Annotations in the wild

Laurent Denoue[1] and **Laurence Vignollet**[2]

Abstract. We believe that storing web annotations on annotation servers limits the widespread adoption of web annotation technology. Instead of relying on annotation servers, we propose to encode annotations as an extended URL. Because they follow the standard URL encoding, these extended URLs can be readily used and embedded in Web documents. We then envision specific search engines that would be able to index these extended URLs and provide interesting new services.

1 INTRODUCTION

On the Web, you can create a link using a standardized format (i.e. URL) and embed it into your documents without relying on an external server. You can publish your links when you wish so by simply making the document publicly available. Others can then leverage the links you have created, like building better web search engines (e.g. Google).

But what would be the Web if every link you created had to be stored on a link server? What would it be if everyone used a different scheme to encode a link? What would it be if you could not embed a link into your documents? Sadly, today's annotation systems are implemented with this model:

- Users rely on a remote annotation server to create the annotation,
- Once created, the annotation is not given a unique URL: users cannot easily link to it from their documents,
- Every annotation system uses its own format and set of APIs to retrieve annotations, making it hard for third party applications to index and reuse these annotations.

In this paper, we argue for the same bottom-up approach that made the web so popular: the possibility for anyone to link to any document from any document:

- Users don't need a remote service to create hyperlinks: hyperlinks are self-contained,
- Hyperlinks can be embedded into documents,
- Hyperlinks are standardized.

Here, we propose to extend hyperlinks so that they encode the URL of the annotated page and the annotation made on this page.

2 SELF-CONTAINED ANNOTATIONS USING EXTENDED URLS

An annotation can be described as a couple (anchor, description). The anchor encodes the attachment point of the annotation (e.g. the whole document or part of a document); the description is what the users attaches to this anchor (e.g. a textual comment). The description is optional. For example, the anchor is sufficient if users want to highlight a part of a document without adding a comment.

We propose to use the existing URL format to encode an annotation, and call it an extended URL. In the current URL format, it is possible to point to a specific part of a document by specifying a tag name after the # sign in the URL. These tag names are extracted by the Web browser and used to locate the similar name in the current Web page. We propose to use the same mechanism to allow anybody point to a part of a document.

In our scheme, the simplest extended URL looks like:

http://www.cnn.com#anchor=<encoding>

where http://www.cnn.com is the document being annotated and <encoding> is a string allowing the browser to find a specific part in this document. We will describe the <encoding> part later. Note that this scheme is compatible with the popular GET method where parameters are passed after the ? sign. The # sign is always appended after ?:

http://www.cnn.com?id=12032002#anchor=<encoding>

The description of the annotation can also be included in the extended URL. For example:

http://www.cnn.com#anchor=<encoding>&description=my%20comment

will annotate the part of the document http://www.cnn.com identified by <encoding> with a description containing "my comment". The description can also include a URL, allowing users to annotate a document with any object identifiable with a URL (e.g. picture, video, program).

3 ENCODING PARTS OF A DOCUMENT IN THE EXTENDED URL

After having identified a specific resource with a URL, the anchor also needs to identify a part of the document pointed to by the URL. Because the documents being annotated can change, the encoding should allow a user or an application to:

- Detect that the anchor point is not valid anymore
- Attach the annotation on the new version of the document

XPointers [1] have been proposed to link to sub-parts of a document. But because they mainly rely on the structure of the document, we think that XPointers have two limitations:

- It is hard to attach the annotation when the structure of the document changes
- The content pointed to by the XPointer is not human-readable

Following research from David Bargeron [2] and Ping Lee [3], we suggest that the encoding uses a more "human-level" encoding. For example, the encoding of a highlight

[1] FXPAL, Palo Alto, CA USA

[2] Syscom, University of Savoie, France

in a document could be the simple string being highlighted. Because the string is not always unique, the encoding can also keep the rank of the string in the whole document. Thomas Phelps and Robert Wilensky at UC Berkeley have specifically designed robust encoding strategies using the content of a document [4]. Although we envision a "human-level" encoding for anchoring annotations, we also acknowledge that some applications will need to specifically anchor annotations to the structure of the document. Xpointer-like encodings can then be used.

