand. If you only have one drawing, or ten, it's not that interesting, but things are different in case there are hundreds of them, and chances are most people will have *thousands*.

Plain SVG

When you're done with your drawing, Inkscape will still insist in saving it in the *Inkscape SVG* format every time. A useful workaround to this is to use Inkscape from the command line (we will be doing this a lot), like this:

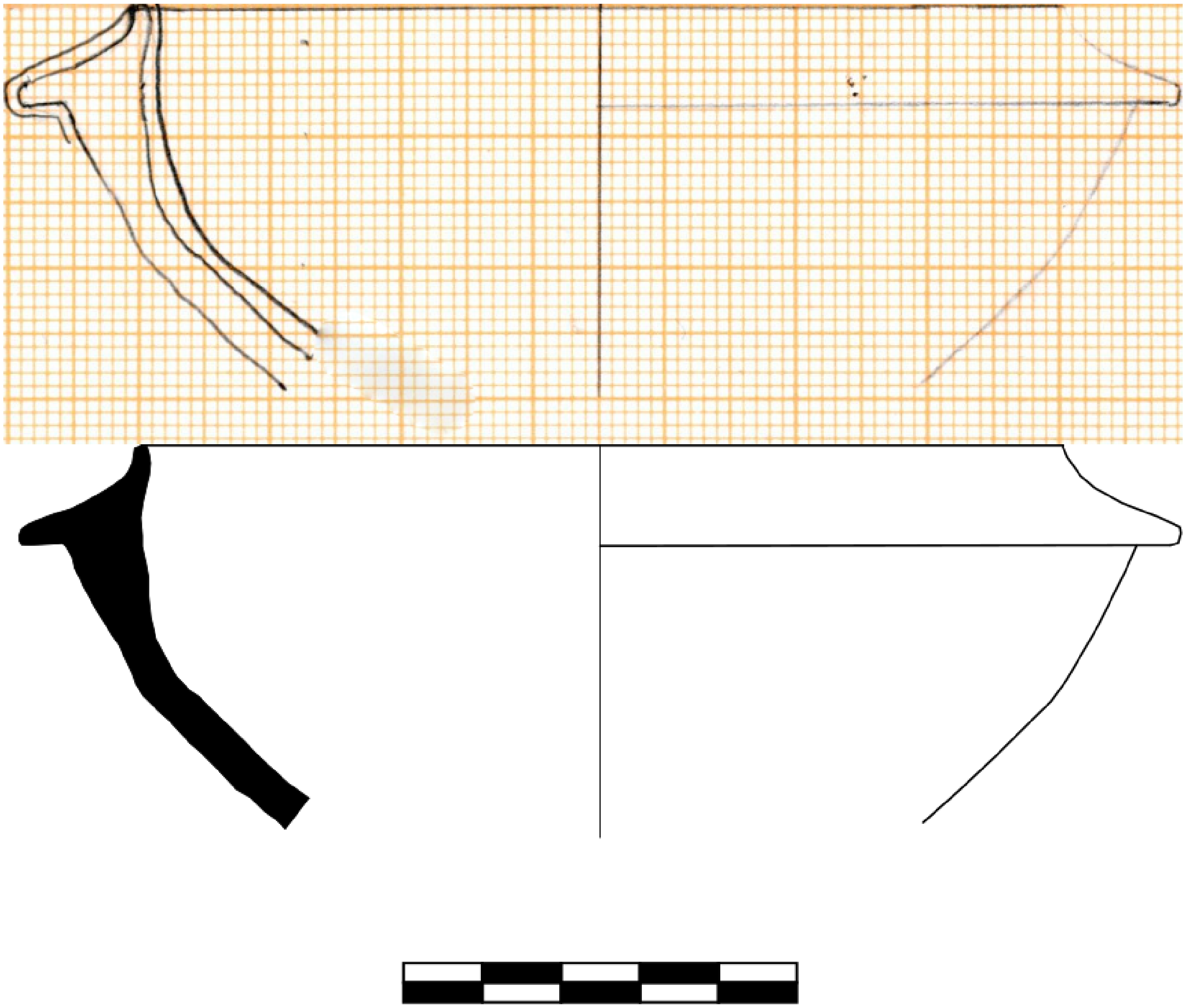
```
inkscape -l drawing-plain.svg drawing-inkscape.svg
```

Metadata

You should add metadata (author, title, description) using the “Properties” or “File properties” dialog of the drawing program. The description and title can be as verbose as needed, and preferably consistent within a collection.

