Towards Hermeneutic Markup: An architectural outline

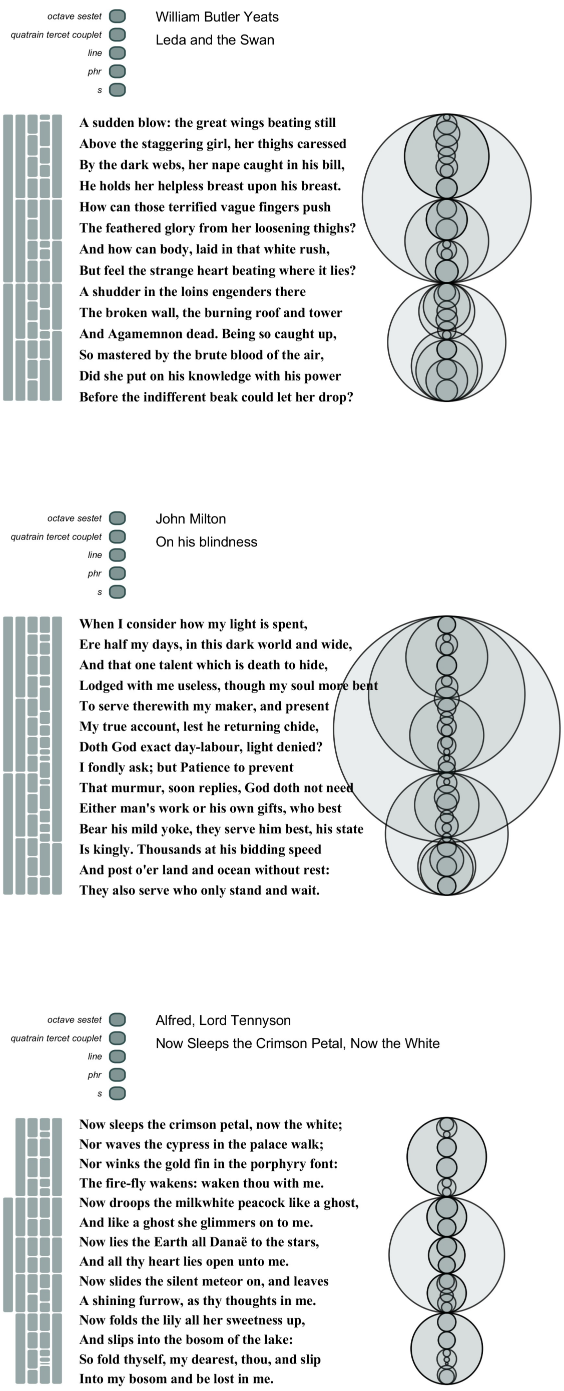
Piez, Wendell
wapiez@mulberrytech.com
Mulberry Technologies, Inc., USA

By "*hermeneutic*" markup I mean markup that is deliberately interpretive. It is not limited to describing aspects or features of a text that can be formally defined and objectively verified. Instead, it is devoted to recording a scholar's or analyst's observations and conjectures in an open-ended way. As markup, it is capable of automated and semi-automated processing, so that it can be processed at scale and transformed into different representations. By means of a markup regimen perhaps peculiar to itself, a text would be exposed to further processing such as text analysis, visualization or rendition. Texts subjected to consistent interpretive methodologies, or different interpretive methodologies applied to the same text, can be compared. Rather than being devoted primarily to supporting data interchange and reuse – although these benefits would not be excluded – hermeneutic markup is focused on the presentation and explication of the interpretation it expresses.

Hermeneutic markup in its full form does not yet exist. XML, and especially TEI XML, provides a foundation for this work. But due to limitations both in currently dominant approaches to XML, and in XML itself, a number of important desiderata remain before truly sophisticated means can be made available for scholars to exploit the full potentials of markup for literary study, as implied, for example, by ideas such as Steven Ramsay's Algorithmic Criticism or what I described in 2001 (following Rockwell and Bradley) as "*exploratory markup*" (Piez 2001. See also especially Buzzetti, 2002 and McGann, 2004).

Prototype user interfaces designed to enable one or another kind of *ad hoc* textual annotation or markup have been developed, for the most part independently of one another (several are cited.). This shows that the idea of hermeneutic markup, or something like it, is not new; but none of these have yet made the breakthrough. An important reason is that hermeneutic markup in its full sense will not be possible on the basis simply of a standard tag set or capable user interface, because it will mean not just that we can describe a data set using markup (we can already do that), but that we can actively develop, for a *particular* text or family of texts, an appropriate, and possibly highly customized, means and methodology for doing so.

