

Revisions: A Digital Humanities Interface Experiment

Introduction

An interface is the means through which two systems interact. It is a common boundary that brings into dialogue multiple distinct worlds. In the context of the digital humanities, the computer interface is the *space* where physical texts meet digital methods and modes of representation. It is where humanistic inquiries confront computational methods. The computer interface is where readers experience what Hayles (2003) has described as both the translation *and* interpretation of a text into digital form (p. 263). By both enabling and structuring particular kinds of reading practices the computer interface can be imagined as the stage for what Drucker (2009) calls the performative production of a text (p. 170). Given the multiple and complex roles that a digital interface can play, it should come as no surprise that Compton and Siemens (2013) have called interface design “...one of the most pressing issues in the future of the scholarly edition” (p. 2).

This paper takes seriously the import accorded to the interface within the digital humanities. It will probe some of the possibilities and limits of the computer interface as a reading and research tool by unpacking theoretical and practical aspects of interface design. Drawing on the work of digital humanities scholars, and our own interface design prototyping project, this paper will describe how we used a design-oriented research (Fallman, 2007) approach to reflect on broader philosophical questions related to the future of the book and the future of reading practices. Specifically, this paper will describe how our narrowly-construed interface design project helped us dig deeper into questions such as: what is an adequate representation of a text?¹ What can, should or must be included in the translation into digital form? What can, should or must be excluded? How can digitization processes enable or

¹ View our prototype interface at: <http://184.154.14.176/~revision/>

foreclose the formulation of research questions and the application of research methods? How might an interface affect, extend or circumscribe reading practices? Ultimately, this paper will show the complexity involved in designing an online reading and research interface, as well as reveal interesting tensions that emerge between theoretical conceptions of the interface and actually-existing tools and systems. Drucker (2013) suggests that “the task of designing an interface is probably the best exercise in reading one can provide”(p. 217). Similarly, this paper concludes by suggesting that the task of designing an interface is probably the best exercise one can engage in to unpack the implications of digitization for reading and research practices.

DOR: Building an interface for Johnson’s revisions

Fallman (2007) describes design-oriented research (DOR) as a research method that seeks to produce new knowledge by including design activities in the research process. DOR involves producing prototypes to help researchers think through complicated theoretical issues and concerns. For this project, we employed a DOR framework to help us think through complicated theoretical issues and concerns related to the design and use of digital interfaces in the humanities. We believed that by engaging in a prototyping exercise we could create a unique opportunity to ground highly-abstracted issues related to the translation, interpretation and representation of texts. Specifically, we thought it would be interesting to catalogue features, functionalities and tools that digital scholars describe as important or desirable in their theoretical texts, and see what happens when we try to implement those same features, functionalities and tools in an online space. By engaging in this kind of play, we engage a practice that Schön (as cited by Fallman, 2007) has called “problem setting.” In problem setting, speculative claims made in literature are tested against prototypes built during design processes. The resulting research objects help probe the contours of existing paradigms and can

help researchers to think beyond the limits of those paradigms. By doing so, DOR can lead researchers to reconsider or redefine the problems and issues at stake within a given field. In other words, by refining the problem description it may be possible to find new ideas or new ways of doing things. For this project we use DOR to help us understand how the problem of the interface is framed in digital humanities literature, how this problem can be practically addressed using XML markup and other digital tools, and what new ideas might emerge from bringing these two perspectives together.

For our prototyping exercise we set ourselves the task of digitizing and building an interface for reading, a two-page spread from the revision notes that Samuel Johnson made to the “B section” of his 1755 publication of *A Dictionary of the English Language*. This primary text is comprised of both printed text and handwritten revision notes. All of the handwritten revisions that Johnson made to the “B section” were intended to be included in the 1773 version of the *Dictionary*, however these changes were never actually incorporated into the final publication. Johnson scholar Allan Reddick suggests that the revisions made to the “B section” were omitted from the publication for the simple reason that these notes were misplaced and never given to the compositor (Johnson, 2005). In 2005, more than two hundred and thirty years after Johnson revised his work, the “B section” modifications finally saw publication when Reddick edited a scholarly text containing photographic reproductions of the original revised pages, a transcription of all the suggested changes, as well as his own notes and annotations on this work. Johnson’s *Dictionary* has rich and complicated history and we thought it would be interesting to see if we could build an interface that could capture some of the dynamism of this text. Specifically, we wanted to build upon Reddick’s work by building an interface that would allow scholars to examine Johnson’s original 1755 publication of the *Dictionary*, the revisions he made in the “B section” and Reddick’s notes and annotations, all in a single space.