4 IMPLEMENTING EXTENDED URLS

We have extended Yawas [5] to let users create extended URLs. Currently, Yawas only supports text highlighting. From a Web browser, the user selects a textual content and chooses "Highlight" from a context-menu. Yawas creates an extended URL for this highlight and automatically copies it into the system's clipboard. This approach allows users to quickly embed their annotations into emails or other documents. Users can also create many annotations on one document and export all of them. The encoding is quite simple for our prototype. From the highlighted text, we keep the 10 and last 10 characters, followed by the original size of the highlighted text, followed by the rank of this string in the whole document.

Users having Yawas installed on their machine can access these annotations. Because there is no search engine available for retrieving annotations attached to a web page, users currently list URLs of documents where Yawas should look in for annotations. Users can also receive an annotation by email in the form of an extended URL. When they click on the link, the web browser opens the page. Current web browsers silently ignore the parameters after the # sign. In the current implementation of Yawas, users choose "Import" from the context-menu in Internet Explorer and paste the extended URL. Yawas then parses the string and highlights corresponding parts of the document. We are working on a new version where Yawas-enabled browsers will be able to understand these extra parameters and highlight the corresponding passage in the document without the user having to manually paste the extended URL.

Users can also create more than one annotation on a web page A, store them in a web page B and send a link to page B by email. The recipient of the email just clicks on the link and the browser loads page B. When page B is loaded, Yawas checks for all extended URLs embedded in this page and stores them in a local annotation file. Using the REFRESH meta tag in HTML, page B can automatically redirect the recipient to page A that was originally annotated. Because Yawas already parsed the embedded URLs in page B and extracted its extended URLs, highlights are then automatically displayed in page A.

5 CONCLUSION AND FUTURE WORK

Our implementation tries to demonstrate the flexibility of using extended URLs to embed annotations in any document supporting hyperlinks (e.g. HTML and Word documents).

Although the web encourages sharing, we should keep in mind that annotations should not only be sharable using a web server. We believe that forcing users to do so limits them from adopting web annotation technologies. A typical limitation is privacy: annotating a document is a personal activity, and not all users would be willing to share all annotations they create. Unfortunately, most annotation systems designed so far promoted the sharing of annotations, forgetting why people annotate in the first place.

Bookmarking is a particular form of annotation. People accumulate personal bookmarks over time. After a while, they might consider a subpart worth sharing with others and decide to publish a Web page containing their best bookmarks.

We hope that designers of annotation systems will externalize their annotations in a form of extended URLs. Automatic 'agents' could then crawl the web for retrieving these extended URLs and start building an annotation search engine. Many services could then be implemented. For example, we showed in a previous work how to improve Web page classification by using annotations (versus the full text) [5]. It would be very interesting to extend this research by using annotations created by many users. More generally, annotations give us an understanding of not only what document users like, but what in the document they like.

Here, we proposed a simple solution for encoding the anchor point of an annotation based on the content as opposed to the structure. Obviously, different techniques can be used to perform this encoding. Instead of having to choose one over the other, one solution would be to publish the decoding algorithm for each one so the research community can investigate different encoding schemes without preventing annotation technology to be deployed.

ACKNOWLEDGMENTS

We would like to thank all members of the w3c-annotation list who gave us interesting ideas and feedback on the design of annotation systems.

REFERENCES

[1] World Wide Web Consortium, 'XML Pointer Language (XPointer) Version 1.0' (2001)
[2] D. Bargeron, 'Robust Annotation Positioning in Digital Documents', Proceedings of CHI 2001, 285-292 (2001)
[3] L.P. Lee, 'CritLink: Better Hyperlinks for the WWW', Proceedings of Hypertext 98 (1998).
[4] T. Phelps and R. Wilensky, 'Robust Hyperlinks Cost Just Five Words Each', UC Berkeley Computer Science Technical Report UCB//CSD-00-1091, Berkeley,USA (2000).
[5] L. Denoue and L. Vignollet, 'Personal Information Organization using Web Annotations', Proceedings of Webnet 2001, Orlando, Florida (2001).