A demonstration of a prototype markup application helps to show the potentials and challenges, in a very rudimentary form [screenshots appear in Figure 1.] This graphical and interactive rendering of markup in the source files presents an interpretation of the grammatical/rhetorical structure (sentences and phrases) as well as verse structure (lines and stanzas) in the text. Unfortunately, while the encoding for the sonnets here is not inordinately difficult – "*milestones*" are used, in a conventional manner, to denote the presence of structures that overlap the primary structure of the encoded document – the code that renders it (not included in the package) incurs significant extra overhead to run, because XML technologies are ill-fitted to manage the kind of information we are interested in here, namely the overlapping of these structures that characterizes the sonnet form. XML doesn't do overlap. As long as a sentence or phrase overlaps a line – a very common occurrence and important poetic device – the normative XML data model, a "*tree*", cannot capture both together. In order to do processing like what happens here, one or another workaround has to be resorted to. So while XML is being used here, it is a clumsy means to this end.



William Butler Yeats
Leda and the Swan

A sudden blow: the great wings beating still
Above the staggering girl, her thighs caressed
By the dark webs, her nape caught in his bill,
He holds her helpless breast upon his breast.
How can those terrified vague fingers push
The feathered glory from her loosening thighs?
And how can body, laid in that white rush,
But feel the strange heart beating where it lies?
A shudder in the loins engenders there
The broken wall, the burning roof and tower
And Agamemnon dead. Being so caught up,
So mastered by the brute blood of the air,
Did she put on his knowledge with his power
Before the indifferent beak could let her drop?

John Milton
On his blindness

When I consider how my light is spent,
Ere half my days, in this dark world and wide,
And that one talent which is death to hide,
Lodged with me useless, though my soul more bent
To serve therewith my maker, and present
My true account, lest he returning chide,
Doth God exact day-labour, light denied?
I fondly ask; but Patience to prevent
That murmur, soon replies, God doth not need
Either man's work or his own gifts, who best
Bear his mild yoke, they serve him best, his state
Is kingly. Thousands at his bidding speed
And post o'er land and ocean without rest:
They also serve who only stand and wait.

Alfred, Lord Tennyson
Now Sleeps the Crimson Petal, Now the White

Now sleeps the crimson petal, now the white;
Nor waves the cypress in the palace walk;
Nor winks the gold fin in the porphyry font:
The fire-fly wakens: waken thou with me.
Now droops the milkwhite peacock like a ghost,
And like a ghost she glimmers on to me.
Now lies the Earth all Danaë to the stars,
And all thy heart lies open unto me.
Now slides the silent meteor on, and leaves
A shining furrow, as thy thoughts in me.
Now folds the lily all her sweetness up,
And slips into the bosom of the lake:
So fold thyself, my dearest, thou, and slip
Into my bosom and be lost in me.

Figure 1: Screenshots of three sonnets with rendition of overlapping (verse and sentence/phrase) structures. The interface (implemented in W3C-standard SVG) is dynamic and responds to user input to highlight overlapping ranges of text.

But overlap is only part of the problem. Consider Alfred Lord Tennyson's *Now Sleeps the Crimson Petal, Now the White*. This too is a sonnet, after a fashion, although it does not have a conventional sonnet's octave/sestet structure. Since this application does not work with a schema, this is not a problem here. Yet as texts or collections grow in scale and complexity, having a schema is essential to enforcing consistency and helping to ensure that like things are marked up alike. A framework for this application must not only find a way to work around the overlap; it must also deploy a schema (or at any rate some sort of validation technology) flexible enough – at least if this instance is to be valid to it – that such outliers from regular form are permissible, even while attention is drawn to them (see Birnbaum 1997).

Currently, XML developers generally (setting aside the problem of overlap) do not consider this to be problematic in itself; indeed, part of the fun and interest of markup is in devising and applying a schema that fits the data, however strange and interesting it may be. What is not so fun is to have to repeat this process endlessly, being caught in a cycle of amending and adjusting a schema constantly (and sooner or later, scripts and stylesheets) in order to account for newly discovered anomalies. Sooner or later, when exhaustion sets in or the limits of technical knowhow are reached, one ends up either faking it with tags meant for other purposes (thereby diluting markup semantics in order to *pretend* to represent the data), or just giving up.