Our challenge was to produce a diachronic reading experience that would seamlessly bring primary sources and secondary research together into an integrated online interface. We wanted to show the actual changes and additions made to this text over time, as well as signal that this text—like every text—is part of a larger sociohistorical process. To this end we pursued two interlaced goals: the first was to explore ways to mark-up the revisions in XML so that we could technically build a tool where readers could tack between Johnson’s original published dictionary, Johnson’s revisions, and Reddick’s notes. The second was to draw on scholarship from the digital humanities to design an interface would intuitively be understood as staging a reading experience across time, and as revealing the historicity of the text.

Some Theoretical Approaches to the Interface

In order to attend to our interface design goals we began by surveying some of the literature on interface design in the digital humanities. We wanted to better understand the specific qualities and characteristics that scholars working in this field valued in their digital reading and research tools. This survey of the literature revealed two broad themes that would inform our own prototyping process.

Digital Publishing vs. Digital Scholarship

The first and most salient theme that emerged was the requirement that digital reading and research tools foreground scholarship over publishing. Much of the discussion surrounding the design and development of useful digital interfaces focused on the need to move beyond simply making a text available for reading online. Although converting an analog text into a digital format is helpful, and can make a text accessible to more people, such a practice does not always allow researchers to take full advantage of the affordances offered by digitization. Simply

transcribing a text into a digital space does not in and of itself allow for experimentation with computational strategies. In order to ask new questions of a text and in order to pursue new methods to investigate a text, digitization projects must make use of sophisticated metadata, coding and data storage tools. Indeed, as Bath and Harkema (2013) suggest, the best digital humanities interfaces do more than simply make a text available for *viewing*, these tools make a text available for *study* (p. 4). Similarly, Crompton and Siemens (2013) argue that a successful digital humanities interfaces “...offers new ways of ordering and therefore encountering textual content” (p. 3). By making use of mark-up and database tools, digital interfaces should enable innovative ways of engaging with texts, and provide new ways to think *through* and *with* materials. Tasovac (2010) explains this approach as a move away from the creation of static objects and a step towards the development of services and interactive methods (p. 2). The focus here shifts from digital tools enabling the presentation of text to the reader towards supporting each researcher’s unique investigative processes. Some examples of the specific kinds of tools that digital humanities scholars describe as useful for their research include: interactive visualizations, formatting materials as relational databases that can be queried in numerous ways, and developing avenues to support user-generated content and interactions (Rockwell, Ruecker, Windsor, Ilovan, & Sondheim, 2013; Tasovac, 2010; Wagner, Bratteteig, & Stuedahl, 2010; Wooldridge, 2004).

Situatedness and Meaning-Making

In addition to opening up a text for a different kind of study, digital humanities literature suggests that successful interfaces acknowledge the situated and partial perspectives of every reader, and of every interface. Successful digital interfaces should not make claims to any kind of objectivity, but rather admit that they structure specific kinds of reading

experiences, and consequently, that they make particular arguments about the text. Drucker (2011) summarizes a key aspect of this theme when she argues “[i]nterface and its relation to reading ...[is] an environment in which varied behaviors of embodied and situated persons will be enabled differently according to its many affordances” (p. 12). In other words, every person will approach a given interface from a unique perspective and, as such, it remains an important task for designers and researchers to unpack the effects and affects of the specific functionalities and arguments built into each tool. By doing so, it may be possible to make more evident the specific manner in which digital reading and research tools inform interpretations of a text. Burdick et al. (2012) pick up on many of the same ideas when they argue that “[e]very migration from analog to digital is a translation that stages *a certain experience* of artifacts encountered online” (p. 102). Every interface is an interpretation of the materials it represents. Or, to borrow from Sperberg-McQueen (1991), every interface is a theory of the text. As an already-existing interpretation, every interface can enable or disable particular kinds of analyses and meanings depending on who is engaged in the reading practice.

When Theory Meets Practice

Once we recognized the two key themes that emerged from the literature, we turned to our prototyping exercise to experiment with ways of designing and implementing an interface. We wanted to see if we could design a tool that would meet three interrelated goals: the first, was to develop a digital tool that would enable scholarship rather than mere publishing. Next, we wanted to build an interface that would acknowledge the situatedness of reading and meaning-making practices. Finally, we wanted to develop a tool that would signal the rich history and life of the *Dictionary*.

