PREPRINT

APPEARED in VGI cahier '97.

PLEASE REFER TO THE PUBLISHED VERSION ONLY.

TOWARD A UNIFORM REGISTRATION OF TEXTUAL SOURCES IN THE HUMANITIES

TWO DUTCH ENCODING PROJECTS USING THE TEI GUIDELINES

Hans Brandhorst, Peter van Huisstede Arjan Loeffen Frans Wiering

(ICS, Utrecht University)

Keywords: SGML, TEI, TMI, Music description, Printers devices, Emblem books

---

Contents

Introduction

Textual sources play a central role in many kinds of humanities research. Most research projects nowadays use a printed copy of the source text. When the source text is available in electronic form new, and largely unforeseen applications come to light. Electronic corpora and archives can be build to record and access a variety of textual phenomena, that may be primary (as in transcriptions) or secondary (as in critical annotations). These phenomena may be recorded explicitly in the source. For example, a missing fragment may be explicitly recorded as such. Others may be derived by inspecting the source texts. For example, a stage direction in a 18th century play may concern an unidentified actor or role, which can be automatically derived from the data as recorded. When the source texts are recorded electronically, dedicated programs can be build to access the text and offer an environment for selecting, viewing, analyzing and comparing text fragments.

In this the primary goals of textual research do not differ much from those that govern scholarly research on fixed data, as in classical, record-oriented database systems. Ultimately the data is recorded for retrieval, combination, and reporting -- in short, for deriving *new information* from it.

The wealth of database-oriented software applications in the humanities require the data to be *structured* in order to be able to extract required information from it. Similarly, text-oriented applications in this field require the text to be structured in order to serve as an information base for text-oriented research.

Sect. 1 Need for a text encoding language

For software to be able to process such textual information, distinct parts of the text should be *encoded*. The encoding may concern the physical or printed aspects of the text, its logical, narrative or linguistic structure, analytical insertions, corrections, and so on. This implies that regular encoding schemes as intrinsic to popular word-processors does not suffice; these focus on presentational aspects of the text, not the *information* encapsulated in the text. Also, the encoding strategy is *fixed* to this particular aspect of the text. Focussing on the most prominent requirements for a text encoding strategy, it should be

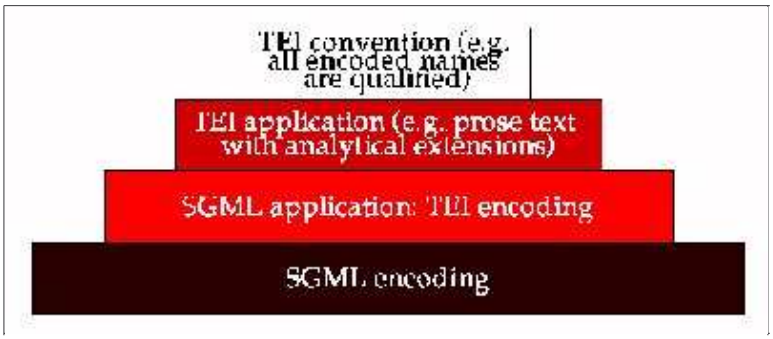
- *exchangeable*: several parties should be able to share their data and interpretations.
- *robust*: the material should be equally accessible now as at any moment in the future without having to convert the material to suit new working environments.
- *open*: the material should be saved in a form that is independent of the software used to access it in a particular project. Even within a single project several software products, with different goals, may require access to the same data (browse, print, collate, query, merge, edit, *et cetera*).
- *rich*: all information deemed relevant to the encoder should be encoded, and no software/encoding constraints should exist.
- *structured*: in encoding the texts relations between the sections should be recorded. Inherently, these relations should be subject to *validation*, i.e. invalid structures should be signalled.
- *extensible*: for each kind of text, and each kind of interpretation of the text an encoding scheme must be definable.
- *modular*: text in humanities projects are usually very large and must be manageable as a collection of

separate modules, or subtexts.

- *multimedial*: normally texts do not consist of characters only. Other media or structures, such as images and sounds, or external database objects, must be integrated within the source text.

- *hypermedial*: the text should allow for relations to be recorded explicitly between sections of the text(s) or any other medium. Such relations may be the result of a scholarly interpretation (e.g. thematic analysis), but also record relations made explicit or implied by in the source text (as in a reference to an illumination).

An essential consideration here is that such endeavours should be based on a common approach toward encoding the texts. This concerns four levels, depicted in the figure below. This layered construction will be explained in the following sections.



Consensus on encoding strategy is layered.

Sect. 2 Toward a common ground for encoding textual information: SGML

In order to make textual sources in the humanities accessible for structure- and text-oriented research software, and to record such information in a stable, exchangeable form, the choice of the *Standard Generalized Markup Language* (SGML) as both the data model and encoding language is attractive. The generic nature of the language, and the relatively simple data model it is founded on, opens the path to sophisticated research based on highly structured and enriched electronic textual sources. SGML is standardized as ISO/IEC 8879:1986 (A:1988) (Iso8879).

> *Note:* The standard is available in annotated form as Goldf90. Several books describe aspects of the language and related standards, see a. o. Bradl97, Alsch95, Colby96, Travi95, Maler96, Donov97.

It is an *encoding language*; most notably, 'tags' are placed within textual material in order to express the structural and informational aspects of the fragment identified within the complete text. Such a fragment is called an *element*. The occurrence of, and the relations between elements is governed by a *grammar* for the type of document encoded, which is called a *document type definition*, or DTD. For each document type a DTD must be defined; SGML only offers the tools for defining such a DTD -- it does not define these itself. The DTD, then, consists of declarations. Each element type occurring in the document is declared within the DTD. Such a declaration specifies what sub-elements may occur in element content, and what attributes can be specified for the element type. As such, a DTD is the product of thorough document structure analysis.

