DocScapes: Visualizing Document Structures with SVG

Hugh Cayless *(hcayless@lulu.com)*
Lulu (<http://lulu.com>)

The task of searching for and browsing documents online can be a frustrating one. Documents in search results are typically treated as atomic units rather than structured collections of information. This paper proposes some ideas for enhancing search and browsing by producing graphical 'document-scapes' that summarize document characteristics and provide links into the content of documents. The advantage of this type of summary is that it can compensate for some of the visual cues (available when browsing bookshelves) that are lost in the digital environment. It is possible to visually summarize document size, structure, density, and the presence of metadata in such a way that users will be able to tell, at a glance, the difference between (for example) an interview and a monograph, or a play and a catalog. The work in this paper focuses on a particular vocabulary of document markup, TEI, and a particular collection, *Documenting the American South* at the University of North Carolina at Chapel Hill (<http://docsouth.unc.edu>.

A great deal of work has been done on the visualization of collections and search results (see <http://www.cs.umd.edu/hcil/research/visualization.shtml> for a summary of online material). There is, however, a remarkable paucity of scholarship focusing on the visualization of documents themselves. No doubt this has to do with the difficulties of dealing with heterogeneous collections. Comparing the varying structures of text, XML, and PDF documents, for example, might not be an especially useful exercise. The technique discussed in this paper can easily be applied to relatively homogeneous collections of XML documents, however, and could in theory be generalized to other document types.

The techniques used in this project are relatively simple. Essentially, what is involved is the transformation of XML from one vocabulary to another; in this case TEI to SVG. Scalable Vector Graphics is an XML application that allows for the representation of vector graphics in an XML format. This means that the structure of a document in, for example, TEI, can be turned into an image via the same processes used to display the document in HTML or to covert it to PDF for printing. Since other document formats can be parsed to generate SAX (Simple API for XML) events, they too could be fed into an XML processing pipeline and turned into *DocScape* images.

There are a number of variables which may be used to distinguish documents marked up in TEI without recourse to semantic distinctions like subject vocabularies. Since TEI documents are subdivided by division (`<div>`, `<divN>`, `<front>`, `<back>`, etc.), each document has its own internal structure. Different types of document may have very different internal structures. For example, a dictionary will consist of a set of entries (`<entry>` tags) inside its divisions while a monograph will contain chapters, sections, and paragraphs (`<p>`). The relative size and structure of nested divisions can be represented graphically in a fairly compact space. Differing types of content, on the other hand, can be represented using color.

TEI documents also differ in size (obviously) and this can be an important metric. Size can be represented visually in a number of ways. *DocSouth*'s collection varies widely in terms of absolute size, from short pamphlets to large books and government documents (up to 800 pages in length). The representation of relative size must therefore be considered quite carefully. The first iteration of *DocScapes* did this using border thickness. A pixel was added to the border width for each 100 pages. This sort of scale does not help in handling the important distinction between the moderately sized (10-50 page) document, and the very short (1-2 pages), a distinction which encompasses important differences of genre. The next generation of *DocScapes* will use more complex SVG capabilities, such as drop shadows to indicate relative size.

Another important metric is the relative size and complexity of the TEI Header metadata. *DocSouth*, whose documents are largely derived from catalogued library holdings, has very detailed and thorough header information. By contrast, a TEI document that was 'born digital' might have fairly minimal metadata. A visual distinction of different levels of metadata density will be useful for collection managers and searchers alike.

A *DocScape* image is composed of the elements outlined above: the document itself, any header metadata and structural container elements (e.g. `<div>`s in TEI, `<section>`s in *DocBook*, etc). The four TEI Header sections are represented by blocks of color at the top of the image. The nested divisions are visualized as nested blocks, moving first left-to-right then top-to-bottom, and so on. The nested blocks start from different ends of the light/dark scale, so top-level containers are light green, then their children are dark green, etc. In addition, the image attempts to quantify the number of paragraphs per page or section using color saturation. The relative size of the document is indicated by the border thickness of the entire image (see figures 1 and 2).