In the metadata editor, mark the file as distributed under a Creative Commons Attribution license – ensuring ease of circulation and attribution for single files even when detached from the entirety of the archive.

Digitisation should be done according to the required graphical standard (e.g. section on left or right side, hatching colours, thickness of lines). An example is shown below, comparing the hand-made drawing with the SVG counterpart. Depending on the complexity of the drawing, this procedure is not time-consuming and can be performed in five minutes.

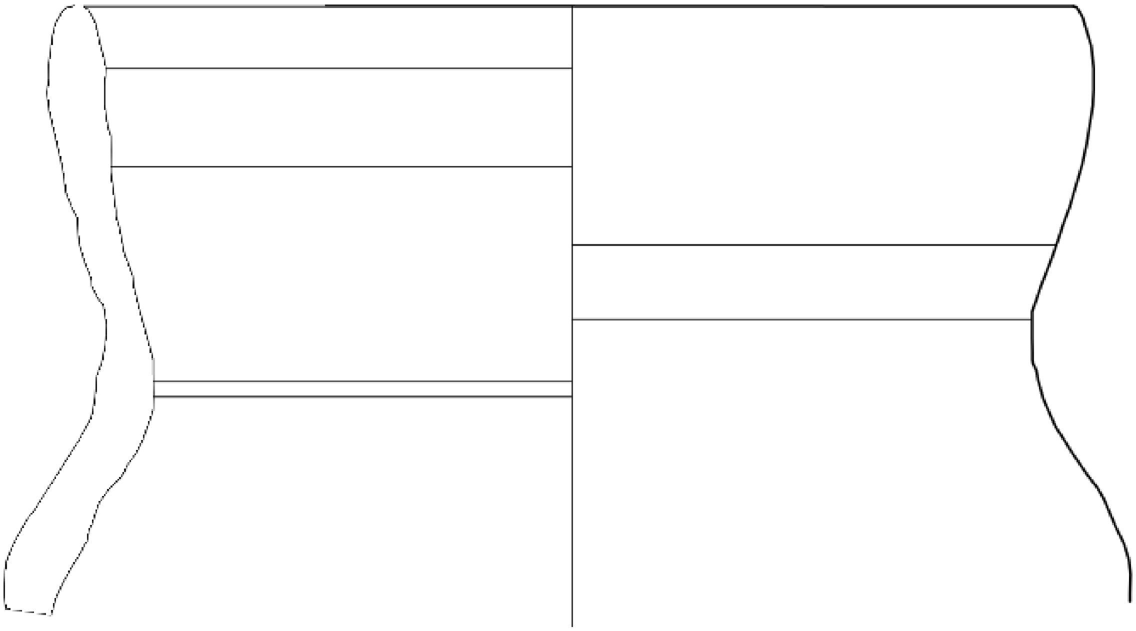


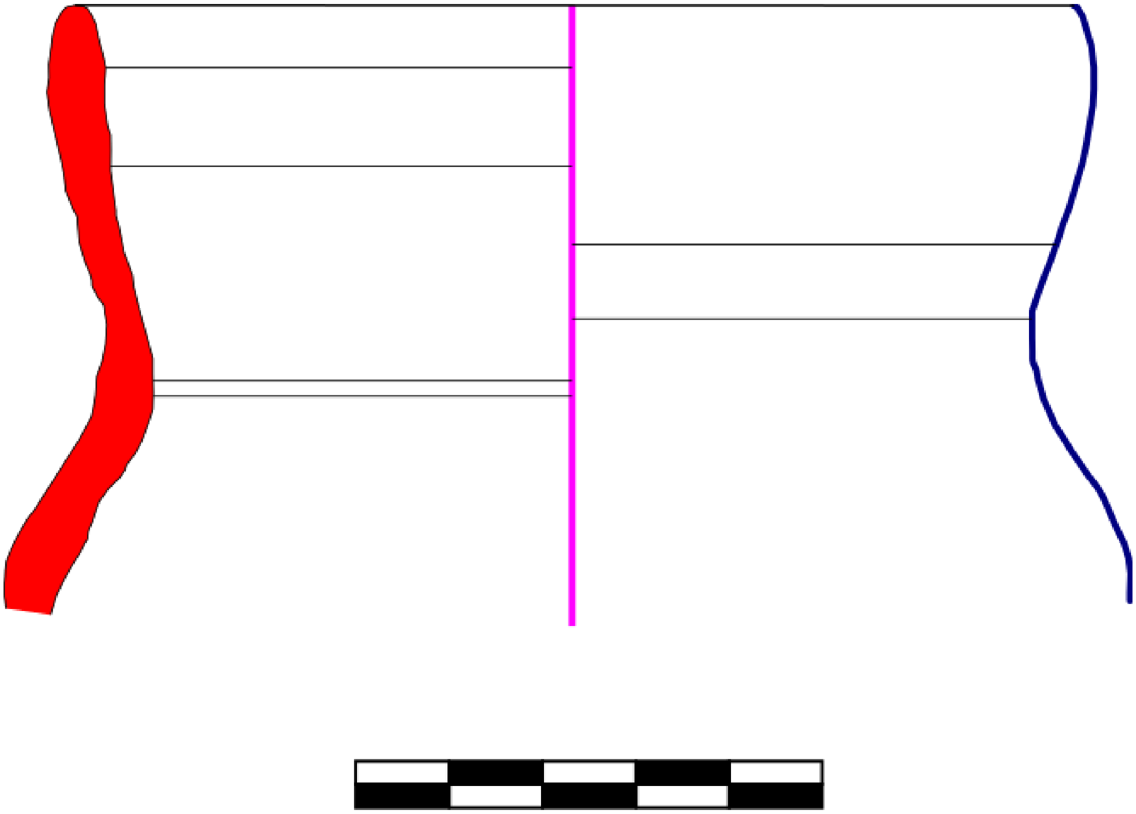
This workflow is also recommended as a reference in case of a conversion process from legacy or proprietary formats to SVG.

2.2 Advanced workflows

The basic workflow guarantees a solid base, but keeps possibilities for further processing confined to metadata. Advanced workflows make processing of geometric data attainable, but they require more digitisation time.