To understand the impact of information-directed encoding, in particular the way this is achieved in SGML, a small fragment is shown below.

```
<performance>
 ....
 <p>The setting and lighting were designed by <name desc='miel'>Jo
 Mielziner</name></p>.
 <p>The incidental music was composed by <name>Alex North</name></p>.
 <p>The costumes were designed by <name>Julia S&ouml;ze</name></p>.
```

```
 <p>Presented by <name rend=unmarked>Kermit Bloomgarden</name> and
 <name rend='unmarked'>Walter Fried</name> at the <rs type='place'>Morosco
 Theatre in New York</rs> on <date>February 10, 1949</date>.</p>
</performance>
```

This simple fragment shows how a particular performance of a play is recorded. Part of the encoding concerns the structure of the original text (e.g. by recording paragraphs, <p>), part concerns an interpretation (e.g. by recording a name and a date). These components are recorded by *start- and end-tags*, that identify *elements*. Elements may be nested as shown. They may also have attributes, given additional information on the element, that should not be considered part of the element text itself. In the example, <name rend='unmarked'> specifies that the name is not marked in the printed form of the text; <name desc='miel'> is a reference to some other element in the document holding a description of the named person. The attribute value 'miel' is the identifier of some description element, declared as IDREF, for *Identifier Reference*.

> *Note:* An attribute IDREF='x' must refer to an element with an ID='x' attribute. Thus elements may refer to elements within the same document. Such hyperlinking is very limited, but coupled with the use of external entities it forms the basis for powerful hyperlinking, as defined in the HyTime standard, for *Hypermedial/Time-based structuring language*, ISO/IEC 10744:1992 (Iso10744). A revision of the standard is due june 1997.

Finally, special (non-standard) characters are represented by a reference to a named character image, as in ö, representing the character 'ö'.

The fact that performance texts may consist of 1 or more paragraphs, and that a name element may be entered in paragraph content and may have attributes, is expressed in an *element and attribute list declaration*, which is part of the *document type definition*, or DTD. The DTD is the complete grammar for a specific document type, holding declarations for all elements that may occur in the document. As an example, the <performance> element and the <p> element may have been declared as:

```
<!ELEMENT performance - - (p*)>
<!ELEMENT p           - - (#pcdata | name | date | rs)*>
```

The last declaration implies that only <name>, <date> and <rs> elements may occur in the textual content of <p>. In turn, the <name> element may have been declared as:

```
<!ELEMENT name        - - (#pcdata)>
<!ATTLIST name        rend CDATA #implied
                      desc IDREF #implied>
```

Both attributes may be specified for a <name> element; neither is required.

The examples given intend to illustrate three aspects of SGML encoding:

1. The encoder must be able to understand the information captured by the text; for instance, she/he must be able to identify a place name within the source text. Also, she/he must be able to identify the correct encoding for such fragments. This process is partly supported by available SGML word-processors.

   > *Note:* SGML word-processors assist in finding possible tags for specific sections, and give some guidance in specifying required or possible *attributes* for elements, as in <rs type='place'>. Currently these programs do not assist in determining consistency in the elements used to encode a phenomenon throughout the document, nor do they allow for attribute values or textual element content to be validated in accordance with a particular DTD (see below), when this augments standard SGML validation.

2. The encoder must understand the overall structure of texts that record performances; for instance, she/he must introduce elements in conformance with the *grammar* for such texts. In SGML this grammar is recorded in the DTD. Luckily, SGML editing tools inherently validate proposed

encodings against the DTD, and offer immediate feedback on what encoding may be adapted in a particular context.

3. Within the realm of SGML validation many different constructs may be used to convey the same information. In the example, <rs>, for 'referring string', is used to record a place name. The DTD may also define a <place> element to be applied in a similar context. The choice for <rs> or <place> may depend on the level of precision required. This immediately requires scholars that need to exchange the electronic sources to decide on the precision, or even the 'flavour' or the encoding applied within the bounds of the same DTD.

Declarations do not only concern elements as shown, but also data that should not be treated as text. Such 'data portions' are modelled as *external data entities*, and may concern images, sound recordings, formatted formulæ, or even complete SGML documents. For such data portions the encoding strategy applied should be recorded in order to be able to merge the data in the current document. As such, *notations* are declared, and referred to by name when some external data entity is captured in the document. An example will be given later.

The introduction of such external data entities, formatted in any (unspecified) way, to be incorporated or referred to from within the document makes the SGML document a hypermedial object. The encoder may freely link and merge such media using standardized constructs. She/he must however know about the nature and validity of the external data in the context of insertion.

The previous short overview shows that the language meets the requirements outlined earlier. In summary:

- First, SGML offers a way to encode information within documents in accordance with a *grammar* for that kind of information. This grammar determines what parts can occur in the text, and how these are structurally related. This also applies to cross-references and hypermedial links. As a result the encoded source can be queried by its structural form, as well as by its textual content.

- Second, SGML is designed such that the encoded documents can be exchanged freely between software/hardware platforms, which is also essential for archiving purposes. The introduction of SGML has in fact liberated the electronic text from two counter-productive axioms: that it is of use only when printed, and that it is bound to a single software product.

Many SGML-based software products are available nowadays. This concerns SGML word-processors, (internet) browsers, converters, document managers and on. The software is, by nature, integrated on the data level, i.e. the 'SGML document'.

> *Note:* This way complete document production and management environments come into existence, that are build out of software products only related by the encoding language used. See http: www.falch.no/people/pepper/sgmltool or the SGML home page for an overview of products.

The creation of an SGML document does not require special computer skills. However, as SGML documents by nature record the structure and information of the text, knowledge on the problem domain is usually a requirement. To take this a bit further, in humanities research the creation of an SGML document *itself* may well turn out to be a scientific achievement.

