
Annotonia: Annotations from Browser to TEI

Greg Tunink

techgique+conftool@unl.edu
University of Nebraska-Lincoln Libraries
United States of America

Karin Dalziel

kdalziel2@unl.edu
University of Nebraska-Lincoln Libraries
United States of America

Jessica Dussault

jdussault@unl.edu
University of Nebraska-Lincoln Libraries
United States of America

Emily J. Rau

erau2@unl.edu
University of Nebraska-Lincoln Libraries
United States of America

Introduction

The *Willa Cather Archive (WCA)* at the University of Nebraska-Lincoln (UNL) is currently working on transcription and annotation of 1500 letters to be released in 2018. As editors will write several thousand annotations, the workflow logistics are complicated. Annotonia (a portmanteau of "annotation" and *My Ántonia*, a 1918 Willa Cather novel) is a solution developed within the Center for Digital Research in Humanities (CDRH) that allows editors to write annotations directly on letters in a browser and insert those annotations into Text Encoding Initiative (TEI) XML files. Multiple editors review annotations, track letters' annotation statuses, and generate a new TEI file incorporating the annotations, avoiding having to manually edit each file. Annotonia utilizes both pre-existing, customized open source software and new software developed for this project. This paper describes the difficulties faced, the workflow of Annotonia, and its prospects for future annotation work.

The challenge

The *Complete Letters of Willa Cather* is a National Endowment for the Humanities-funded project to publish a digital, fully annotated edition of the letters of Willa Cather, a 20th century American novelist. The project includes student editorial assistants, staff, and faculty both at UNL and working remotely. Because of differences in locale, technical abilities, and Cather-related expertise, a solution was sought for drafting, revising, and encoding letter-specific annotations that would fit the skillsets of all collaborators.

The *WCA* has two types of annotations: authority files and letter-specific information. Editors skilled with TEI XML curate and encode people, places, and works shared across the corpus into authority files. Letter-specific annotations are more challenging to manage as they are written by a wider group of scholars, many unfamiliar with encoding XML. Editors previously used a cumbersome process of pasting documents into Word files, sharing them for annotation, and laboriously copying received annotations back into the XML. This tedious process introduced errors and was deemed unsustainable for the large number of anticipated letter-specific annotations.

With these difficulties in mind, *WCA* editors collaborated with CDRH developers to envision tools that would allow viewing the content of letters with all associated materials; writing letter-specific annotations that might include images, links, or other materials; and exporting finalized versions of these annotations as TEI XML. Editors also identified the need to view annotations that had already been written so information contained in them would not be repeated.

Project requirements

- Display letters via HTML with existing controlled vocabulary annotations for people, places, and works.
- Provide an interface where editors can browse, edit, approve, and flag annotations.
- Export annotations to their preservation format, TEI P5 XML.
- Insert <ref> tags into the TEI XML corresponding to ID's in the annotation file above.

Existing software review

Before building Annotonia, we looked at existing technologies and methods for annotating HTML. Web-based annotation is not a new concept, by any means. The value of annotating HTML has long been recognized for its collaborative strengths, allowing users to identify errors, review, comment on, and bookmark documents (W3C Digital Publishing Interest Group,

2014). The W3C currently has a Web Annotation Working Group dedicated to creating specifications for "interoperable, sharable, distributed Web annotation architecture" (W3C Web Annotation Working Group, 2017; Web Annotation Data Model, 2017).

Most annotation software reviewed was not suitable for the project. Many existing solutions (such as [Annotation Studio](#), [Recogito2](#), and [Editor's Notes](#)) did not allow in place annotation, instead requiring one to upload documents to their system. This would strip out the existing annotations, making work more difficult for editors. Some (such as [PundIT](#)) were not fully open source. Others (Such as [Hypothes.is](#)) were developed for public input and would require stripping out many features for our needs. Other problems encountered included installation roadblocks, lack of maintenance, and poor documentation. Importantly, our requirement of exporting annotations and embedding them into TEI documents was not supported by any existing systems.

