The Role of the Text Encoding Initiative (TEI) in the Authoring and Interchange of XML Documents

Lou Burnard[1] and Sebastian Rahtz[1]

Oxford University Computing Services
13 Banbury Road, Oxford OX2 6NN, UK
http://www.tei-c.org.uk/

Abstract. The Text Encoding Initiative's Guidelines for electronic text encoding and interchange original aim was to provide exhaustive recommendations for the encoding of literary and linguistic textual materials. In this paper we suggest that the Guidelines are also suitable for the authoring of new material, such as websites and academic papers. With only slight extensions, they can be used to describe the structure and contents of a website, while their modular architecture readily permits customization for multimedia and hyperlinking behaviours. Despite their 'book-centric' nature, the Guidelines are sufficiently flexible to cater for information organized in a completely non-book-like manner.

We argue that its non-prescriptive nature makes the TEI a good candidate for mapping amongst other possibly more prescriptive XML schemes. As subject-focussed XML-based schemas proliferate, there will be an increasing demand for a non-specific interlingua for the interchange and integration of digital textual data. Our claim is that its extensive coverage, pervasive take-up, rigorous documentation, and well-thought out class mechanism uniquely qualify the TEI for this key role.

1 Theoretical Preliminaries

We begin with what might be called a 'reality check': what (exactly) is the purpose of introducing markup into a digital resource, whether it is a representation of a pre-existing document, or the creation of an entirely new one?

It's commonplace in the literature[1] to assert that markup makes explicit, and thus processable, a theory about (or interpretation of) the pre-existing document, for the benefit of others. This assertion is often made as a defence against the criticism that in producing a digital representation of some pre-existing document, the encoder necessarily corrupts or reduces it in some essential respect. On the contrary, it is claimed, the essence of a document is precisely that which its markup identifies: markup is not imposed on a text, but rather reveals it. Since much of the practice of scholarship is devoted to the establishment of consensus as to what exactly the constituents of such textual readings should be; markup is simply a particularly useful way of recording that consensus in a given case.

What, however, of new documents, in which (it seems reasonable to assume) the author has definitive knowledge as to what the constituents are, or are intended to be? As Sperberg-McQueen et al 2000 point out, markup can also be regarded as *performative* rather than descriptive of an interpretation: when I insert a `<p>` element into my document as I type, I *ipso facto* insert a paragraph, as far as any reader familiar with the semantics of the `<p>` element is concerned: no interpretative act is needed, beyond that needed to decode the writing itself[2]. What, however, if I insert a `<garble>` element? What would be necessary for other users of my document to know what I intended?

This may seem a frivolous or overly academic question. However, it seems more than likely that the availability of a powerful and general metalanguage such as XML, with all the abilities it gives for the definition of arbitrary encoding languages, thereby also increases the risk of a new babel, and of the fragmentation of the digital community. If historical records are encoding using a 'historical markup language', linguistic data using a 'linguistic markup language' and illustrations using a 'visual markup language', how will we encode a document in such a way that the language of historical illustrations can be studied? Worse, how will we choose amongst the different options each of these languages is likely to offer for concepts common to all of them?

This is of course by no means a new problem: it is, indeed, the central issue in all standardization efforts, (as it is, arguably, for language itself). How can one consolidate the known, agreeing on common vocabularies for things agreed to be the same, while at the same time permitting for the addition of the new and customization for the specific? In this paper we describe how the TEI scheme attempts to answer this question and attempt to assess its success in so doing with reference to one particular unanticipated application.

---

[1] See amongst many other examples Renear 1995, Robinson 1996, Burnard 2001

[2] Purists will point out that the decoding of writing is itself equally an act of interpretion. But it is made with reference to a shared semiotic system to which reader and writer alike subscribe on equal terms.

2 The TEI and its Development

In November 1987, representatives from an extraordinarily wide range of different institutions worldwide met at Vassar College in upstate New York for a meeting funded by the US National Endowment for the Humanities and sponsored by the ACH. Its objective was to debate the possibility of providing standardized 'guidelines' for the production of electronic texts. (Burnard 1988). The proceedings were intensive, and the debate laid the foundation for an ambitious project which would, over the next few years, have a major influence on the developments of digital libraries, language corpora, and scholarly data sets in general.