Sect. 3.1 Toward a common ground for encoding textual sources in the humanities: TEI

As explained in the previous section, in addition to a standard language for encoding (such as SGML), we need an encoding convention, in the form of a DTD, that mirrors the requirements of humanities research on textual sources. Such a DTD standardizes the way common structures in texts of a specific type are to be encoded, i.e. defines an *application* of the SGML language, offering the encoder a framework for recording information of that type. Also, such a convention allows software to be built that is able to process the

documents in a more sophisticated fashion, as processing directives for elements and attributes in certain context can be assumed in advance.

> *Note:* For instance, the element <link from='n341' to='n345'> can be understood as identifying all elements that occur between the elements with *identifiers* 'n341' and 'n345'. Such *interpretation* of the element is not part SGML (Sperb95), and therefore not of general SGML software products. See also Loeff96 on added validation and SGML semantics.

The *Text Encoding Initiative* (TEI) offers such a convention for recording textual sources though the publication of the **Guidelines for Electronic Text Encoding and Interchange** (Sperb94). The project started in 1987, a year after the ISO/IEC standard was published. By the efforts of several working committees and the voluntary work of hundreds of researchers who have -- largely through electronic media -- shared their experience in text encoding, the specific recommendations as recorded in the guidelines have been compiled, forming a complete SGML application.

> *Note:* The TEI Guidelines have been prepared with grant support from the *U.S. National Endowment for the Humanities*, *Directorate General XIII of the Commission of the European Union*, the *Andrew W. Mellon Foundation*, and the *Social Science and Humanities Research Council of Canada*. The project is sponsored by three professional societies active in the area of computer applications to text-based research: the *Association for Computers and the Humanities*, the *Association for Literary and Linguistic Computing*, and the *Association for Computational Linguistics*.
>
> The authoritative reference for the complete TEI guidelines is Sperb94. In three issues of **Computers and the Humanities** a broad description of the initiative and the tag sets was published. These are collected in Ide95.
>
> The internet address for the TEI is http: www.uic.edu/orgs/tei/. For a complete overview of TEI projects see http: www-tei.uic.edu/orgs/tei/app/topics.html. The main channels of (electronic) discussion are TEI-LIST (use) and the TEI-TECH (construction) mail servers. These mailing lists are archived at http: CandL.let.ruu.nl/ [Archive discontinued].

The project originally set out to define a set of tags to be used in encoding textual sources in the humanities, such as prose text, drama, poetry and on, along with sets to define shared aspects of such texts such as those for recording alignments between texts, recording names and dates, and merging texts in a corpus. A concise statement on the initial goal of the project, and its current status, is found in the preface to Sperb94, quoted here:

"The impetus for the project came from the humanities computing community, which sought a common encoding scheme for complex textual structures in order to reduce the diversity of existing encoding practices, simplify processing by machine, and encourage the sharing of electronic texts. It soon became apparent that a sufficiently flexible scheme could provide solutions for text encoding problems generally. The scope of the TEI was therefore broadened to meet the varied encoding requirements of any discipline or application. Thus, the TEI became the only systematized attempt to develop a fully general text encoding model and set of encoding conventions based upon it, suitable for processing and analysis of any type of text, in any language, and intended to serve the increasing range of existing (and potential) applications and use."

The TEI directives intend to give guidance in:

- the interchange of electronic texts. Local encodings should be translated to the proposed SGML encoding (SGML/TEI as a *lingua franca*), and *vice versa*. Rules for defining strict conformance to the TEI guidelines are given.
- application-independent, local processing of textual material. TEI tries to standardize the encoding conventions and terminology such that software acquired will conform to the problem domain recorded in the texts.
- singular projects with regard to acquisition and preparation of texts. A useful checklist is provided

from which scholars planning to create electronic texts should select the subset of features suitable for their project.

The TEI conventions are currently applied in projects concerning archive and museum information, classical and medieval literature and language, dictionaries and lexicographies, electronic publishing projects, language composition and teaching, historical materials, language corpora, legal, literary, music historical and religious texts *et cetera*.

The work is not yet complete, though a first edition of the guidelines has been published in 1994 in an official version Sperb94. This printed version is known as the 'P3', for 'proposal 3' (the P1 and P2 versions were made available for public comment electronically only), and since its publication as the 'green book'.

> *Note:* P1 was published in 1990. All versions of the guidelines are publicly available in electronic form (see http: www-tei.uic.edu/). Information on proposed changes for P4 can be found at http: www.uic.edu/orgs/tei/trc/.

Sect. 3.2 The essentials of TEI encoding

Without going into details on the construction of the TEI guidelines we offer a small overview, which will clarify the impact of TEI conformance in text encoding. A TEI document consists of a header and a body text.

- The header classifies the document body in a highly structured way. It records information on the primary material, but also the electronic copy, and decisions made in the encoding process.

  > *Note:* The TEI header has been proposed as a generic classification scheme for bibliographical references, as discussed in Heery96.

- The text body holds the text itself. It is assumed to have a structure organized in divisions and subdivisions; a division may play different 'roles' in different contexts. For example, specifies a chapter division in a prose text, while identifies a stanza in verse text. The text types have the element in common. Thus a *default text structure* is defined.

The text body can be any of the following types:

- *prose*, which does not add special constructs to the 'default' text structure, i.e. any text may have 'prose' features;
- *verse*, where texts are sectioned into verse structures, ultimately into lines and line groups;
- *drama*, for drama texts, adding specific constructs for recording captions, moves on stage, camera settings and on;
- *dictionary*, for electronic dictionaries, introducing elements for recording grammatical aspects, etymological relations and such;
- *speech transcription*, recording speech events, pauses, vocal aspects and on;
- *terminological database*, for recording database information in support of other encodings.

