Classifying the Chimera

Federico Meschini *(fmeschini@tin.it)*
Tuscia University

While the *Digital Library* (*DL*) concept is an extremely vague one, its paradigm, the real implementation, is, if possible, still more elusive.

Since the late 80s and during the 90s, the primary concern and task in the textual digital resources field was really a basic (but not simple) one: how can we put texts in computers? How can we encode, manage and memorize them? Which stuff digital texts are made of?

Other important problems were perceived (visualization, for example) but they were temporarily put aside for practical reasons. The Text/Data relationship is really a Castor and Polydeuces' one: is Text a particular kind of Data or Data a particular kind of Text? The main choice, and long-term winning, in the textual encoding issue was the use of the powerful *Standard Generalized MarkUp Language* (*SGML*), and – more specifically – the rules (or better *guidelines*) established by the Text Encoding Initiative (TEI), which could actually be considered the de facto standard for encoding humanistic texts in digital format.

The transition from SGML to the *eXstensible MarkUp Language* (*XML*) was, more than an evolution, a sort of Copernican revolution for two main aspects: the introduction of the new stylesheet technology for displaying an XML file, in particular the powerful *Extensible Stylesheet Language Transformation* (*XSLT*), and the great diffusion and development of open-source software able to manage XML documents. This transition has of course also taken place in the TEI: starting with the P4 version of the Guidelines, XML is the technology now used. As a logic result different open-source software tools for implementing a Digital Library with texts encoded in TEI/XML are now available.

These tools are divided into two main categories. In the first group there are software created in hard Information Technology contexts (Database Management Systems, Native XML Databases, Publishing Framework, etc.), and these programs need to be adapted to the specific aims of a digital library[1]. In the second, and this is a new trend, these tools are being developed in academic contexts[2], created from the scratch, or using programs from the first group as the basic core; in both cases the final function is the utilization with digital cultural resources and overall TEI encoded texts. Compared to the first group, the logical added value of these tools is that often they provide some specific features for the textual field[3].

The choice is now much wider than it was just a few years ago, but it could also cause some confusion in selecting the right tool. As the experts of knowledge management well know, too much information, if not well structured and organized, is equivalent to no information at all, thus being completely useless. Each software has its own peculiarities, which should be evaluated and confronted against the characteristics of the texts being encoded and the general needs and aims of the project of which the digital library is part. What is the main aim of a project? It's the visualization one, with perhaps a multiple output feature? It's the research, with some form of advanced textual analysis? It is possible to combine all these aspect? And if somebody has already found a solution for our problems how can we find it over the net?

Trying to find a solution for these problems, or better, trying to share the same problems to have common solutions, during the TEI Members Meeting 2003, there was the first reunion of the TEI Presentation Tools Special Interest Group[4]. The Presentation Tools SIG has two main initiatives: the creation and update of a tool list and of a sample collection of texts for testing.

The first version of the tool list has been presented during the TEI Meeting in Baltimore, last October. This list is actually a digital document encoded with the TEI/XML standard, and in this way it's currently published, using an XSLT stylesheet, in HTML[5]. With a simple structure this list presents the various software in an alphabetical order with a short description and the links to the various implementations. From the descriptions and the links it's possible to have an idea of the distinctive features of each tool, that for example the software *XPhilologic* is very good for full-text search and document retrieval[6] and that *Apache Cocoon* could be implemented so to have an XML framework for format scalable output of the same TEI document[7] And again, with *Anastasia* is possible to have an electronic text/image edition of a medieval manuscript[8] and *eXist* is a powerful native XML database that could be used for queries and researches.[9]

But this is not enough. Perhaps from a list you can obtain some information, but what is really needed (and planned since the beginning) it's an higher level of classification, and this become more necessary as the number of such software is increasing[10]. It's a software written in Java or in Perl? Which are requisites for running it on a computer? It's XML-aware? It allows XSLT transformation? The texts are stored in the file system or in a database? What are its peculiar features? It could be integrated with other software in order to augment the possibilities? It can be customized? It's clear that a simple list cannot answer to all these questions.

Many discussions have been made about the kind of classification to apply to the tool list and in my opinion it should be made using practical rather than theoretical principles, with a sort of empirical and pragmatic observation, including also the links to the most possible numbers of the concrete implementations of these tools, so to highlight the best practices and the particular features of each digital library.

A good way of realizing this classification could be the use of the standard ISO 13250[11] or TopicMaps, and the respective *XML Topic Map* (*XTM*) syntax[12]. A TopicMap is based on the definition of a general topic, the particular and real occurrences of that topic, and the associations between different topics, thus in my opinion it's the best way to obtain a complete classification, which will include the various aspects, from the most technical, concerning the programming languages used or the technical specification needed, to the functionalities of visualization, text research and analysis.[13]

So what is now a TEI document should be elaborated in a XTM document, detecting, separating, organizing, linking and classifying all the information that now are presented in a linear structure.

Once created, the XTM file representing the TopicMap can be used and navigated in several ways. Being an XML file, it is possible to apply the same technologies used for the TEI texts, but there are also available some dedicated software which can exploit the great potentialities of this standard as, for example, the *Omnigator* from *Ontopia*[14], or the *TM4J*[15], a java open-source package expressly developed for creating, manipulating and publishing topic maps.

The TopicMap technology has been presented for the first time related to the TEI during the 2003 meeting, and it's growing in interest from this community, for its possibility of adding a metadata semantic layer to the digital collections[16]. Moreover, thanks to the possibilities of merging different XTM documents each representing a different map, the Presentation Tools TopicMap could be integrated with other map about other subjects, the textual content for example[17] or the documentation of the local views of the DTDs[18], thus creating the basis for the definition of what could become a 'TEI Ontology'.