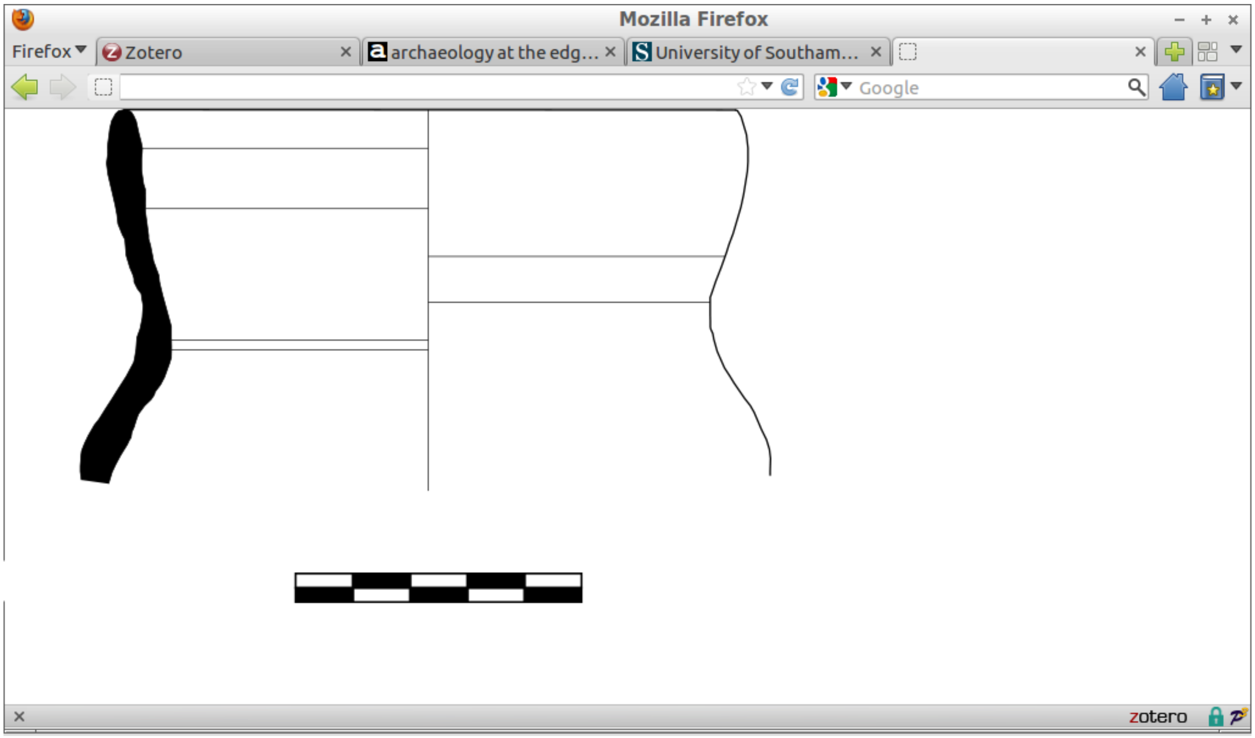
The first kind of advanced workflow is inspired by programs that calculate vessel capacity (a recent overview in [EBT2009]), and is based on a logical separation of the elements of the drawing: rotation axis, external profile, section are the three main elements to be considered, together with the metric scale. Drawing programs have a feature to give single objects an id attribute. The proposed convention is to adopt simple, unique and explicit labels like axis, profile, section, scale. An example in shown below, with different colours highlighting the separate elements. With such data available inside drawings, it is possible to automatically extract size and measurements in metric units. Comparison of vessel profiles (e.g. as described in [MTP2010]) is also technically possible and certainly not novel [SW1975].