Our main interest in our experimental phase was in testing some of the hypotheses made in digital humanities literature. We were not interested in developing a tool from scratch or even creating a fully-functional and finished product. As a result, we made the decision to design our interface using XML and modular components from the open source content management system Drupal. We deployed XML using the Text Encoding Initiative's (TEI) guidelines. TEI provided us with a more standardized framework for analyzing and marking-up our material. Instead of developing an encoding strategy from scratch, TEI enabled us to draw upon the work of scholars engaged in digitization and mark-up projects, and focus on tagging methods that would enable us to meet our design goals. Similarly, we turned to Drupal because it is a platform that specifically enables developers to select from libraries of existing tools to build customized digital products (See Appendix A for a description of the tools we used). Working with sets of already-existing tools enabled us to design quickly, to move through several prototypes, and to focus on developing our understanding of various theories of the interface, all without getting bogged down in the details of building a system from scratch.

As we moved back and forth between the theories described in the literature and the practical application of those theories in XML and Drupal, we found that some concepts articulated by scholars were easier to apply than others, that some interesting ideas related to interface design and interactivity far outstripped the capabilities of already-existing digital tools, and that some concepts may, in fact, best be described as 'vaporware'--meaning they do not currently exist and are not likely to ever exist (Galey & Ruecker, 2010). By engaging in a design project alongside a theoretical investigation of interface we ultimately gained a better perspective of the problems of interface within the digital humanities.

Digital Publishing vs Digital Scholarship

By virtue of how we deployed XML to code our materials and how we implemented various Drupal modules, we believe we were able to create an interface that allows for the *study* of the revisions that Johnson made to his *Dictionary*. Our project does not merely present this content for *viewing*. Indeed, our interface enables the study of multiple versions of the *Dictionary* text across time, including the 1755 published text, Johnson's revisions, all of Reddick's notes and annotations, as well as a view of what the "B section" would have looked like had Johnson's revisions actually been published as part of the fourth edition of the dictionary. Additionally, readers have the opportunity to view all the content of the two-page spread, or hone in on a specific entry for a closer examination. Moreover, the specific manner in which we mobilized our XML mark-up allows for the dynamic search and display of other query results, for example generating a list of all the entries that underwent revisions, or list of all the adjectives listed in the *Dictionary*. Our interface enables easy comparison of numerous documents, and of numerous dimensions of those documents, within a single frame. Such a tool helps structure multiple entryways into Johnson's materials. By doing so, our interface may help develop the kinds of dynamic relations between reader and text that Drucker (2003) suggests are integral to interpretation and the production of meaning.

In addition to carefully marking up our materials in XML so that they could enable the greatest number of views and queries, the most important aspect of our design process was approaching the material from the perspective of a researcher. We could not design in a vacuum, we had to imagine this prototype in use. Although no one on our team is an expert in lexicography, in Johnson nor in the eighteenth century, we found--exactly as Bath and Harkema (2013) note--that we could not begin the prototyping process without first considering the kinds of questions are likely to be asked of this material (p. 4). Subject expertise must meet technical expertise in order to develop a useful and successful digital humanities interface. If we were to

attempt to build upon our experimental prototype in order to develop a finished product, we would certainly need to add a scholar familiar with Johnson's work to our team. If, as Crompton and Seimens (2013) argue, "[i]nterfaces both engage and shape the practices of the research communities they serve," then we need to draw on the knowledge and expertise of a researcher situated within studies of the eighteenth-century in order to better understand the kinds of research methods and research questions that would be deployed within this context. More than anything else, the process of making materials digitally available for *study* revealed the complicated acts of interpretation that occur at the level of code, the at-times uncomfortable decisions that must be made to obscure certain elements from view, and the ability to justify those decisions. These are not easy decisions, and they are not decisions that a single designer or researcher can make.

As we experimented with designing an interface that could enabled the study of our dictionary materials we found that we were able to mobilize existing tools to develop a range of entryways into our texts. There are many tools that a researcher or developer can draw upon to produce new encounters with the text. The challenge in meeting this requirement comes from being able to ask the right questions of the material, and finding ways to translate those questions into code and system. Although technologies already exist to facilitate the online study of texts, putting together the right combination of technical and subject expertise on a research team remains a more salient challenge.