Between 1990 and 1994, the Text Encoding Initiative operated as a major international research project involving over a hundred distinguished scholars world wide. It obtained major funding in both Europe and North America for an extensive programme of consultation and debate, which resulted in publication of a detailed and very broad-based taxonomy of text encoding practices. Its goals were to facilitate the interchange and integration of scholarly data in the broadest sense, providing support for all texts, in all languages, from all historical periods. Achievement of these goals was seen as requiring two somewhat contradictory subgoals: firstly the provision of 'guidance for the perplexed', that is, specification of which features of text should in general be represnted in a digital text; secondly the provision of assistance to the specialist, by defining general encoding principles which could be applied to as-yet undefined encoding requirements. The project thus sought to bring about both a user-driven codification of existing best practice and a framework into which unpredictable extensions could be fitted.

The TEI as a research project came to an end in 1994, with the publication of its major deliverable, the two volume 1700 page reference work known as P3, the TEI Guidelines (TEI, 1994). This work may be thought of as a comprehensive way of looking at what texts are and how to organize them. It is expressed as a very large set of over 600 element definitions, organized as a modular set of DTD fragments, together with a mechanism for customization, extension, and specialization of the set. The TEI and its users then set about the evangelical task of producing tutorial guides, applying the recommendations in practice and codifying shared practice in the rapidly expanding fields of digital libraries and text collections, language corpus building, electronic editing and similar projects.

Wide adoption of the Guidelines has brought about a demand for continued support and maintenance of the document itself, while the rapid proliferation of new XML-based vocabularies has also suggested areas in which its coverage might be expanded. After a sustained period of negotiation and debate, the TEI was reconstituted as a not-for-profit membership consortium, incorporated in the Commonwealth of Virginia, with executive offices in Bergen, Norway. A first meeting of its members, held in Pisa in November 2001 held elections to a new TEI Scientific Council, which met for the first time in London, January 2002. This Council

approved a new scientific programme, to include as a priority the completion of a new XML-based edition of the Guidelines and this new edition, known as P4, appeared in print in June 2002 (TEI, 2002).

With a stable membership base (over 50 institutions and individuals joined the new Consortium worldwide during its first year) and a new tranche of research funding, the TEI Consortium has been able to initiate a new work programme. One major priority is to catch up on technical developments since publication of P3: hence one work group has been chartered to address character encoding issues, in particular how best to make use of Unicode in the TEI community; while another is addressing issues of 'stand-off' markup and hyperlinking in the context of new W3C standards such as XLink. The TEI Technical Council has also set up a work group to address problems of migrating large collections from an SGML environment to XML, and another to investigate and make recommendations on TEI training services.

How successful the new TEI Consortium will be in attracting and retaining members remains to be seen. In scope and depth of coverage there is still little to compare with the sophistication of the original Guidelines and so it seems probable that, as long as people want to use a general encoding scheme of this kind, they will be using some version of the TEI. The question posed by this paper is however to ask whether or not the Guidelines have a slightly different potential role: as an interlingua or *lingua franca* mediating among an ever-increasing number of special-purpose encoding vocabularies.

3 The Scope of 'intelligent' Markup

We claimed above that the function of markup was to represent a reading of a text. Since there may be almost as many readings as there are readers, and certainly as many different kinds of readings as there are purposes for reading, it must be an ultimately hopeless task to delimit encyclopaedically the scope of markup. Thinking solely of the kinds of activities and readings common in the field of textual scholarship, we might reasonably expect a markup scheme to be able to support a diplomatic or orthographic transcription of an original source, together with links to images of that original source, preferably aligned at textually significant junctures. We might reasonably expect the markup scheme to identify and possibly disambiguate semantically significant components of the surface text such as proper nouns, dates, times, etc. (This 'named entity recognition' has recently become of particular importance in text retrieval and data mining applications, but has a long philological tradition). For purposes of linguistic analysis, we might also wish the encoding to convey morphological or other linguistic characterizations down to the individual word level, as well as to mark up higher level features of syntax or discourse coherence. For deeper exploration of the topics addressed we might

want to embed and to distinguish references or quotations from other sources, to add additional references to such sources, and of course to add whole layers of additional annotation of various kinds, ranging from meta-textual matters such as corrections or additions or comparative readings, to purely editorial commentary and discussion.