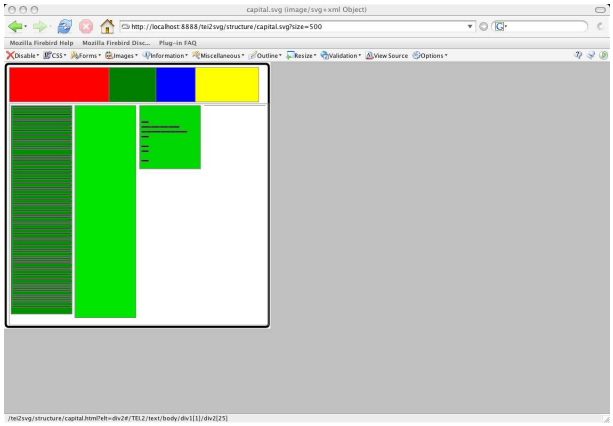

Figure 1

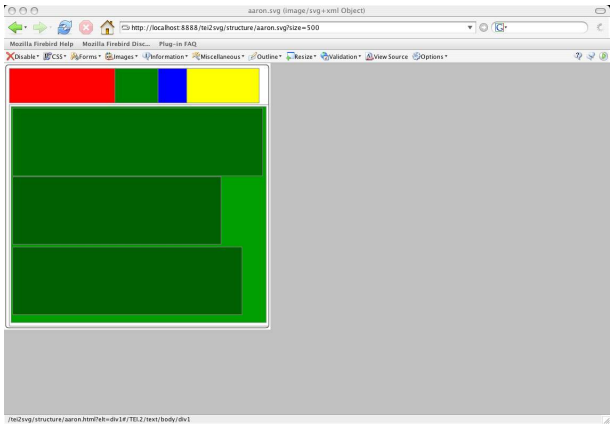

Figure 2

Figure 1 provides a nice example of a document with a very heterogeneous internal structure. The first section is a catalog, with many nested TEI `<div>`s, while the following divisions are more narrative in nature. Figure 2, on the other hand, represents an interview. The more densely packed paragraph structure in this document is represented by the lighter shade of green in the nested sections.

In addition to these basic elements, it is possible to use the capabilities of SVG to group many documents on a single page and dynamically zoom into the ones that are of interest. The document sections may also be linked to the documents themselves, so that it is possible to drill into the texts from their visual representations. Finally, it is possible to layer other information, such as the occurrence of search terms onto the documents. Figure 3 is an example of a *DocScape* with personal names, locations, and dates plotted on the image surface. My paper will outline the techniques and principles involved in developing DocScape visualizations and will discuss ways in which they may be used in digital libraries as a means to browse textual content.

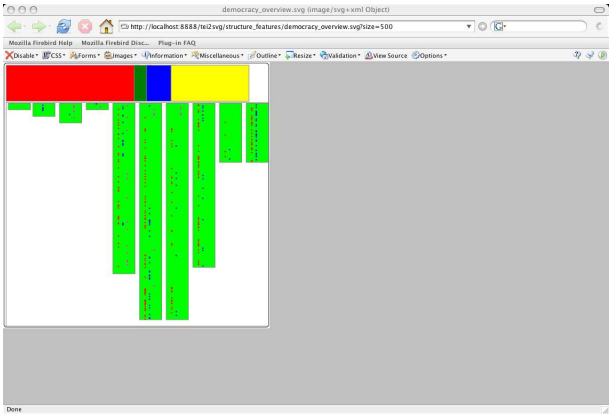

Figure 3

Bibliography

Börner, K. "Extracting and Visualizing Semantic Structures in Retrieval Results for Browsing." *Proceedings of the fifth ACM Conference on Digital Libraries.* 2000. 234-235.

Campeseto, O. *Fundamentals of SVG Programming: Concepts to Source Code.* Hingham, MA: Charles River Media, Inc., 2003.

Clark, James. *Transformations (XSLT), Version 1.0 (W3C Recommendation).* W3C, 1999. Accessed 2005-03-15. `<http://www.w3.org/TR/1999/REC-xslt-19991116>`

Clark, James, and Steve DeRose. *XML Path Language (XPath), Version 1.0 (W3C Recommendation).* W3C, 1999. Accessed 2005-03-15. `<http://www.w3.org/TR/1999/REC-xpath-19991116>`

Clark, James, Jun Fujisawa, and Dean Jackson. *Scalable Vector Graphics (SVG) 1.1 Specification (W3C Recommendation).* W3C, 2003. Accessed 2005-03-15. `<http://www.w3.org/TR/2003/REC-SVG11-20030114>`

Documenting the American South. University of North Carolina at Chapel Hill. Accessed 2005-03-15. `<http://docsouth.unc.edu>`

Hornbæk, K., and Erik Frøkjær. "Reading Patterns and Usability in Visualizations of Electronic Documents." *ACM Transactions on Computer-Human Interaction (TOCHI)* 10.2 (2003): 119-149.

Sperberg-McQueen, C.M., and L. Burnard, eds. *TEI P4: Guidelines for Electronic Text Encoding and Interchange.* Text Encoding Initiative Consortium, 2002.

Venn, B. *Add Interactivity to Your SVG.* IBM developerWorks, 11 December 2003. Accessed 2005-03-15. `<http://www-1`

06.ibm.com/developerworks/web/library/x-svgint/>

Visualization. Human-Computer Interaction Lab / University of Maryland. Accessed 2005-03-15. <http://www.cs.umd.edu/hcil/research/visualization.shtml>