Situatedness and Meaning-Making

Designing an interface that did not overcode reading practices was a much more difficult goal to meet. In part, this difficulty stemmed from the need to unpack exactly what is meant by keeping an interface open to ambiguity, and to the partial and situated perspectives of

each reader. What does this look like? What does it feel like to use? How do we know when we have achieved an appropriate level of ‘openness’? Another reason for this difficulty is that existing digital technologies do not make room for ambiguity or partiality. Systems and code are highly structured, follow a rigid logic and are not equipped to handle uncertainty. How can we mark-up a text to allow for the greatest range of interpretation and analysis? How can we import our XML into an online space in ways that promote flexibility and alternative perspectives?

We did not come to satisfactory answers for any of these questions. We found that most of our interface’s flexibility was contingent on the kinds of tags we deployed in our mark-up, and on the generous use of attributes within those tags. The more descriptive we were with our mark-up, the more we could target for user-generated queries and structured display within our online reading and research tool. However, at the same time, we realized that there was a limit to what we could tag and then *productively* put to use within the interface. Although almost every single word of our various dictionary texts could be descriptively marked up in some way, there was no way for us to make use of every description *and* produce a useable interface. The tenets of usability stipulate that online interfaces should contain an obvious starting point, make use of consistent logic, observe existing conventions and foreground the materials presented rather than the interface itself (Blair-Early & Zender, 2008). Balancing these requirements with an impulse to provide hundreds of different views and feeds of the content was extremely difficult. We struggled to find good ways to provide readers flexibility in reading and research while also making a useable tool. We were cognizant of the dangers of predisposing readers to our interpretations of the material, particularly since we are not Johnson experts, however we found it difficult to develop an interface that was clean and easy-

to-use, but that also enabled novel combinations and recombinations of materials in as many ways as possible—particularly given the short timeframe in which we were operating.

The various starts and stops we encountered as we experimented with our open interface design led us to realize that the scope of this goal may be too large for a single project to achieve. Rather, we hypothesized that ambiguity and openness may be able to emerge from the various and multiple iterations of a single project, as well as from multiple networked projects. Rather than placing the burden of designing for partiality and situatedness on a single digital humanities project, a better approach may be for digital researchers to make their coding and development processes open to others.

If digital humanities teams are free to extend, build upon and change existing work, they may be better equipped to build interfaces that can better contend with uncertainty and partiality. Imagining digital humanities projects as feeding into larger research ecosystems may allow individual readers to pick and choose the tools and perspectives that suit their own research agendas and may provide opportunities for even more novel combinations and recombinations of materials. We like this approach because it does not require that an individual project must theorize and code for maximum ambiguity. Rather, a project can strive for openness and flexibility by exposing the coding and decisions that were built into its system, and by allowing other readers to download, use, manipulate and re-upload code to produce new branches of research within the same sphere. It is perhaps only by engaging with numerous projects and multiple approaches that one can begin to account for and engage the situated and partial perspectives of many readers and researchers.

A Possible Model for the Digital Research Ecosystem

Given our interest in exploring praxis and in ROD we thought we would offer some thoughts on a model for creating such a digital research ecosystem. For the same reasons we turned to XML and Drupal, we would advise making use of already-existing tools to facilitate sharing and exchange. Specifically, we think it may be interesting to explore the possibilities offered by a version control system like Git. Git is a revision tool that was specifically designed to help programmers and developers keep track of their code, as well as any changes made to their code by themselves or others in the programming community. Git allows users to access project code, grab pieces they need, then recombine and rewrite as necessary to fulfill the aims of their individual projects. Every time a developer rewrites or adds to a piece of code it creates a 'branch' or a 'fork' within Git that subsequent users can then draw upon for their own programming needs. Digital humanists could make use of Git to, for example, upload their XML files and other coding materials to a platform so that other researchers can study their decisions, clone their code and reuse and extend their work on other projects. As Wittern (2013) suggest, such a practice would allow for local corrections, annotations and/or additions to be merged with, or exist alongside, other projects (p. 5). Although no tools currently exist to upload a new XML file to a platform like Drupal and have the resulting online tool automatically change to accommodate the new material, a robust versioning system may allow scholars to branch or fork their research in new, unexpected and productive ways.