These 'document types' can be freely combined with additional sets of elements, that cover an aspect not intrinsic to a particular document type. These set introduce elements for, among others:

- *linking, segmentation, and alignment*, i.e. for connecting disseparate information within documents;

- *simple analysis*, for recording semantic or syntactic interpretations of text fragments;
- *transcription of primary sources*, recording aspects of transcribed manuscripts, also for compiling a basic critical apparatus;
- *critical apparatus*, for recording a full critical apparatus through witness lists;
- *names and dates*, for recording names and dates to a high level of formality;
- *formulæ notations and contents*, for tabular information, scientific formulæ and introduction of other media;
- *certainty and responsability*, for documenting encoding decisions;
- *graphs, networks and trees*, for recording graph structures such as genealogies and grammatical relations;
- *language corpora*, for building complete corpora of texts.
- *feature structure*, for recording aspects of text through associations between a name and a value as recorded in an external document, the so-called *feature system declaration*

As such, within the DTD a TEI conforming document will select at least one 'base tag set' (e.g. drama) and any combination of 'additional tag sets' (e.g. simple analysis). The document could well have the following overall structure:

```
<tei>
 <teiHeader> ..information on this electronic text... </teiHeader>
 <text>
  <div type='play'>
   <div type='cast'>           ..cast of the play.. </div>
   <div type='act' n='1'>
    <div type='scene' n='1'>    ..act 1, scene 1.. </div>
    <div type='scene' n='2'>    ..act 1, scene 2.. </div>
   </div>
   <div type='act' n='2'>
    <div type='scene' n='1'>    ..act 2, scene 1.. </div>
    <div type='scene' n='2'>    ..act 2, scene 2.. </div>
   </div>
  </div>
 </text>
</tei>
```

Zooming in on some particular part of the play recorded here, we may find the fragment:

```
<div type='scene' n='1'>
 <head rend=italic>Actus primus, Scena prima.</head>

 <stage type=setting rend=italic> A tempestuous noise of Thunder and
 Lightning heard:  Enter a Ship-master, and a Boteswaine. </stage>

 <sp>
  <speaker>Master.</speaker>
  <p> Bote-swaine.</p>
 </sp>
 <sp>
  <speaker>Botes.</speaker>
  <p> Heere Master: What <app>
                          <rdg wit=H>cheere</rdg>
                          <rdg wit=M>cheer</rdg>
                         </app>?</p>
 </sp>
 <sp>
```

```
 <speaker>Mast.</speaker>
 <p> Good: Speake to th' Mariners: fall too't, yarely, or we run our
 selues a ground, bestirre, bestirre.
 <stage type=move>Exit.</stage></p>
</sp>
....
```

The elements <sp> (speech), <speaker>, and <stage> (stage direction) are introduced by the tag set for drama texts. The <app> (apparatus) and <rdg> (reading) elements is introduced by the critical apparatus tag set.

Sect. 3.3 TEI supported constructs in software

The previous has shown that a well defined set of elements and element relations has been defined for encoding common text types and analytic aspects of humanities research. This however primarily results in richly encoded texts; it does *not* imply that relevant information can be extracted from these texts. No processing specifications have been treated. The question therefore remains to what extent available software supports these conventions for scholarly purposes. And, to what extent future software may support the implied processing requirements.

Here we will focus on making the TEI document available to the reader. The normal way of doing so is the use of *browsers*, or 'SGML document viewers'. Browsers are all about presenting the information in an attractive way. Therefore they are well suited for specifying how this should be done. All SGML browsers use a style sheet for this. A *style sheet*, then, is a specification of what form should be given to an element or data entity found in a specific context. In doing so, the immediate context, and the attributes can be queried and the text displayed may be tailored to mirror this. Style sheets are documents themselves. An attempt to standardized the style sheet is the DSSSL standard, for *document style semantics and specification language*, ISO/IEC 10179:1996 (Iso10179).

The style sheet concept builds on the fact that this kind of 'browser behavior' can be described in a generic way. Similarly, the creation of a table of contents can be specified externally (passed as a so-called 'navigation sheet'). The table of contents is normally show along with the document text, and serves as a structured navigation tool. These external specifications, to summarize, tailor a generic mechanism to suit specific needs.

SGML browsers and editors do not automatically support TEI constructs that cannot be mapped onto general SGML constructs and/or style/navigation specifications. For instance, while a browser will be able to signal a link entered as an IDREF attribute <link target=XYZ> by e.g. placing an icon, and will activate the link 'on mouseclick', it will not be able to understand <fragment start=frag1start end=frag1end>, intended to identify a text fragment not contained in a single element. Such added semantics must be 'programmed' into the browser; they cannot be expressed within a style- or navigation sheet. If these semantics are made part of the browser, it is said to *support* this construct. Some of the current browsers support specific TEI constructs, most notably the *extended pointer syntax* as mentioned in the section on the TMI below.

Similarly, some browsers support HyTime constructs allowing for *webs* to be accessed, linking several documents and data objects in other notations in a variety of ways. As HyTime constructs are recorded by SGML elements, a web --again-- is recorded as an SGML document. The elements in the document do not record the source text, but rather information on relations existing within the source text(s). The web thus describes derived information by linking seemingly disseparate portions of the documents as a result of some kind of analysis. It offers a 'superimposed view' on the text by --in terms of the browser-- allowing to group together these sections or link in from one section to another. In the process the source documents are left untouched; accordingly, webs may be added, changed and removed freely as a result of new insights. In the section on printers devices below the practical implications of this are illustrated further.

What should have become clear from the previous is that, just as a generalized SGML processor cannot foresee the many different constructs used to encode a particular phenomenon, an SGML browser cannot cover each and every function implied by the encodings. This observation calls for a generalized browser that allows such semantics to be specified externally, to be read in along with the data. No such browser exists yet.

In the remainder of this article we will describe two projects using the TEI guidelines for encoding textual sources: the *Thesaurus Musicarum Italicarum* and a corpus of descriptions of printers devices and emblem books.