As explained above, handling of metric units is a weakness of SVG. The <viewPort> element can be used to work around this problem, but to the best of our knowledge no software program is capable of providing an interactive management of this SVG feature. Thus, the only way to pursue this task is to manually edit the source of the drawing. For anyone who has ever written HTML, this is going to be a normal operation. XML-conforming editors should be preferred. The W3C Recommendation (Scalable Vector Graphics 1.1) contains detailed instructions on how to approach this task, but our experience showed that more than ten minutes may be required for each drawing.

The third and final step of an advanced workflow is the actual publishing of SVG drawings on the Web. While applications of any complexity can be developed for the task, it is certainly recommended to start with simple steps, taking advantage of the fact that modern web browsers can natively display SVG drawings. An HTML index with links to the SVG files makes it possible to publish drawings even on a minimal web infrastructure, as may be the case with low-cost shared hosting, only by uploading the containing directory via FTP. A basic example is shown below.



3 Issues with publishing pottery drawings

Also known as “small details you should care about”.

You probably already have some drawings. It is very easy to export them to SVG using your drawing software (at least if you’re not using AutoCAD, but still it’s not very difficult to convert DXF to SVG).

You might think: “that’s all!”. After all, you have chosen the best format for publishing vector graphics on the web, your drawings can now be both viewed and downloaded at their best quality with very small file size. And this is true.

But you can do more, if you want. So, if converting to SVG was too difficult or time-consuming for you, it’s OK. Instead, if you are starting a new catalog and you want to explore the possibilities that SVG offers, this is for you.

3.1 File size

SVG files generated by vector graphics programs tend to be quite large, and their source is filled with extra attributes that are generally useless. It’s almost like creating an HTML page with a word processor like Microsoft Word or LibreOffice Writer.

There are two ways to keep the file size small:

- use GZIP compression, that is supported by almost all viewers and editors, and generates files that have a very recognizable `.svgz` file extension;
- tidy the SVG source using a program like Scour, that will remove empty XML tags, group entities with the same attributes, remove unnecessary decimal digits from vertex coordinates and other similar tasks that would be very tedious to do by hand.

The two can also be combined, of course. but tidying is more important, because it makes the SVG source more readable, better structured and less likely to contain useless data.

4 Text editors

Any text editor can be used for SVG source files. It's plain XML, nothing else. Still, some editors are better than others.