Is it reasonable to assume that a single encoding scheme could or should be able to handle all of this variety? It certainly seems desirable that a single common digital source can be used for multiple purposes: for editor and publisher alike, the ability to maintain a single common text, from which a variety of different kinds of edition can be extracted, is a common goal, all too frequently compromised by the exigencies of particular production problems. Another reason for pursuing the goal of a common multipurpose encoding is the observed existence of what we might call *Frequently Answered Questions*: that is, the fact that there is a relatively small number of technical problems which tend to re-appear in many different appplication contexts. For example, technical problems seemingly specific to implementation of a system in which transcribed text is aligned with digital page images have a striking similarity to the technical problems seemingly specific to an application which seeks to align transcribed text with an audio or video stream.

Because the TEI was designed for scholarly use, its design reflects a very generic perspective, in which all texts are alike, yet every text are different, and in which multiple views on the nature of the material are the norm rather than the exception. The technical details of the process by which the TEI achieves this goal are discussed in some detail both in the Guidelines and in several review articles (examples include Burnard 2000; Sperberg-McQueen et al 1995; Burnard 1993); briefly summarized, the key features are that the hundreds of elements defined in the TEI are organized into a small number of *modules*, which may be selected and combined according to a well-defined set of rules, and which may also be extended using a system of semantic and structural classes. Rather than recapitulate the details of this mechanism, we present in the next section a specific instance of its application: the definition of a TEI DTD appropriate to the needs of technical authoring.

4 The TEI as an Authoring DTD

To use the TEI for authoring may seem at first sight to contradict the aims of the TEI, in that an authoring application will normally seek to be *prescriptive* rather than *descriptive*: its object will be to give authors clear and unambiguous rules about what markup to use for particular purposes, rather than to offer them a wide and potentially confusing range of choices. Moreover, in most technical authoring environments, authors are explicitly discouraged from specifying the appearance of their text, but rather required to focus on the text's intended function, which

may be specified at quite a fine level of detail, thus potentially needing the ability to markup *ad hoc* objects for particular projects.

Does this rule out the TEI? On the contrary, we think the TEI provides all the categories we are likely to need in our tagging; plus the ability (by means of its subsetting facilities) to constrain authors to use only the distinctions which are needed; plus the ability to add new categories where necessary. The first of these is a consequence of the TEI's wide range of markup derived from a synoptic view of conventional publishing effects; the second and third are both provided by the TEI's customization mechanism, as further discussed below.

Let us consider, as an example, ways in which we might configure the TEI to use it for web-page authoring[3]Our web authors, like many others, are busy technically-minded people, who do not want to be offered a choice of inputting verse, or critical text apparatus. We began by defining a DTD which used the prose base module, with the figures and linking additional modules from the full TEI scheme, but omitting many unnecessary elements (such as `<gap>`, `<corr>` and `<orig>`). Since our object was primarily to produce XML for use on the web[4], it seemed desirable to streamline the way in which external entities such as graphics or hyperlinks are specified: we therefore modified the `<xref>` and `<xptr>` elements to allow a *url* attribute, and also modified the `<figure>` element to allow additional *width*, *height* and *file* attributes.

Customizing a scheme such as the TEI is not simply a matter of defining a modified subset however. It is also necessary to add new specialized elements: for example, in a technical documentation environment, it may be convenient to use more specialized tags than the TEI provides for identifiers of various kinds, so that authors may distinguish, for example, `<fileName>`, `<URL>`, `<variableName>`, or `<reservedWord>` elements. This might be done simply by using the existing (already rather overworked) TEI type or rend attributes on the generic `<ident>` element provided in the TEI DTD, but for authors of such material, it would be far preferable to use distinct tags. In the same way, we may need to define some entirely new additions to the inline markup class to describe, for instance, modern artefacts like sound and video clips.

Many editing tools, designed to facilitate the production of well-formed and syntactically correct markup, are now available. Such tools (Xmetal, for example) greatly facilitate the production of marked up documents, for example by offering visual feedback to represent the function of elements selected, by constraining the choice of elements at particular contexts, and so forth. To use such tools to the

[3] For the most part the changes discussed here derive from our experience in setting up the Oxford University Computing Services web site at www.oucs.ox.ac.uk, which is now authored entirely in TEI XML.