Sect. 4 Applying the TEI: The Thesaurus musicarum italicarum

The Renaissance was a period of particularly intense musical activity, not only in practical music, but also in music theory. Theorists such as Johannes Tinctoris (c. 1435-1511?), Franchinus Gaffurius (1451-1522), Heinrich Glarean (1488-1563) and especially Gioseffo Zarlino (1517-90) display in their writings an encyclopedic knowledge of both musical and non-musical subjects, acuteness in reasoning, and great understanding of practical music. A digital corpus of Latin theoretical writings from the Renaissance and earlier times, the Thesaurus musicarum latinarum (TML), has been under construction since Spring 1989.

In September 1995, work on a complementary corpus of Italian music treatises, the *Thesaurus musicarum italicarum* (TMI) has begun as an initiative of the Department of Computer and Arts, Utrecht University. A number of institutions from the Netherlands and abroad participate in this project. The first phase of the project is the preparation of an experimental CD-ROM with the music treatises of Zarlino; it will appear before the end of 1997. In the second phase, a number of similar writings by Zarlino's contemporaries will be added, and the system will migrate to a WWW server with controlled access.

> *Note:* The treatises are the *Istitutioni harmoniche* (in two editions, 1558 and 1589), the *Sopplimenti musicali* (1588), and the *Dimostrationi harmoniche* (1589), all published in Venice.
>
> *Note:* Formal partners in the project are the *Department of Computer and Arts*, the *Department of Musicology*, the *Arts Library*, the *Electronic Library Utrecht* (all belonging to Utrecht University) and the *National Library of the Netherlands* (*Koninklijke Bibliotheek*). Considerable funding for the project has been received from the IWI project group of the *SURF* foundation. Informal collaboration exists with the Music Departments of Royal Holloway (University of London), University of Delaware (Atlantic City, USA), and the Université catholique de Louvain (Louvain-la-Neuve, Belgium).
>
> TML materials are available from an Internet site, gopher: iubvm.ucs.indiana.edu/11/tml. Texts are in ASCII format with minimal markup; illustrations are in GIF format. Most of the principal Latin texts about music are now available in the TML, and new materials are added regularly.

Sect. 4.1 Hypertext/media as a scholarly reading strategy

One aim of the TMI is to develop a prototype of a software environment that suits traditional scholarly practice. Thus, in the 'digital workshop' of the TMI there will be a number of tools that the user will recognize from the non-computerized world. More importantly, the electronic documents will be presented in a natural manner, enhancing, but not radically altering, traditional reading practice of primary sources. This may seem largely a question of providing the right software tools, but, as it was emphasised in the introduction, the fundamental requirement is that texts must be *structured* in order to serve as an information base. It follows from this requirement and the TMI's aim mentioned above, that for this project the documents must be structured in a manner that conforms standard scholarly practice.

Music treatises from the past are seldom read from cover to cover, and we assume this is the case for many other kinds of primary sources in book format. One reason for this is the nature of the sources, another the nature of research.

The nature of Renaissance music treatises is such that they encourage non-linear reading. Many treatises owe their existence to prolonged and violent debates. The information they contain is thus by definition partial and incomplete, and the statements they contain can only be understood in relation to other texts. But generally, hardly any music treatise from the Renaissance is self-contained. Only few of them focus on one subject and treat it comprehensively. The model is rather to discuss music from a number of different angles (mathematics, notation, counterpoint, performance etc.); it is left to the reader to integrate these views. Moreover, it is often assumed that the reader already possesses a certain level of knowledge of music, but even more so of classical authorities on subjects like poetry, theology, rhetoric and mathematics. Usually, only the most rudimentary references are provided to authoritative texts, but being unable to interpret these clues often means loosing the thread of the argument. Zarlino, who discussed an exceptionally large number of such authorities in his writings, had himself the status of such an authority from the end of the sixteenth until far in the eighteenth century. As a result, a later writer claiming to discuss, for example, Plato's views of music, may in fact rather be working from Zarlino's interpretation of these than from the philosopher's own writings.

> *Note:* As in the case of Giovanni Maria Artusi (1540-1613).

Renaissance music treatises thus interconnect in a dense, intertextual, web of information. Actually, many more materials belong to the same web: for example writings in other languages, on other subjects, and from other periods. So do objects in other media, such as musical works, instruments, and music iconography. Potentially, the Global Information Web is unlimited, of which the music treatises form only a sub-web. The contents of this corpus are represented in the most logical manner if the information web itself rather than the individual text is taken as the starting-point for system development.

Research is usually problem-driven, and the first task is to localize and extract the relevant information from a context which is not relevant to the problem. While the amount of potentially relevant source materials may be very large, significant information is usually scarce and scattered over a number of documents. Secondary literature may contain references to some of the information, but these are usually selective. Further materials may be traced by searching indexes and tables of content of potentially relevant sources, or if a source lacks these, by browsing the entire document superficially.

Further study of the findings may include evaluation of each, comparing them to establish of links, the following up of internal and external references (which may be to yet unknown findings) the identification of sources, allusions etc.; comparing them with the objects (such as poems, music, art-works, buildings, or locations) they describe, the annotation of findings and links and finally the integration of results in a report. From the point of view of research, too, sources form an information web rather than appearing as individual entities. The knowledge that is generated from the study of this web can also be considered as a continuation of the same web.

The TMI must then be an extensible, hypermedial system. It must be extensible in order to be able to integrate new source materials and particularly new knowledge into it It must be hypermedial in order to account for relationships within and between texts, and between texts on the one hand, and images and music within or outside the treatises on the other.

Such a system can be realized as a stand-alone application, consisting of digitized sources and a browser with facilities for annotation, such as SoftQuad Explorer. To realize the requirement of extensibility fully, however, it would rather take the form of a WWW server. It could then be linked to other webs and services, and new primary sources and secondary information could be added to it by researchers. In this manner, studying and editing the sources would become a truly collective effort.