With these considerations, programmers chose to work with Annotator and its back end complement, Annotator Store, because the software interacts through an API and uses XPath's for pinpointing annotations' locations (Open Knowledge Foundation, 2016a; 2016b). Annotator proved to be easy to install and extend with community plugins, such as rich text editing, keyword tagging, and filtering. Annotator Store's use of Elasticsearch reinforced the decision to use it, as Elasticsearch is widely used and receives generous community development and support. Annotator deals only with the annotation part of the requirement and not categorizing or workflow, which had to be developed in-house.

Annotonia: the solution

The first step was simply to display the letters' TEI XML in a browser in a way that minimally rearranged the structure of the documents. TEI Boilerplate uses browser XSLT capabilities to create an HTML/TEI hybrid representation of TEI documents with small alterations for links, images, and other elements (Walsh et al. 2013). Boilerplate was therefore useful for constructing HTML from documents, while allowing for TEI files to be added and removed quickly from the site structure and workflow.

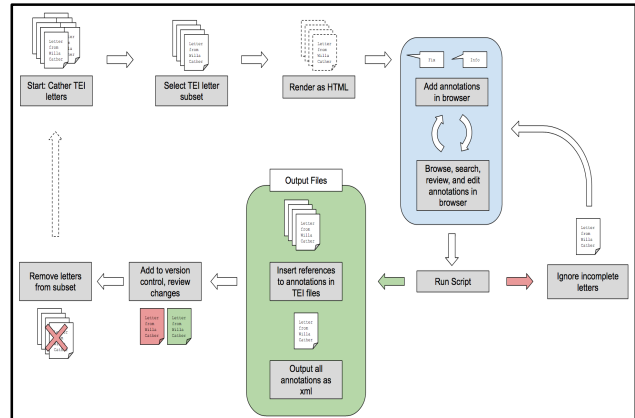


Figure 1: The Annotonia workflow for a single batch of letters

In order to be able to annotate the documents displayed using TEI Boilerplate, programmers embedded Annotator in the web page and modified it so it had suitable options for the WCA editors. These included stripping out some of the text editing capabilities and adding workflow-specific options.

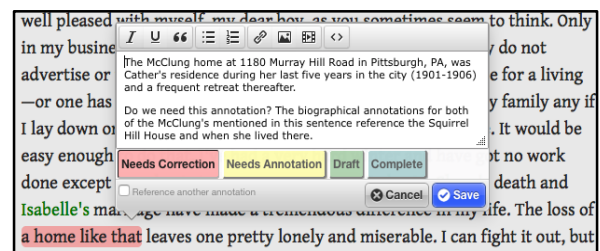


Figure 2: An example of creating and categorizing an annotation with Annotonia

Annotator does not include an interface for searching, browsing, and editing all of the annotations from the Annotator Store. PHP pages were written to provide these capabilities.

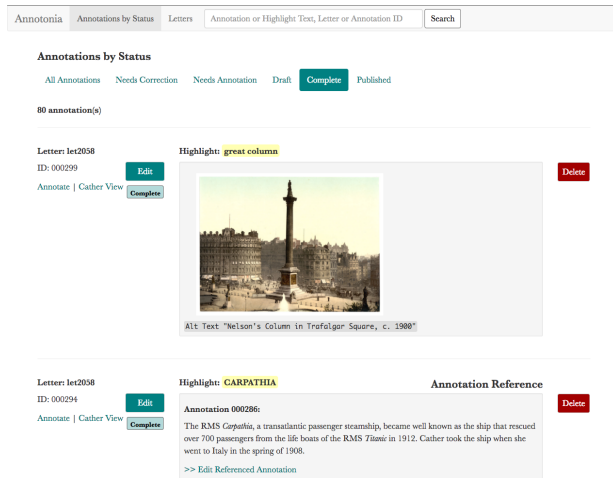


Figure 3: Annotation review page including an annotation with an image and a referenced annotation

The last requirement for Annotonia was a conversion script that inserts annotation references into TEI documents while exporting an authority file containing the annotations. The conversion script takes a subset of XML files, queries the Annotator Store API, adds tags for annotations based on XPaths, and outputs a list of possibly incorrect insertions that require review.