Conclusions and Future Directions

Our DOR approach to interface design has shown us how valuable it is to combine research and practice when thinking through issues in the digital humanities. Engaging in such a design project provides the unique opportunity to bring together theoretical concepts relating interface design with robust tools like XML mark-up and Drupal modules. This combination of

approaches helped to enrich our understanding of the problem of the interface, as articulated by literature in the digital humanities. Specifically, our prototyping project helped us understand some of the key features of a successful interface, helped us recognize what is and is not possible to technically implement, and helped us posit models for a more open and flexible digital research ecosystem. In addition, contributing to a design exercise helped us better understand the materials we were digitizing because this work required us to read closely, as well as consider larger questions that may be asked of this document by other researchers. We had to think both broadly and deeply about this material. Moreover, this exercise helped us better understand theories of the interface and consider the implications of various digitization practices. The task of designing an interface is, indeed, one of the best exercises one can engage in to unpack the implications of digitization for reading and research practices.

References

- Bath, J., & Harkema, C. (2013). There's More Than One Way to Skin a Book: Experimental Interfaces for Reading Illustrated Books. *Scholarly and Research Communication*, 3(4). Retrieved from <http://src-online.ca/index.php/src/article/view/61>
- Burdick, A., Drucker, J., Lunenfeld, P., Presner, T., & Schnapp, J. (2012). *Digital humanities*. Cambridge, MA: MIT Press.
- Crompton, C., & Siemens, R. (2013). Introduction: Research Foundations for Understanding Books and Reading in the Digital Age: Text and Beyond. *Scholarly and Research Communication*, 3(4). Retrieved from <http://src-online.ca/index.php/src/article/view/112>
- Drucker, J. (2003, April 25). *The Virtual Codex from Page Sapce to E-space*. Lecture presented at the History of the Book Seminar, Syracuse University. Retrieved from <http://www.philobiblon.com/drucker/>
- Drucker, J. (2009). *SpecLab: digital aesthetics and projects in speculative computing*. Chicago: University of Chicago Press. Retrieved from <http://myaccess.library.utoronto.ca/login?url=http://books.scholarsportal.info/viewdoc.html?id=/ebooks/ebooks2/ucpbooks/2012-03-30/1/9780226165097>
- Drucker, J. (2011). Humanities Approaches to Interface Theory. *Culture Machine*, 12(0), 1–20.
- Drucker, J. (2013). Reading Interface. *PMLA*, 128(1), 213–220.
- Fallman, D. (2007). Why Research-Oriented Design Isn't Design-Oriented Research: On the Tensions Between Design and Research in an Implicit Design Discipline. *Knowledge, Technology & Policy*, 20(3), 193–200. doi:10.1007/s12130-007-9022-8
- Galey, A., & Ruecker, S. (2010). How a prototype argues. *Literary and Linguistic Computing*, 25(4), 405–424. doi:10.1093/lc/fqq021

- Hayles, N. K. (2003). Translating Media: Why We Should Rethink Textuality. *Yale Journal of Criticism*, 16, 263–290.
- Johnson, S. (2005). *Samuel Johnson's unpublished revisions to his Dictionary of the English language* (Facsimile ed.). Cambridge, UK ; New York: Cambridge University Press.
- Rockwell, G., Ruecker, S., Windsor, J., Ilovan, M., & Sondheim, D. (2013). The Face of Interface: Studying Interface to the Scholarly Corpus and Edition. *Scholarly and Research Communication*, 3(4). Retrieved from <http://www.src-online.ca/index.php/src/article/view/56>
- Sperberg-McQueen, C. M. (1991). Text in the Electronic Age: Textual Study and Textual Study and Text Encoding, with Examples from Medieval Texts. *Literary and Linguistic Computing*, 6(1), 34–46. doi:10.1093/lc/6.1.34
- Tasovac, T. (2010). 'Reimagining the Dictionary, or Why Lexicography Needs Digital Humanities. *Digital Humanities 2010*. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.385.3540&rep=rep1&type=pdf>
- Wagner, I., Bratteteig, T., & Stuedahl, D. (2010). *Exploring digital design: Multi-disciplinary design practices*. Springer. Retrieved from <http://books.google.com/books?hl=en&lr=&id=zf3mh5cHjBsC&oi=fnd&pg=PR5&dq=%22design+that+bring+about+these+digital+elements+in+our+lives.+This+book%22+%22and+the+Department+of+Media+and+Communication.+The+book+was%22+%22the+process+of+making+a+book.+Our+first+thanks+go+to+the+departments%22+&ots=IHWuS6mGe4&sig=SsaZ5m7Ro704UVlssmpnZiq9UMY>
- Wittern, C. (2013). Towards an Architecture for Active Reading. *Scholarly and Research Communication*, 3(4). Retrieved from <http://src-online.ca/index.php/src/article/view/59>

Wooldridge, R. (2004). Lexicography. In S. Schreibman, R. G. Siemens, & J. Unsworth (Eds.), *A companion to digital humanities* (pp. 69–78). Malden, MA: Blackwell Pub.