In addition to hardware, such an information web presupposes software functionalities that have only be realized in part so far. But the fundamental issue is that the data be structured in the right manner. Is SGML able to do so? And if this is true, has TEI inherited this capability? Two issues deserve special

consideration:

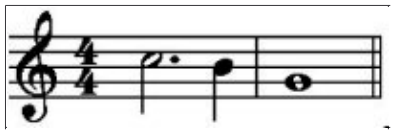
- the representation of non-textual information, such as music and images;
- the representation of links between textual, musical and graphical information.

These will be dealt with below.

1. Notations other than SGML may be employed in an SGML document, provided that they are recognizable as such. For this, the notation type must be identified by a notation declaration in the DTD, as was mentioned in the Introduction. For example, if DARMS is used to encode music examples, the following notation declaration must be added:

> *Note:* DARMS is an acronym for Digital Alternate Representation of Musical Scores. In the TMI, the Note Processor DARMS dialect is employed.

```
<!NOTATION DARMS PUBLIC
 "-//Thoughtprocessors//NOTATION Note Processor DARMS//EN">
```



Example 1. Music fragment.

Data using such a notation type can be handled in two ways. They may be included in the document instance within elements defined for this purpose. SGML parsers and browsers will skip these data, or pass them to another program. Example 1 could be encoded as follows:

```
 <music format="DARMS">!G !M4:4 6H. 5Q / 3W //</music>
```

Note that the attribute format= indicates the notation being used; the attribute itself must be declared in the DTD. Alternatively, the DARMS code could be stored in a separate file as an *external entity*, to be invoked from the document instance.

2. The ID-IDREF mechanism of the SGML standard can be used for cross-references within a single document (see the introduction). There is no such standard mechanism for references between different documents in SGML. The HyTime standard does provide for these, even if the documents are not in SGML format or even non-textual. While HyTime extends the possibilities of SGML enormously, it remains nevertheless wholly within SGML syntax. Unfortunately, only a small proportion of HyTime functionalities tends to be incorporated in SGML software, if they are at all.

SGML is thus able to represent the desired information, and being an SGML application, TEI inherits this capability. Including images and sound in a TEI application causes no special problems. Music notation is a case in itself, as it is rather to be seen as a special kind of text than as an image. This will be discussed below. For external links TEI has its own *extended pointer syntax* which is more compact than HyTime and moreover supported by several software packages, such as *Panorama Pro*.

> *Note:* The following shows an application of the extended pointer syntax in a reference from Zarlino's **Sopplimenti to his Istitutioni**:
>
> ```
> ...Quindeci chorde, da quello c'hò scritto nel <xref doc='zar.ist'
> from='ID (ch232)'>cap. 32. della 2. parte delle Istitutioni</xref>, & da
> quello che scriue prima Boethio...
> ```
>
> The attribute doc= identifies the external document and the attribute from= the actual element referred to. Note that both are stored in attributes. In this respect, TEI is more compact than HyTime, in which the locations are treated

as attibutes, but the external documents as separate elements.

Alternatively, TEI is extensible and it is easy to add HyTime links if necessary: obviously, it is better practice to use existing tags than to define new ones for the same kind of content.

Sect. 4.2 Text edition with TEI

The principal argument for employing TEI is that it provides a rich and extensible markup system for historical sources, as has been explained above. It is a model for *describing* the structure of existing documents, rather than one *prescribing* how documents must be structured. And the content of the documents can be *enriched without being altered* by means of special tagsets.

The last quality of the TEI is particularly interesting from the point of view of edition technique. On the most basic level, errors and corrections can be stored together in the same document, as for example in the following fragment from Zarlino's Sopplimenti:

```
Ma prima f&agrave; un bellissimo discorso, &amp; da buon <sic
corr='Mathematico'>ocitamehtaM</sic>; accioche al<supplied>-</supplied><lb>
cun non prenda marauiglia, ch'ei habbia pi&ugrave; tosto pigliato il
numero 18. ch'un'<lb>
altro in questo fatto onde dice.
```

Some explanation is in order. In the first place, the tag <lb> serves to indicate the original line endings. Secondly, hyphenation in this source is often implicit, leaving it to the editor to decide whether one or two words are intended (which is not always obvious in 16th-century Italian). In this case, a hyphen has been added, which is indicated by the <supplied> tag. And thirdly, a curious printing error occurs in the first line, marked with the tag <sic>, which possesses an attribute corr= for the editor's conjecture of the intended text. TEI possesses two related tag sets for such purposes, one for transcription and another for the critical apparatus. There are also tools in these for the rendering of alternative readings: the use of <app> and <rdg> for this has already been demonstrated above.

Documents that are marked up in this manner may become very complicated and in fact unreadable. Here an important aspect of SGML philosophy comes to the rescue, namely that only document structure is encoded, and that the presentation can be as the user wishes. SGML viewers therefore have the standard option of using different style sheets for the same document. In the case of the above example, a hypothetical style sheet 'source view' would result in the following view:

Ma prima fà un bellissimo discorso, & da buon ocitamehtaM; accioche al cun non prenda marauiglia, ch'ei habbia più tosto pigliato il numero 18. ch'un' altro in questo fatto onde dice.

Another, called 'edition view', would display the same fragment as follows (differences shown in bold):

Ma prima fà un bellissimo discorso, & da buon *Mathematico*; accioche *alcun* non prenda marauiglia, ch'ei habbia più tosto pigliato il numero 18. ch'un'altro in questo fatto onde dice.

And the possibilities are not limited to these.