[4] In fact, however, several of our pages are reformatted for print delivery using a variety of XSL-T style sheets.

full a well-tailored and specialized DTD is essential. However, the more specific a DTD becomes, the more difficult it becomes to ensure compatibility and interchange. The TEI addresses this problem in two ways: firstly, it requires that new elements be classified according to the existing structural classes before they can be used; secondly, it provides, via an architectural form attribute, a way of stating an underlying TEI generic element for each more specific element used.

Each of our specialized forms of the TEI `<ident>` element is declared as a member of the data model class, thus constraining their appearance to an appropriate point in the model without the need to redeclare any other content models. Each of them also states that it is a specialization of the underlying `<ident>` element by virtue of supplying that string as the value of its teiform attribute. In this way, a TEI conformant application when presented with a document in which such otherwise unknown elements appear can apply fallback procedures appropriate to the `<ident>` element. A TEI text analysis program is unlikely to know (or care) how to process a `<variable>` or a `<reservedWord>` as such, but it can usefully process it as if it were an `<ident>`. In this way, the TEI scheme allows us to combine the virtue of extensibility with that of generality.

The resulting OUCS DTD is a view of the TEI with a mere 128 elements (of which many are in the `<teiHeader>` element, which our authors generally ignore). We provide a simple CSS stylesheet within Xmetal which provides visual feedback as material is written, and a range of different XSLT stylesheets to render the material on the web in various ways (for example, to provide views of the same page appropriate to the needs of visually impaired users). For the purposes of the present discussion however, the key point is that those elements we have adopted unchanged from the TEI are used according to documented syntax and semantics, while, for those elements which we have modified, their relationship to the TEI standard scheme is preserved unambiguously. Our authors see all and only the elements they need and yet our documents can still be transparently interchanged with other sites using the TEI scheme.

5 Conclusion: What is a DTD for?

Using a DTD of this comparatively restricted kind has obvious advantages at data preparation time (e.g. to enforce consistency), but is somewhat redundant at other times, since if a document is well-formed, its DTD can be (almost) entirely automatically recreated from it. Moreover, since DTDs allow very little in the way of content validation (as opposed to structural validation), one might reasonably ask even whether this limited value is worth the effort. With the arrival of XML Schema languages in which more sophisticated content validation becomes possible, it seems increasingly likely that DTDs will disappear as validation tools.

However, the abbreviation DTD has two distinct expansions in the SGML standard: it may be regarded as short for document type *declaration*, or for document

type *definition*. To quote the standard itself: **(document) type declaration:** A markup declaration that formally specifies a portion of a document type definition... A document type declaration does not specify all of a document type definition because part of the definition, such as the semantics of elements and attributes cannot be expressed in SGML.[5]. Corresponding to this distinction, markup languages generally require distinct sets of documentation: firstly, a set of SGML (or XML) declarations, catalogs etc, suitable for immediate use with document instances; and secondly, associated descriptions which say what the elements are intended to represent, and gives examples of their use. Of course, the DTD files may contain comments, but in general the model is that of traditional computer programming, which regards documentation as a separate (and often later) stage from writing code.

The TEI developed differently, partly because it did not start specifically as a project to write SGML DTDs; the initial effort concentrated on abstract models of markup, instantiated (for want of anything significantly better) using SGML. Crucially, the TEI scheme is itself defined in a small SGML-based markup language, rather than directly writing DTD code. The scheme follows the 'literate programming' WEB model developed and popularized by Donald Knuth (Knuth, Donald E.) in which a single document contains both formal code and documentation for that code; this single source can then be processed to produce multiple outputs. The TEI literate programming system (jocularly named ODD, for One Document Does it all) as originally specified (`http://www.tei-c.org/Vault/ED/edw29.sgm`) underwent several modifications during the process of implementing and using it for production of the TEI Guidelines: a paper presented at the ALLC-ACH Conference in Paris in 1994 describes the production system more fully (Sperberg-McQueen 1994)[6]

For our present purposes, the key point is that the TEI scheme is essentially a set of definitions, rather than a set of declarations. In work reported elsewhere, we have demonstrated how the ODD approach facilitates the automatic production of a version of the TEI scheme expressed using an XML schema language such as Relax NG (Rahtz 2002). Here we suggest that this approach also facilitates the use of the TEI as an interlingua.

