**FLANDERS — Challenges**

<0 images>

**Challenges of an XML-Based Open-Access Journal: *Digital Humanities Quarterly***

Flanders, J., Piez, W., Walsh, J. and Terras, M.

*Digital Humanities Quarterly* was founded in 2005 as ADHO’s first online open-access journal. In the ensuing ten years, the journal has been conducted as an ongoing experiment in standards-based journal publishing. This paper reflects on the results of that experiment to date, with emphasis on a few areas of particular challenge that reveal important methodological tensions in the ways that the digital humanities research discourse is expressed.

# Background and Technical Infrastructure

*DHQ*’s technical design was constrained by a set of higher-level goals and needs. As an early open-access journal of digital humanities, the journal had an opportunity to participate in the curation of an important segment of the scholarly record in the field. This meant that it was more than usually important that the article data be stored and curated in a manner that would maximize the potential for future reuse. In addition to mandating the use of open standards, this aim also strongly indicated that the data should be represented in a semantically rich format. Of equal concern was the need for flexibility and the ability to experiment with both the underlying data and the publication interface, throughout the life of the journal, without constraint from the publication system. Both of these considerations moved the journal in the direction of XML, which would give us the ability to represent any semantic features of the journal articles we might find necessary for either formatting or subsequent research. It would also permit us to design a journal publication system, using open-source components, that could be closely adapted to the *DHQ* data. At the journal’s founding, several alternative publishing platforms were proposed (including the Open Journal System), but none were XML-based and none offered the opportunity for open-ended experimentation that we needed.

*DHQ*’s technical infrastructure is a standard XML publishing pipeline built using components that are familiar in the digital humanities. Submissions are received and managed through OJS through the copyediting stage, at which point articles are converted to basic TEI using OxGarage (http://www.tei-c.org/oxgarage/). Further encoding and metadata are added by hand, and items from the articles’ bibliographies are entered into a centralized bibliographic system that is also XML-based. All journal content is maintained under version control using Subversion. The journal’s organizational information concerning volumes, issues, and tables of contents is represented in XML using a locally defined schema. The journal uses Cocoon, an XML/XSLT pipelining tool, to process the XML components and generate the user interface.

# *DHQ*’s Evolving Data and Interface

As noted above, *DHQ*’s approach to the representation of its article data has from the start been shaped by an emphasis on long-term data curation and a desire to accommodate experimentation. The specific encoding practices have evolved significantly during the journal’s lifetime. The first schema developed for the journal was deliberately homegrown and was designed based on an initial informal survey of article submissions and articles published in other venues. Following this initial period of experimentation and bottom-up schema development, once the schema had settled into a somewhat stable form we expressed it as a TEI customization and did retrospective conversion on the existing data to bring it into conformance with the new schema. At several subsequent points, significant new features have been added to the journal’s encoding: for example, explicit representation of revision sites within articles (for authorial changes that go beyond simple correction of typographical errors), enhancements to the display of images through a gallery feature, and adaptation of the encoding of bibliographic data to a centralized bibliographic management system.

These changes to the data have typically been driven by emerging functional requirements, such as the need to show where an article has been revised or the requirements of the special issue on comics as scholarship. However, they also respond to a broader set of requirements that this data should represent the intellectual contours of scholarship rather than simply interface. For example, the encoding of revision notes retains the text of the original version, identifies the site of the revision, and supports an explanatory note by the author describing the reason for the revision. Although *DHQ*’s current display uses this data in a simple manner to permit the reader to read the original or revised version, the data would support more advanced study of revision across the journal. Similarly, although our current display uses the encoding of quoted material and accompanying citations in very straightforward ways, the same data could readily be used to generate a visualization showing most commonly quoted passages, quotations that commonly occur in the same articles, and similar analyses of the research discourse. The underlying data and architecture lend themselves to incremental expansion.

# Analysis

The approach *DHQ* has taken offers several significant advantages and also some corresponding disadvantages. The most important advantages are

• The autonomy the journal has to control all aspects of its own data modeling and interface.

• The high value of the resulting data, from a historiographic perspective.

• The ease of long-term curation of the data, including continuing evolution of our modeling decisions.

• The ease of long-term evolution of the publication infrastructure, including migration to other XML-based systems as needed.

• The scalability of a template-based infrastructure: with the system in place, each article requires no incremental work in styling or design; all effort goes towards consistent representation of semantically valued features.

These advantages all carry a burden of cost and effort: autonomy and control necessarily entail responsibility for maintaining appropriate levels of expertise and undertaking the labor necessary to build and revise technical systems. Because our article work flow includes some hand encoding in TEI, our managing editors need to be better trained and more expert than if they were simply formatting articles in Word and exporting PDF. However, there are also some less obvious tradeoffs. *DHQ*’s publication model gains its efficiencies and scalability through an emphasis on uniform handling of repeated features, but this means that it is comparatively difficult to accommodate individual authorial requests for special handling. These entail not only extra effort at the time of publication but also the long-term prospect of special attention during the future data curation activities and updates to the interface and publication system. Authors familiar with content management systems such as WordPress or Scalar are accustomed to being able to exercise a significant level of control over the formatting and behavior of their text and accompanying media such as images and video. Long-term data curation is a less visible feature of such publishing systems.

Even more interesting and challenging are the special cases that entail semantically distinctive features. Although such submissions are rare, they have provided some valuable test cases in which the data being represented is not a straightforward ‘article’ but some other rhetorical mode: commented program code, dynamic HTML that provokes reader interaction, an article in the form of a comic book. In handling these cases, *DHQ* has sought to find ways to accommodate the distinctive form of the original piece while also giving it a proxy presence within the standard *DHQ* XML archive, so that its content can be searched and analyzed as part of the larger *DHQ* corpus of DH scholarship. As these cases accumulate, the editors seek to identify repeated needs that could become part of the regular *DHQ* feature set.

In the full version of this paper, we will consider in greater detail the role of authorial design in digital humanities publication, and the possible convergences between XML-based systems like *DHQ* and content-management based systems like Scalar.

# Future Directions

*DHQ* is now completing a multiyear project to centralize its bibliography, and the next step will be to develop interface features that exploit this data. We are also in the planning stages of a project to explore internationalization of the journal through a series of special issues dedicated to individual languages. In both cases, these amplifications of the journal represent natural extensions of the journal’s existing architecture, and although both are substantial projects, they are made feasible by the investment already made in strongly modeled data and an extensible publication infrastructure. In the fuller version of this paper, we will discuss both of these developments in greater detail.