One constraint that paper publishing puts upon editorial technique is that only one rendering of a document can be conveniently printed, while variant readings to are recorded in the apparatus at the bottom of the page or elsewhere. It is of course possible to print a number of versions of the same document, but these are actually separate editions and the coordination of them is difficult and again dependent on the constraints of printing technology. While we would by no means want to argue that information technology has totally discarded such constraints, we remain convinced that electronic editions are much better suited to deal with the 'dynamics' of the transmission of historic documents.

Sect. 4.2.1 Names

For a number of reasons, personal names occurring in historical sources are a troublesome category of information, while at the same time the readers may be especially interested in them. The threads in the 'web of music treatises' are after all often references to authorities or artists and their works. It is therefore of crucial importance that the TEI can deal with these in an adequate way.

Names may be incomplete. For example, Zarlino often writes about 'Adriano', but a only few instances he gives the full version 'Adriano Vuillaert'. This is only one of the many historic spellings of the name of this composer, who to the modern public is generally known as 'Adrian Willaert' (c. 1490-1562). Also, the identification in the source may be wrong. A famous example that occurs in many Renaissance music treatises is the ascription of an 11th-century text De musica to either Pope John XX or John XXII. John XXII had indeed expressed his (unfavourable) opinions on polyphonic music in writing, and as this was generally known his name was affixed to a treatise written by an otherwise unidentified John. Already in the 18th century this ascription was discredited. But the debate as to whether the author's full name is 'Johannes Cotto' or 'Johannes Afflighemensis' continues to the present day.

In the TEI, personal names can be labelled generically by the <name> element (see above for an example) or specifically by the <persName> element. The latter method offers more possibility of refinement. If the name 'Adriano Vuillaert' is identified in this manner, one may add a regularized form of his name by means of the reg attribute, as follows:

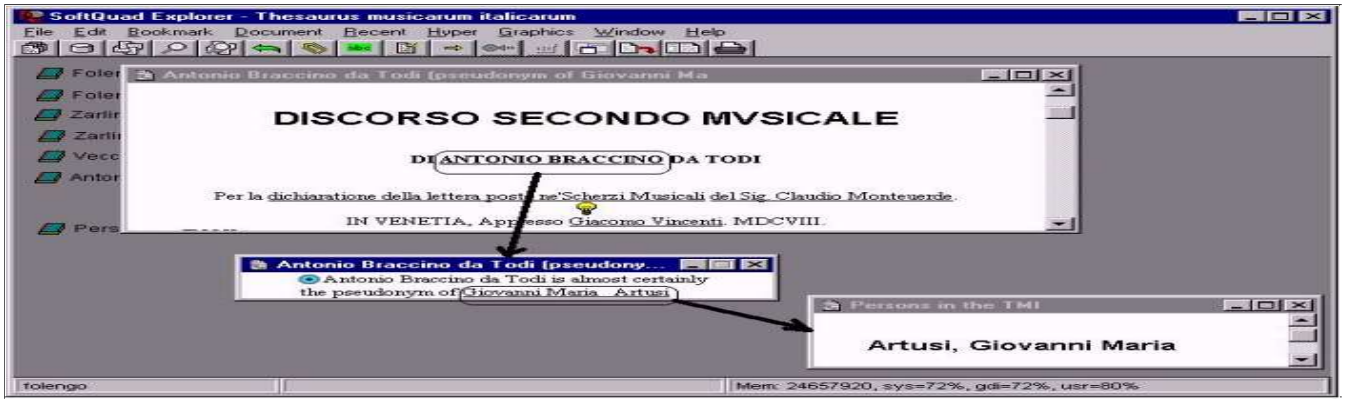
```
<persName reg='Willaert, Adrian'>Adriano Vuillaert</persName>, ueramente
uno de pi&ugrave; rari, che habbia essercitato la prattica della Musica:
```

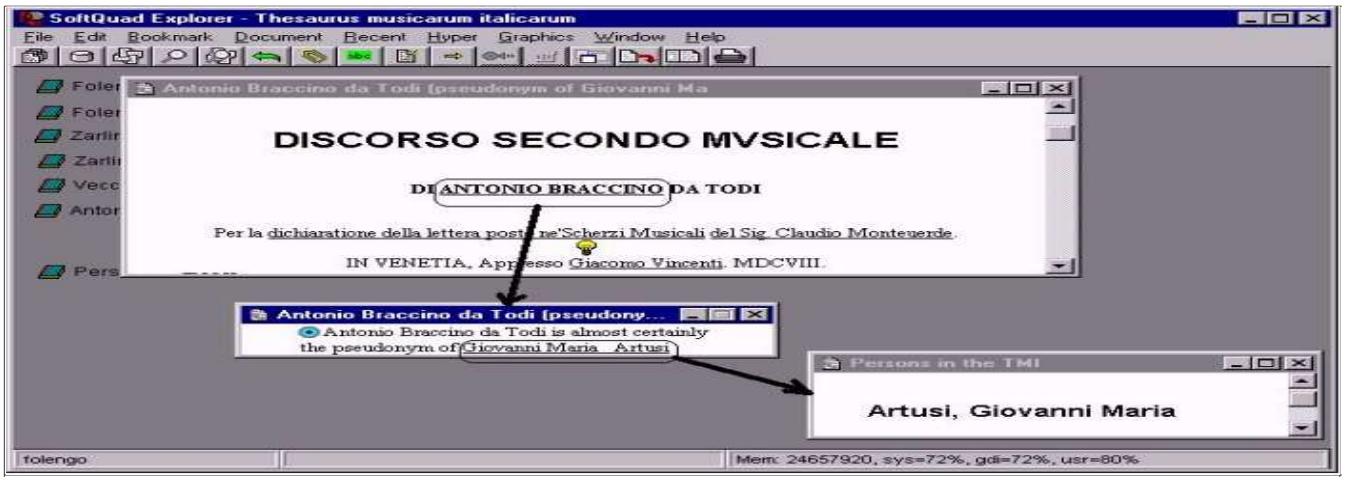
While this may seem adequate at first sight, there are some disadvantages. One is that one's opinion as to what the correct version of a name is may change. Another is that this attribute is meant for regularization, not for identification: if two or more Adrianos occur in the same text, they are all regularized identically. Therefore a second attribute has been introduced, key. It may contain some kind of identifier. It may contain the key of database record with full information about the person in question, or a link to another document with similar information. The identifier must be unique for each person, and not subject to spelling change, as in:

```
<persName key=will.adri>Adriano Vuillaert</persName>, ueramente uno de
pi&ugrave; rari, che habbia essercitato la prattica della Musica:
```

If the spelling of the name is changed, then it needs only to be done once. In the case of ambiguous information, the reference could be to an intermediate record or document.