4.1 Emacs

GNU Emacs has very good support for SVG and it allows you to switch between text mode and image mode.

4.2 Inkscape

Inkscape has a built-in XML tree viewer and editor.

4.3 Other editors

Lots of editors have XML editing capabilities, like tree-viewing and source code highlighting.

5 Software for SVG processing

5.1 Digitisation

This is a non-comprehensive list of vector graphic programs that can be used to digitise your pottery drawings, and subsequently save them to SVG.

From the point of view of *SVG Pottery*, the most important feature is the quality of the resulting SVG file, in terms of tidy source and metadata preservation.

Inkscape

Inkscape is a vector drawing software, available under the GNU GPL license, for all major operating systems.

Inkscape uses SVG as its default format, but it has some additional features compared to the standard SVG specification.

Inkscape can also edit metadata of SVG documents, like title, description, creator and licensing.

Dealing with units in Inkscape

SVG is far from ideal for handling real-world units like millimeters. Some useful advice for dealing with the limitations when using Inkscape can be found here:

- http://inkscape.13.x6.nabble.com/Units-in-Inkscape-td4971507.html
- http://wiki.inkscape.org/wiki/index.php/Units_In_Inkscape

Adobe Illustrator

Adobe Illustrator is a proprietary vector drawing software, available from Adobe in the *Creative Suite* pack.

Illustrator has its own proprietary format, but can import and export SVG natively.

If you use Illustrator for your drawings, you can easily save them to SVG.

The basic steps to follow, besides the drawing itself, are:

- add metadata (author, title, description) via the *File* → *File info* menu item
- *File* → *Save as...* and choose SVG as format
- the SVG Options dialog is very important:
  - be sure to enable the *Include XMP* option, that will include metadata in the SVG file
  - use the *Show SVG Code* option to have a preview of the generated SVG
  - stick to SVG 1.1 (default)

Other programs

Other vector graphics programs have poor or no support for SVG:

- Xara Xtreme: has no support for SVG
- Corel Draw: *unknown*

5.2 Editing and automation

The following tools are for automated editing and processing of SVG files.

Their use is suited for advanced users.

Batik

Batik is a complete SVG implementation for Java developers. It also has a few small tools that could be useful when post-processing your drawings, like:

- `rasterizer`, an SVG-to-bitmap converter
- `svgpp`, a pretty printer for SVG source code
- `squiggle`, a visualization program

Ink Generator

ink-generator is an Inkscape extension to substitute text and data in automatically-generated files, starting from an SVG model and a CSV data file.

Scour

Scour is a Python module that aggressively cleans SVG files, removing a lot of unnecessary information that certain tools or authors embed into their documents. The goal of scour is to provide an identically rendered image (i.e. a scoured document should have no discernable visible differences from the original file) while minimizing the file size.

Warning: Scour is intended to be run on files that have been edited in Vector Graphics editors such as Inkscape or Adobe Illustrator. Scour attempts to optimize the file, and as result, it will change the file's structure and (possibly) its semantics. If you have hand-edited your SVG files, you will probably not be happy with the output of Scour.

Never use scour to overwrite your original file!

xmldiff

xmldiff is a Python tool that shows the differences between two similar XML files.

6 Other use cases of SVG in archaeology

SVG is already being used in a few archaeological environments, and there are some cases that are worth mentioning here:

- Oxford Archaeology guide to plotting site plans using QGIS and then Inkscape for the advanced graphical editing of the layout [OAL366] – even though the focus is not on SVG as a format but on vector graphics;
- an interactive image viewer on the Web was developed for the Sikyon Survey Project using Scalable Vector Graphics (SVG) and Ajax [CHARNO2007];
- SVG as a potential tool for archaeologists is discussed in [WRIGHT2006], including some of the ways vector graphics are used in archaeology, and outlines the development and features of SVG, which are then demonstrated in the form of a case study using large-scale plan and section drawings converted to SVG from AutoCAD;
- a combined approach using SVG adn X3D to record and visualise archaeological textiles is presented in [PAGI2010], offering a detailed overview of how SVG is used.

Most of these cases share a common focus on site plans, or spatial data in general. The tendency of spatial archaeology to be very advanced in the use of information technology, and the wide gap that separes other subdomains in this respect, is beyond the scope of this document.