Extending a schema is found to be a problem not only because validating and processing any model more complex than a single hierarchy is a headache even for technical experts, but also, more generally, because current practices assume a top-down schema development process. Despite XML's support for processing markup even without a schema, both XML tools and dominant development methodologies assume that schema design and development occurs prior to the markup and processing of actual texts. This priority is both temporal and logical, reflecting a conception of the schema as a declaration of constraints over a set of instances (a “type”), appropriate to publishing systems built to work with hundreds or thousands of documents, with a requirement for backwards compatibility (documents encoded earlier cannot be revised easily or at all) and limited flexibility to adapt to new and interesting discoveries. The centrality of the schema within this kind of system inhibits, when it does not altogether frustrate, the flexible development of a markup practice that is sensitive, primarily, to a text under study, and this conception of a schema's authority works poorly when considering a single text *sui generis* – the starting point for hermeneutic markup. In hermeneutic markup, a schema should be, first and last, an apparatus and a support, not a Procrustean bed.

All these problems together indicate the outline of a general solution:

- A data model supporting arbitrary overlap.
- Interfaces, including a markup syntax, that facilitate the creation, editing and analysis of texts using this data model, with the capability of defining *ad hoc* elements and properties (attributes) on the fly.
- A transformation technology supporting (in addition to data transformations) analytical tools applicable to the markup as such (not just the raw text), with the capability of managing elements and their properties in sets, locating them, listing them by type, sorting, visualizing and comparing them.
- Schema-inferencing capabilities for describing the structural relations within either an entire marked-up corpus, or within identifiable segments, sections or profiles of it.
- In connection this, a schema technology that supports partial and modular validation.

A system with all these features would support an iterative and "*agile*" approach to markup development. We would start by tagging. (In a radical version of this approach we might start by tagging for presentation, perhaps using just a lightweight HTML or TEI variant for our first cut.) Then we introduce a provisional schema or schemas capable of validating the tagging we have used. This requires assessing which cases of overlap in the text are essential to our document analysis, and which are incidental and subject to normalization within hierarchies. Having refined the schema, we return to the tagged text, to consider both how its tagging falls short (with respect to whatever requirements we have for either data description or processing), and how it may be enhanced, better structured and regularized. During this process we also begin to develop and deploy applications of the markup. We then revise, refactor and extend both tagging and schema, applying data transformations as needed, in order to better address the triple goals of adequate description, processing, and economy of design.

Such a system would not only be an interesting and potentially ground-breaking approach to collaborative literary study; it would also be a platform for learning about markup technologies, an increasingly important topic in itself. Moreover, hermeneutic markup represents an opportunity to capitalize on investments already made, as texts encoded in well-understood formats like TEI are readily adaptable for this kind of work.

Many of these capabilities have already been demonstrated or sketched in different applications or proposals for applications, including W3C XML Schema (partial validation); James Clark's Trang (schema inferencing for XML); LMNL/CREOLE (overlap, structured annotations, validation of overlap); JITTs (XML "*profiles*" of concurrent overlapping structures); and TexMECS (overlap, "*virtual*" and discontinuous elements).

The presentation will conclude with a demonstration of various outputs from the data sources used in the demo, which provide building blocks towards the kind of system sketched here. A range-analysis transformation can show which types of structures in a markup instance overlap with other structures, and conversely which structures nest cleanly. Complementary to this, an "*XML induction processor*" is capable of deriving well-formed XML representations of texts marked up with overlapping structures – from which, in turn, XML schemas can be derived.

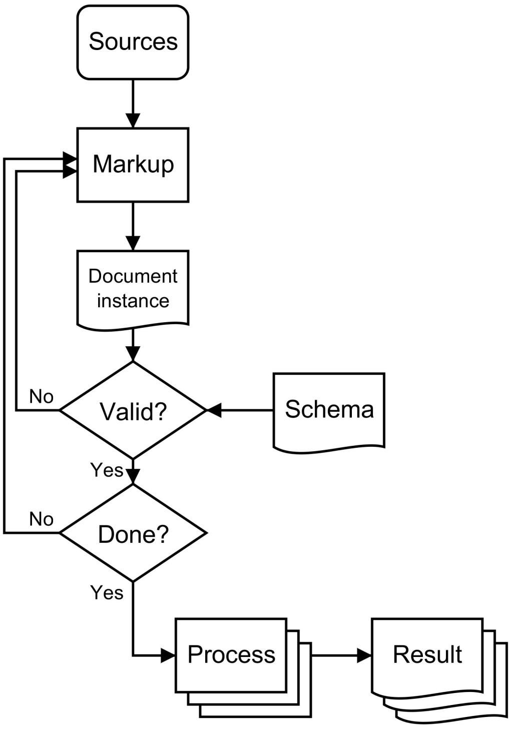


Figure 2: A workflow diagram showing the architecture of present (XML-based) markup systems. Both schema and processing logic are considered to be static; modifying them is an activity extraneous to document markup and production.