Use and extension

The collection of new scripts and open source software comprising Annotonia has been able to handle the workflow requirements of the *WCA*'s letter annotations, though there is room for improvement. The areas that require the most attention are the rendering of TEI in HTML and the scripted editing of TEI files. TEI Boilerplate necessarily alters the TEI in order to mimic the behavior of HTML links and images, which means that occasionally the location of an annotation in the HTML view is difficult to match up programmatically due to differing XPaths.

A new alternative to Boilerplate, CETELcean, which promises to preserve "the full structure and information from your TEI data model," may be one possibility to address this problem (Cayless and Viglianti, 2016). Programmers would need to evaluate how easily it can be incorporated into the Annotonia workflow, as well as the ability to load the Annotator JavaScript libraries in the page.

Some aspects of the Annotonia code base are tailored for use only with TEI Boilerplate display and *WCA* file naming conventions. These functions would have to be generalized to make Annotonia easier to integrate with other tools and projects. Meanwhile, the

Annotator community continues to improve the software. A new version is forthcoming which includes modifications to enhance the experience of creating, saving, and updating the highlighting of HTML content. When this version of Annotator is released, it may require some reworking of existing customizations, but updating Annotonia to incorporate its new features will support a broader variety of projects.

Reception

The *WCA* began using Annotonia in September 2016. Guidelines and instructional videos demonstrating Annotonia's functions have alleviated difficulties from the varying technical skills of editors. Although still in the beginning stages of annotating individual letters, the tool works well for collaborating on, drafting, and revising annotations. Editors have estimated that each annotation automatically handled saves around 5 minutes of time, so the potential time savings is several months of work. Through an iterative process, improvements have been added as users uncover inefficiencies and provide feedback. By 2018, several thousand annotations will be created with Annotonia and published with the complete letters on the *WCA*.

Further applications

A frequent difficulty for digital archival projects is efficiently proofreading and annotating texts. In the CDRH, these workflows on other projects resemble the *WCA*'s old process of marking up Word documents, or, worse yet, printing out entire websites and annotating by hand. Designing solutions by combining open source software with well documented, configurable scripts and workflows has proven to be effective in providing flexibility to cover a variety of needs. As we apply this method to extending and generalizing Annotonia for other projects like the *Walt Whitman Archive* and *The William F. Cody Archive*, we will further refine deployment and documentation.

The PHP and conversion components of Annotonia have been published ("[Annotonia Status](#)" and "[Annotonia Converter](#)"), and the customized pieces of existing software will be published soon. Full publication of Annotonia will involve further documentation and testing. It is the hope of the Annotonia team that this tool will not only prove to be useful internally, but will provide inspiration to other text based editions seeking to automate annotation processes.

Bibliography

Cayless, H. A. and Viglianti, R. (2016). "TEIC/CETELcean." *GitHub*. <https://github.com/TEIC/CETELcean> (accessed 19 October 2016).

Open Knowledge Foundation (2016a). "Annotator - Annotating the Web." <http://annotatorjs.org/> (accessed 11 October 2016).

Open Knowledge Foundation (2016b). "Annotator-Store." *GitHub*. <https://github.com/openannotation/annotator-store> (accessed 11 October 2016).

W3C Digital Publishing Interest Group and Open Annotation Community Group (2014). "Annotation Use Cases." <http://www.openannotation.org/usecases.html> (accessed March 3, 2014).

W3C Web Annotation Working Group (2017). "Web Annotation Data Model." <https://www.w3.org/TR/annotation-model/> (accessed February 23, 2017).

W3C Web Annotation Working Group (2016). "W3C Web Annotation Working Group." <https://www.w3.org/annotation/> (accessed October 11, 2016).

Walsh, J., Simpson, G., and Moaddeli, S. (2016) "TEI Boilerplate." <http://dcl.ils.indiana.edu/teibp/> (accessed 19 October 2016).

Willa Cather Archive. "Github Organization." *GitHub*. <https://github.com/Willa-Cather-Archive> (accessed 19 October 2016).