Dixit Convention II

Digital editing beyond XML: an introduction

Fabio Ciotti

Digital scholarly editing theory and practice during the last years have been substantially based on XML formalism and its hierarchical data models, particularly for the great influence of the TEI encoding scheme and Guidelines. But this predominance has always raised a lot of critical or polemic reaction in the Digital Humanities community, for many different reasons.

The title of this workshop in the original DiXiT work-plan was “XML free digital editing”. When I have inherited the responsibility for the organization of this event, taking the role of Supervisor and Responsible of Roma Sapienza unit, I have proposed a little change. I wanted to change the title because I felt that stressing the mere fact that it is possible, desirable or even right to have a non XML based editing project is not that novelty. In fact it I well known that scholarly digital text archiving (and analysis) started in the early '70s of last century on the base of formalisms like MicroOCP/Cocoa, TLG Beta code, ARTFL encoding, and that the very first digital framework for scholarly editing, Tustep , was not at at based on SGML.

Even the “theoretical/ideological” controversy pro and against SGML/XML is very old. One of the most fierce adversary of the “SGML turn” in digital textual editing was Ian Lancashire, English Literature scholar and creator of one of the more famous text analysis tool ever, TACT. If we take a look at a paper he wrote in 1995 entitled “Early Books, RET Encoding Guidelines, and the Trouble with SGML” (Lancashire 1995), we can find almost all the fundamental arguments and critical positions expressed against TEI and XML in the last 30 years. Even more drastic is the judgment that five years before Paul Fortier expressed, as a member of the original Text Encoding Initiative Literature Working Group, against one of the rationales of the adoption of SGML in the definition of the TEI encoding scheme (Fortier 1991):

> My perspective is that coding (inputting or converting text) is not the same as interpreting. Descriptive coding as presented in the Guidelines is squarely in the domain of interpretation. Scholars do not want interpreted texts; they expect to do that job themselves. When possible scholars hire assistants to input texts, and do not expect these assistants to do the interpretation. This whole aspect needs to be brought into conformity with scholarly practice, otherwise the TEI standards will not be respected.

Similarly we could remember the more technically nuanced critics raised by Mark Olsen, one of the creator of ARTFL text base, analysis of the TEI in its “Text theory and coding practice: assessing the TEI” (Olsen 1996):

> The Text Encoding Initiative (TEI) is caught between two vitally important yet somewhat contradictory mandates. The editors of the TEI are writing a data interchange

> format while at the same time working out a mechanism to support theoretically informed encoding specifications for just about any textual object that scholars in a wide variety of disciplines might encounter. Unfortunately, the resulting drafts of the TEI specification(s) reflect this underlying confusion of the task at hand. It is far too variable and flexible to be a usable data interchange format while being informed by theoretical models that have been subject to considerable debate over the last several years.

And how could we forget the venerable overlapping structures/markup polemic (DeRose 2004; Barnard et al. 1995)? At the times of this writings XML was still in the mind of the gods, and TEI was based upon its predecessors SGML, but we can safely say that none of the differences between SGML and XML affect or undermine the severe arguments of these early and eminent critics.

The fact is that XML and the TEI actually did win the war, at least on a pragmatic basis. Why? I think that there are many theoretical, pragmatical and social reasons for this, and I think they are all well founded and sound:

- XML is relatively easy to learn and use compared to other digital technologies, especially if the complexity level of the encoding is low or medium;
- the act of encoding is very proximal to the practice of annotation that is very familiar to the average humanist;
- XML data format is sufficiently portable (especially in the editing phase) between different platform;
- XML processing leave to the user a lot of control on the editing process and on the resulting data;
- XML permits a good level of quality control via its internal syntax and schema based parsing facilities;
- XMLis flexible enough to accommodate a vast range of humanistic users requirements (of wich I feel that the one of interoperability is not in ranking so high in the humanities scholars minds);
- XML now lives in a pretty good ecosystem of related standards and open source applications that make very easy the whole process of a any set of digital scholarly resource.

On the other hand, it is worth pointing out that, even if it is true that the TEI was born as an SGML application, and then turned into an XML based language, its evolution has somewhat led to a sort of abstraction from the language that is actually used as its encoding format and, to some extent, form its underlining hierarchical data model.

We must remember, in fact, that a neat distinction in the usage of XML can be drawn: XML can be adopted as a formal modeling language, in which case we accept the underlying tree based data model as a good way to formally represent the domain of interest. But XML can be used as a mere syntax facility, a serialization language that is independent from the actual data model we are using to represent our domain: this is the normal stance adopted in the definition of the XML syntax of modeling languages like RDF or OWL or Topic Map, which are based on graph data models, or in

the mapping of relational table into XML docs. Actually in the TEI we can find a lot of not (mono)hierarchical features. So, in a sense, TEI is slowly moving from a modeling orientated usage of XML to a syntactic oriented usage of XML.

With this apodictic observation I go back to my explanation about why I decided to change the “XML free” phrase of the original title of this workshop to the actual “beyond XML”. I think that even if it is true that we have had in the past and have still now a lot of technical alternatives to XML, most of these alternatives are no real alternatives, either because they are pragmatically too complex or application dependent, or simply outside of the horizon of expectations of the digital editing community; or because they are theoretically equivalent in term of representational capacity to XML or even weaker than XL in modeling the complexity of humanities objects. It's time to think beyond XML, not against, and let people live with XML for all the tasks for which it is still the better choice, *coeteris paribus*.

One last remark. This call to go beyond is not relevant only for the debate on XML/non XML textual markup, or for the scholarly editing community. I think it is the central theme for the Digital Humanities community at large, and for the future development of our field. The big tent has really grown very big, maybe even too big, and under its covering we see a lot of putative innovative projects, a big mess of tools or services or resources that should radically innovate the Humanities, and are fundamentally based on a small set of fundamental enabling technologies that have at least 40 years: declarative and grammar based markup, relational database, networking environments, information retrieval. It seems that, despite its theoretical assumptions, Digital Humanities has not been able to contribute to its own foundational formalisms and methods. Maybe it is time to go beyond.

References

Barnard, David T, Lou Burnard, Jean-Pierre Gaspart, Lynne A Price, C. M Sperberg-McQueen, e Giovanni Battista Varile. 1995. «Hierarchical encoding of text: Technical problems and SGML solutions». *Computers and the Humanities* 29 (3): 211–31. doi:10.1007/BF01830617.

DeRose, Steven J. 2004. «Markup Overlap: A Review and a Horse». In *Proceedings of Extreme Markup Languages*. Montréal, Québec.

Fortier, Paul. 1991. «The TEI Guidelines (Version 1.1 10/90). A critique». http://www.tei-c.org/Vault/AI/ai3w05d.txt.

Lancashire, Ian. 1995. «Early Books, RET Encoding Guidelines, and the Trouble with SGML». novembre 11. http://homes.chass.utoronto.ca/~ian/calgary.html.

Olsen, Mark. 1996. «Text Theory and Coding Practice : Assessing the TEI». Paper at the Joint Annual Conference of the Association for Computers and the Humanities and Association for Literary and Linguistic Computing 1996. Bergen, Norway.