Appendix A

View our prototype interface at: <http://184.154.14.176/~revision/>

We used Drupal, an open source content management to build our prototype. Drupal system is very popular among programmers and developers due, in part, due to its modular functionality.

This modular functionality was very useful in the parsing of our XML code and our experiments with the interface. In particular, the following Drupal community contributed modules were necessary to make this project possible:

Feeds

<https://drupal.org/project/feeds>

Feeds is the module that allows content to be imported from other sources into the site's content database. By itself, it is able to import RSS, XML, and CSV files. Essentially, this module takes data in any of the supported formats, and maps the content to fields within a custom content type.

The approach for this project was to use Feeds to read our XML and import the data into a content type that we called Entry. Each dictionary entry would be a piece of content on our site with fields such as definition, etymology, citation, etc. Feeds would map the different XML tags to their proper field for each entry.

Feeds XPath Parser

https://drupal.org/project/feeds_xpathparser

While the basic Feeds module supports XML, it does not support parsing of tags such as <entryFree>, on its own. This module allows developers to configure Feeds to read any tags that exist in your XML file, and provides a mechanism to map the data contained within the tags, as well as within the attributes, to fields within your custom content type. In our case for example, we were mapping data within the <etym> tag to the Etymology field within our Entry content type.

Xpath is a query language that is used to navigate through XML documents.

Feeds Tamper

https://drupal.org/project/feeds_tamper

Feeds Tamper is another module that extends the functionality of the Feeds module. It allows one to manipulate the data from the XML file and rewrite it before it is added to the Drupal content type. The functionality for this is best described by an example.

For our project, we needed a way to separate different chunks of content. We will look at the Bombast example and see that there is a handwritten note that Johnson has made for this entry. During our encoding, we used the attribute `xml:id="Johnson"` to note this. However, this code does not help us display this text in different manners, or even hide it, in HTML. We needed Feeds Tamper to take any instance of this code and convert it to HTML that we could

then manipulate with CSS. Feeds tamper allowed us to define the following replacement:

From:

```
<handNote xml:id="Johnson">
```

to:

```
<div class="handNote-Johnson">
```

And of course, the corresponding closing tags.

This was done for a number of XML tags.

@Font Your Face

<https://drupal.org/node/1551382>

This module allowed our interface to display non-system fonts. In particular, we wanted handwritten notes to look like they were written by hand, and so we used this module to help accomplish that. It takes open fonts, in our case, Google fonts and allows for their display as HTML.

Views

<https://drupal.org/project/views>

Finally, views is used to take data and manipulate how it is displayed. This is very important to this project as it allowed us to do basic things like make a list of all of the entries and display them alphabetically.

However, it can also allow the developer or even the end user, to create queries to show certain entries based on different selection criteria.

While not created for this project due to time constraints, Views would enable the developer to create an interface that would show only the entries where Johnson has made a revision, for example. If this project were extended further, it could do things like show a listing of all entries that have citations. Or even complex combinations like showing all entries in the H section, that have additions by Johnson and have accompanied notes by Reddick.

Appendix B

This document describes how we tackled the mark-up and transcription of the revisions Samuel Johnson made to the “B section” of the 1755 edition of *A Dictionary of the English Language*, as well as the notes and annotations on this text published by Allan Reddick in *Samuel Johnson’s Unpublished Revisions to the Dictionary of the English Language*. For future transcriptions, these guidelines could be followed and the resulting XML file could be imported into the parser we have created.

We were strict about following TEI (Text Encoding Initiative) Guidelines so that our resulting XML could be as close to a standard document as possible.

To read this document, a basic understanding of XML will be assumed. Familiarity with parent-child relationships, attributes, and values is expected.

When encoding your document, please be careful to use the proper spacing and case, as this is very important for the parser. There is a big risk of your document not being read correctly if these details are ignored

Document Structure

Each XML document is opened with the <page> tag. The immediate child of <page> for the purposes of encoding the dictionary is the <entryFree> tag. As noted above, this structure must be adhered to in order for the parser to target each entry.