Wherever possible, the TEI defines generic classes of textual object in preference to specific ones: for `<div>`, `<ab>`, `<seg>` rather than chapter, paragraph, metaphor. As we have seen, this may frequently lead the user to create more specific versions of the generic element. If this is done using the the TEI modification mechanism, then the generic information is still available for use. Software encountering markup such as the following: `<metaphor TEIform="seg">`fresh ideas`</metaphor>` can operate

[5] ISO/IEC 8879-4, clause 4.103, cited from Goldfarb 1990, p. 264.

[6] Other XML/SGML-based literate programming schemes are summarized at http://xml.coverpages.org/xmlLitProg.ht.

at either the generic or the specific level. In this sense, the document type definitions underlying the TEI provide a very real opportunity for use as a markup interlingua. Its descriptive nature enables it to encompass other more prescriptive markup schemes.

The TEI is a well-known reference point: because it is stable, and pervasive, its users can readily share data and resources, and modular software development, and can also expect to benefit from a shallower learning curve and reduced training costs, as well as benefiting from the well-known advantages of open source industry-independent standards.

More significantly, we suggest, because the TEI is also rigorous, and well-documented, it has significant potential as an interlingua. Ultimately, we suggest, the TEI is not just about exchanging data between computer systems dealing with literary and linguistic markup. It also provides a language for communication between human beings working in different markup languages.

References

1. Burnard, Lou "Report of Workshop on Text Encoding Guidelines" in *Literary and Linguistic Computing*, vol. 3 no 2, pp. 131-3. Oxford: Oxford University Press 1988.
2. Burnard, Lou "Rolling your own with the TEI" in *Information Services and Use* vol. 13 no 2, pp. 141-154. Amsterdam: IOS Press 1993.
3. Burnard, Lou, Claudia Claridge, Rainer Siemund, and Josef Schmidt "Encoding the Lampeter Corpus" in *DRH98: selected papers from Digital Resources for the Humanities* London: Office for Humanities Communication 2000.
4. Burnard, Lou. "On the hermeneutic implications of text encoding" in *New media and the humanities: research and applications*, ed. D. Fiormonte and J. Usher, pp. 31-38. Oxford: Humanities Computing Unit 2001.
5. Goldfarb, Charles F. *The SGML Handbook* Oxford: Oxford University Press 1990.
6. Renear, Allen, David G. Durand, and Elli Mylonas "Refining our notion of what text really is: the problem of overlapping hierarchies" in *Research in Humanities Computing* ed Nancy Ide and Susan Hockey, Oxford: Oxford University Press 1996.
7. Knuth, Donald E. *Literate Programming*, Stanford University Center for the Study of Language and Information (CSLI Lecture Notes Number 27), Stanford, CA: 1992.
8. Rahtz, Sebastian and Lou Burnard "Converting to schema: the TEI and RelaxNG" (Paper presented at XML Europe 2002)
9. Robinson, P.M.W. "Is there a text in these variants?" in *The literary text in the digital age* ed R.J. Richard. Ann Arbor: University of Michigan Press. 1996
10. Sperberg-McQueen, C.M. and Lou Burnard "The ODD System of Tag Set Documentation" in *Consensus ex machina?* (Resumés du colloque internationale: Laboratoire "Lexicométrie et textes politiques", École Normale Supérieure de Fontenay-Saint Cloud, 1994, pp. 221-222.)
11. Sperberg-McQueen, C.M. and Lou Burnard "The design of the TEI encoding scheme". *Computers and the Humanities* 29.1 17-39. Rpt. in *The Text Encoding Initiative: Background and Contexts*, ed. Nancy Ide and Jean Veronis. Dordrecht, Boston: Kluwer Academic Publishers, 1995

12. Sperberg-McQueen, C.M., Claus Huitfeldt and Allen Renear "Meaning and interpretation of markup". *Markup Languages: Theory and Practice*, 2.3 215-234. Also available from `http://www.w3.org/People/cmsmcq/2000/mim.html`.
13. TEI *Guidelines for Electronic Text Encoding and Interchange (TEI P3)* Sperberg-McQueen, C.M. and Lou Burnard, eds. Chicago, Oxford: ACH-ALLC-ACL Text Encoding Initiative, 1994.
14. TEI *Guidelines for Electronic Text Encoding and Interchange (TEI P4)*. Sperberg-McQueen, C.M., Lou Burnard, Steve DeRose, and Syd Bauman, eds. Bergen, Charlottesville, Providence, Oxford: Text Encoding Initiative Consortium, 2002.