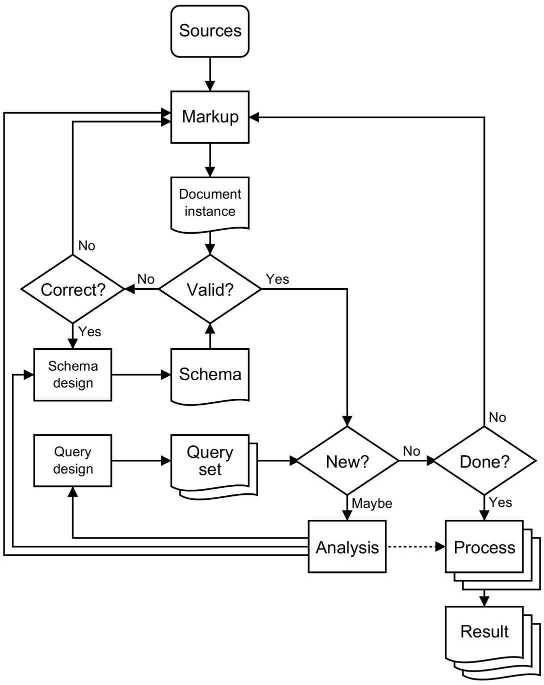


Figure 3: An architecture capable of supporting hermeneutic markup would account directly for document analysis and for the design of schema, queries and processing. While in fact this is often done even today, one has to work against the current tool set to do it, questioning its assumptions regarding the purposes, roles and relations of source text, markup and schema.

A final version of this paper, with the demonstrations, is available at http://piez.org/wendell/papers/dh2010/index.html

References

Birnbaum, David. 'In Defense of Invalid SGML'. *ACH/ALLC*. Kingston, Ontario, 1997.

Buzzetti, Dino (2002). 'Digital Representation and the Text Model'. *New Literary History*. **33(1)**: 61-88.

CATMA. University of Hamburg (Jan Christoph Meister). http://www.jcmeister.de/html/catma-e.html.

Caton, Paul. 'LMNL Matters? (presenting the Limner prototype tagging platform)'. *Extreme Markup Languages 2005*. Montréal, Québec, 2005.

Czmiel, Alexander. 'XfOS (XML for Overlapping Structures)'. *ACH/ALLC 2004*. Göteborg, Sweden, 2004.

Di Iorio, Angelo, Silvio Peroni and Fabio Vitali. 'Towards markup support for full GODDAGs and beyond: the EARMARK approach'. *Balisage: The Markup Conference 2009*. Montréal, Canada, 2009. http://www.balisage.net/Proceedings/vol3/html/Peroni01/BalisageVol3-Peroni01.html.

Durusau, Patrick, and Matthew Brooke O'Donnell. 'Coming down from the trees: Next step in the evolution of markup? (Presenting JITTs, Just-in-time Trees)'. *Extreme Markup Languages 2002*. Montréal, Québec, 2002. http://www.durusau.net/publications/NY_xml_sig.pdf.

Huitfeldt, Claus. *Markup Languages for Complex Documents (MLCD).* http://decentius.aksis.uib.no/mlcd/en.htm.

Image Markup Tool. University of Victoria (Martin Holmes). http://tapor.uvic.ca/~mholmes/image_markup/.

Lancashire, Ian (1995). *Early Books, RET Encoding Guidelines, and the Trouble with*

SGML. `http://www.ucalgary.ca/~scriptor/papers/lanc.html`.

LMNL wiki. `http://www.lmnl.org/wiki/index.php/Main_Page`.

McGann, Jerome (2004). 'Marking Texts of Many Dimensions'. *A Companion to Digital Humanities*. Susan Schreibman, Ray Siemens, John Unsworth (ed.). Oxford: Blackwell.

NINES project. University of Virginia (Jerome McGann, et al.). `http://www.nines.org`.

Piez, Wendell (2001). 'Beyond the Descriptive vs Procedural Distinction'. *Markup Languages: Theory and Practice*. **3(2)**. `http://www.piez.org/wendell/papers/beyonddistinction.pdf`.

Ramsay, Steven (2008). 'Algorithmic Criticism'. *A Companion to Digital Literary Studies*. Susan Schreibman and Ray Siemens (ed.). Oxford: Blackwell. `http://www.digitalhumanities.org/companionDLS/`.

Sperberg-McQueen, C. Michael (1991). 'Text in the electronic age: Textual study and text encoding, with examples from medieval texts'. *Literary & linguistic computing*. **6(1)**: 34-46.