<entryFree>

If the dictionary entry is straightforward with no corrections for typos, no notes by either Johnson or Reddick, this tag does not need to contain any attributes. However, in order to make note of those dictionary entries that have notes or edits attached, the following attributes and values are to be used:

- xml:id="Johnson" - if there are any revisions made to the entry by Johnson
- xml:id="Reddick" - if there are notes or comments that Reddick has added
- etym="yes" - if this entry has information for the word’s etymology

The reasoning behind this is so that the parser can make note of which entries contain any revisions or notes and therefor, allowing the ability to query the entries for those that have certain features.

Following <entryFree> are its three children, <form>, <etym>, and <sense>.

<form>

<form> contains the children <orth>, <pron>, <gramGrp>, and if needed <colloc>. <orth> is the orthographic form, essentially how it is spelled. <pron> is the word’s pronunciation. <gramGrp> represents the grammatical information which indicates whether the word is a verb, noun, etc. Lastly, if needed, the <colloc> tag is used for sequences of words that appear frequently together. For example, in our transcription To Bomb and To Bombard are two such examples. Since Johnson chooses to add this collocation, we must make note of it in our

transcription so that we can display the entries as close to Johnson's original as possible. The alphabetical sort occurs on the <orth> tag, however, if we added the "To" as part of <orth>, the sorting would not be correct. Therefore, collocated words have their own tag, so sorting can still happen correctly by sorting with <orth>.

<etym>

This tag contains encloses any etymology if it is present. If there is no etymology, we still include the self-closing tag <etym /> which is important to the parser.

<sense>

This tag contains the bulk of most of the dictionary entries. <sense> has two children: <def> and <cit>, for definition and citation respectively. The definition is straightforward as it contains the definition of the word. Citation, if there is one contains a quote, enclosed by the <quote> tag and an attribution, enclosed by the <bibl> for bibliographic citation.

Notes and Revisions

Just using the tags above will be sufficient for encoding the bulk of the dictionary. For notes on encoding any revisions by Johnson or notes by Reddick, we use a few special tags.

Johnson revisions and Reddick notes

Within the etymology of the definition, there are some parts of the dictionary where Johnson had made revisions and/or added handwritten notes. For the guide, we will look at the example of Bombast. Below is the complete <etym> section for this entry:

```
<etym>
  [This word
  <msDesc>
  <physDesc>
  <handDesc>
    <handNote xml:id="Johnson">
      <subst>
        <del type="overstrike">seems to</del>
        <add place="above" type="insertion">mi</add>
        <note xml:id="Reddick">J. apparently begins writing the word 'might' above cross-
out.</note>
      </subst>
    </handNote>
    be derived from Bombas-
  <lb />
  tius, one of the names of Paracelsus; a man remarkable for
  <lb />
  founding professions, and unintelligible language
  <add type="insertion">^</add>
  <lb />
  <handNote xml:id="Johnson">
    <add place="margin-left" type="insertion">
      or from
```

```

    <u>bombast</u>
    the
    <lb />
    name of some kind of
    <lb />
    stuff perhaps of dissimi-
    <lb />
    lar materials whence
    <lb />
    it came to be applied
    <lb />
    to style of which one
    <lb />
    part suites all with
    <lb />
    the other, as Fustian has
    <lb />
    the same use
    <lb />
    </add>
  </handNote>
</handDesc>
</physDesc>
</msDesc>
.]
<lb />
</etym>

```

To be able to use the <handNote> tag, we had to add <msDesc> - manuscript description, <physDesc> - physical description, and <handDesc> which describes all the different kinds of handwriting within a document. It is important that you specify that the note is by Johnson by using the xml:id="Johnson" attributes for the <handNote> tag.

If there is a deletion (cross-out) and addition you must use the substitution tag <subst> which has the children for delete and <add> for add. Further, you can specify the type of deletion as in our example with the attribute type="overstrike". The place attribute (place="margin-left") indicates where Johnson has made the addition.

If there are notes added by Reddick, we use the <note> tag, again with attribute xml:id="Reddick" to specify that this note is an addition by Reddick.

Lastly, we follow the line breaks in the text by using the <lb /> tag to specify where every line breaks. This is important in coding for an accurate representation of the text by the interface.