[1] E.g., *Apache Cocoon* <http://cocoon.apache.org/>, *Apache AxKit* <http://axkit.org/>, *eXist* <http://exist.sourceforge.net/>

[2] *Anastasia* <http://anastasia.sourceforge.net/>, *teiPublisher* <http://teipublisher.sourceforge.net/docs/index.php>, *XPhilologic* <http://barkov.uchicago.edu/xphilo/>

[3] See for example the *TAPoRware* set for textual analysis <http://cheiron.mcmaster.ca/~taporware/> or the *Versioning Machine* <http://mith2.umd.edu/products/ver-mach/> for the comparison of different versions and editions of the same text.

[4] <http://www.tei-c.org/Members/2003-Nancy/mm17.html#tap-sig>

[5] Available on line at <http://miro.acs.its.nyu.edu/tei_cms/show.php>.

[6] See for example the demo on the *Brown Writer Women Collection* at <http://barkov.uchicago.edu/xphilo/search.brownwwp.html>.

[7] A good implementation of *Cocoon* with TEI can be found at <http://www.nzetc.org/>.

[8] See the *Caxtons' Canterbury Tales* at <http://www.cta.dmu.ac.uk/Caxtons/>.

[9] See the *Digital Quaker Collection* at <http://esr.earlham.edu/dqc/>.

[10] See the presentations on this subject at ALLC/ACH 2004 <http://www.hum.gu.se/allcach2004/AP/>. Among the others: Kumar, Amit et al., *teiPublisher a repository management system for TEI documents* <http://www.hum.gu.se/allcach2004/AP/html/prop118.html>; Matthew Zimmerman, *Using AMP technology (Apache, MySQL, PHP) for XML publication* <http://www.hum.gu.se/allcach2004/AP/html/prop156.html>; Stephen Ramsay, Geoffrey Rockwell, Stéfan Sinclair, *TAPoRware: Simple Portal Tools for Text Analysis* <http://www.hum.gu.se/allcach2004/AP/html/prop136.html>

[11] <http://www.isotopicmaps.org/rm4tm/>

[12] <http://www.topicmaps.org/xtm/1.0/>

[13] For an introduction to TopicMap see Steve Pepper, *The TAO of Topic Maps, finding the way in the age of infoglut* <http://www.gca.org/papers/xmleurope2000/papers/s11-01.html>.

[14] <http://www.ontopia.net/omnigator/models/index.jsp>

[15] <http://tm4j.org/>

[16] John Bradley, "A Model for Text Analysis Tools" <http://llc.oupjournals.org/cgi/content/abstract/18/2/185>

[17] See John A. Walsh, "Topic Maps and TEI-Encoded Literary Texts", <http://drh2004.ncl.ac.uk/abstract.php?abstract=177>

[18] Stuart Brown, "A Topic Map for the TEI" <http://www.tei-c.org/Members/2003-Nancy/index.html#SB-abs>

Bibliography

Anastasia. Accessed 2005-05-19. <http://anastasia.sourceforge.net/>

Bradley, John. "A Model for Text Analysis Tools." *Literary and Linguistic Computing* 18.2 (2003): 185-207. Accessed

2005-05-19. <http://llc.oupjournals.org/cgi/content/abstract/18/2/185>

Brown, Stuart. "A Topic Map for the TEI." TEI Consortium, 2003. <http://www.tei-c.org/Members/2003-Nancy/index.html#SB-abs>

Kumar, Amit, et al. "teiPublisher a repository management system for TEI documents." Paper delivered at the ALLC/ACH 2004 Conference, Göteborg. 2004. Accessed 2005-05-19. <http://www.hum.gu.se/allcach2004/AP/html/prop118.html>

Pepper, Steve. "The TAO of Topic Maps, finding the way in the age of infoglut." Paper delivered at the XML Europe 2000 Conference, Paris. 2000. Accessed 2005-05-19. <http://www.gca.org/papers/xmleurope2000/papers/s11-01.html>

Ramsay, Stephen, Geoffrey Rockwell, and Stéfan Sinclair. "TAPoRware: Simple Portal Tools for Text Analysis." Paper delivered at the ALLC/ACH 2004 Conference, Göteborg. 2004. Accessed 2004. <http://www.hum.gu.se/allcach2004/AP/html/prop136.html>

TAPoRware. Accessed 2005-03-11. <http://cheiron.mcmaster.ca/~taporware/>

Versioning Machine. Accessed 2003-12-09. <http://mith2.umd.edu/products/ver-mach/>

Walsh, John A. "Topic Maps and TEI-Encoded Literary Texts." Paper delivered at the Digital Resources for the Humanities Conference, Newcaslte Upon Tyne. 2004. Accessed 2005-05-19. <http://drh2004.ncl.ac.uk/abstract.php?abstract=177>

XPhilologic. Accessed 2005-05-19. <http://barkov.uchicago.edu/xphilo/>

Zimmerman, Matthew. "Using AMP technology (Apache, MySQL, PHP) for XML publication." Paper delivered at the ALLC/ACH 2004 Conference, Göteborg. 2004. Accessed 2004. <ttp://www.hum.gu.se/allcach2004/AP/html/prop156.html>

teiPublisher. Accessed 2005-05-19. <http://teipublisher.sourceforge.net/docs/index.php>