There is indeed one example of SVG for pottery drawings, in the excellent "Greek, Roman and Byzantine Pottery at Ilion" web publication ([GRBPILION]). You can not only browse the extensive catalogue of pottery finds from the excavation, but also download the entire site as a compressed archive. Inside the compressed archive the JPEG images of pottery drawings come along with their SVG originals. Actually, GRBP is one of the inspirations for this research. Their approach is purely graphical, meaning that there is no care for the SVG source and its content, as long as it renders to a proper image. There is no grouping of graphic entities based on their meaning within the drawing. The scale is given by a graphical 5 cm scale, that is included in every file.

As suggested in the quickstart of this documentation, this is a very good example for those who don't want to go into the details of SVG and just care about using a standard, open format for publishing their work.

7 Glossary

axis of simmetry if present, the axis of simmetry is usually taken not only to represent the axis of rotation of the pot, but also often to split the drawing in two parts: on one side, a section is provided, as if the pot was cut along the radius; one the opposite side, a profile from outside is shown. There is no accepted standard of which side should be which.

DOM the Document Object Model is a technical way to describe the "tree" of nested XML tags that compose all documents, including SVG. If a software program is able to manipulate the DOM, this means it could do something useful for you, like filtering out certain parts of the drawing based on attributes.

drawing a drawing is a formal representation of a pot(sherd), intented for the purpose of publishing the exact shape with an optimal use of space on printed paper.

8 Indices and tables

- genindex
- search

References

[COSTA2014] Costa, Stefano. 2014. «SVG Pottery: Upgrading Pottery Publications to the Web Age». In Archaeology in the Digital Era: Papers from the 40th Annual Conference of Computer Applications and Quantitative Methods in Archaeology (CAA), Southampton, 26-29 March 2012, edited by Philip Verhagen, 533–40. Amsterdam University Press. http://dare.uva.nl/aup/en/record/500958

[JYY2004] W. Jianting, L. Yan and J. Youguang 2004 "A Design and Implementation of Spatial Database Based on XML-SVG", in SVG Open 2004. 3rd Annual Conference on Scalable Vector Graphics. http://www.svgopen.org/2004/papers/ADesignandImplementationofSpatialDatabasebasedonxmlsvg/

[EBT2009] L. Engels, L. Bavay and A. Tsingarida 2009, "Calculating vessel capacities: a new web-based solution", in A. Tsingarida ed. Shapes and Uses of Greek Vases (7th– 4th centuries B.C.), Brussels, Timperman, pp. 129-133.

[MTP2010] I. Modrzewska, G. Taroni and F. Pianetti 2010, "Un'anfora frammentaria dalla Laguna di Venezia", Archeologia e Calcolatori, 21, pp. 201-210.

[SW1975] S.J. Shennan and J.D. Wilcock, 1975 "Shape and style variation in Central German Bell Beakers: a computer-assisted study", Science and Archaeology 15, pp. 17–31.

[OAL366] C. Robinson, D. Campbell and A. Hodgkinson, 2010 "Archaeological maps from qGIS and Inkscape: A brief guide". Documentation. Oxford Archaeology North. (Unpublished) http://library.thehumanjourney.net/366/

[CHARNO2007] M. Charno 2007, "An Interactive Image Using SVG and Ajax in Archaeology", *Internet Archaeology*, 23 http://intarch.ac.uk/journal/issue23/charno_index.html

[WRIGHT2006] H. Wright 2006, "Archaeological Vector Graphics and SVG: A case study from Cricklade", *Internet Archaeology*, 20 http://intarch.ac.uk/journal/issue20/wright_index.html

[GRBPILION] S. Heath and B. Tekkök (eds.), 2006-2009, "Greek, Roman and Byzantine Pottery at Ilion (Troia)". Retrieved 2011-06-02 from http://classics.uc.edu/troy/grbpottery/

[PAGI2010] H. Pagi, 2010, “When data becomes information: visualizing archaeological textiles” in *Computer Applications and Quantitative Methods in Archaeology, Williamsburg, US*, Oxford, GB, Archaeopress, 285-291. http://eprints.soton.ac.uk/161699/

Index