Example 2 from the TMI demo shows how such information may be displayed in an SGML viewer. If the reader clicks on the name 'Antonio Braccino' in the largest window (containing the music treatise), a second window pop us, with an identification of this pseudonym. Clicking on the name 'Giovanni Maria Artusi' then leads to another document, which in the future will contain some bibliographic information. Links and indeed standard names are represented in the SGML document as much as possible through identifiers rather than through the literal character string.





Example 2. The linking of information about personal names in the TMI.

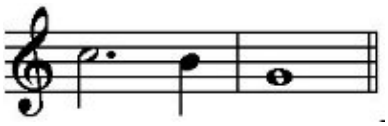
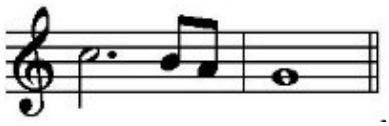
Sect. 4.2.2 The representation of music notation

Music notation is best seen as a two-dimensional form of text. As in ordinary text, the individual characters (notes, rests, clefs etc.) and their sequence are significant, but in addition the relative vertical position of the characters also has a meaning. Since the early sixties, a number of models have been devised for the representation of music notation in a one-dimensional character string. None of these has developed into a standard.

Note: Beyond MIDI: **The Handbook of Musical Codes**, ed. Eleanor Selfridge-Field. MIT Press, forthcoming.

The most influential one was DARMS, which was developed in the period 1961-1975. DARMS can be formatted by a music printing program to a traditional score. (Note Processor, unfortunately outdated MPP, supports this.) In addition, it can be converted into a sound code for synthesizers (MIDI), it can be manipulated by programs (See Alexander Brinkman) in musically significant ways (transposition, part extraction) and finally it can be searched and analysed, as in Otto Pool's program APOLLO (Pool95).

DARMS was designed to represent unformatted musical notation. It is best regarded as a 'musical ASCII', since it is unstructured. Yet to represent primary sources of music notation digitally in an adequate manner, we need to be able to structure them in a TEI-like way.

a:  b: 

Example 3. Variants from two sources, A and B.

Example 3, showing two variants of the same musical fragment, illustrates this point. For this example, the DARMS codes would be:

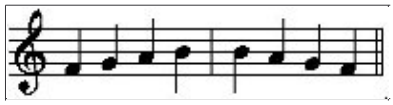
- Source A: !G 6H. 5Q / 3W //
- Source B: !G 6H. 5E 4E / 3W //

If we wish to record this information in one electronic document, we to add structural information as well. As an experiment, the 'musical ASCII' of DARMS has been enriched with TEI markup, in a manner very similar to what has been illustrated above:

```
!G 6H
     <app><rdg wit=A>5Q</rdg>
```

```
          <rdg wit=B>5E 4E</rdg>
      </app>
 3W //
```

For a number of reasons that we will not expound here, the combination of DARMS and TEI is far from ideal. Therefore, we have investigated a number of alternative music representations. The most promising of these was the proposed SMDL standard, *Standard Music Description Language*, Iso10743. However, in its most recent form, the proposal addresses only the *logical domain* of music, informally described in the proposal as 'the composer's intentions with respect to pitches, rhythms, harmonies, dynamics, tempi, articulations, accents, etc.' The *visual domain*, to which music notation belongs, is not the object of the standard. But it is impossible on principle for a researcher to know the composer's intentions, even if he/she is still alive. In the case of the long-dead composers of the Renaissance the only music representation the researcher possesses is notation. The logical domain is an interpretation of this notation, and moreover such interpretations are often controversial. Example 4 illustrates this.



Example 4: Unsingable progression, often found in Renaissance music notation.

Similar melodic progressions are common in Renaissance music notation; in performance the note B would be altered to B-flat, however. In the example, the alteration is almost self-evident, as the reader will find out when trying to sing these notes. In other situations such alterations are much less so, and may be hotly debated among scholars and performers.

> *Note:* For a recent example see Roger Wibberley, 'Josquin's Ave Maria: Musica Ficta versus Mode', **Music Theory online** 2/5 (1996), http: boethius.music.ucsb.edu/mto/issues/mto.96.2.5//mto.96.2.5.wibberley.html and the many reactions archived at http: boethius.music.ucsb.edu/mto/mto-talk/ (July-October 1996).

Encoding the logical information of Renaissance music is clearly an impossibility, since it it ambiguous. But since SMDL focuses on this information, it not suited for the representation of musical sources.

There is yet another problem. Almost all music encoding systems have as their object the common music notation (CMN) that has been used since the end of the 17th century. Zarlino's music notation differs in a number of respects from CMN, and other forms of notation from the same time, notably lute tablatures, do not even remotely resemble CMN.

Nevertheless, there is a common ground in most types of Western music notation. The notation is organized in lines (staves or otherwise). They may stand on their own, but usually a number of them is synchronized in some way, as in polyphonic music. Musical symbols are notated on these lines, and their meaning is expressed by their form, vertical placement on the staff, and groupings with other symbols. Specific notation types differ mainly in the kinds of staff, characters and groupings they actually use.

This analysis suggests that a SGML representation of notation may use one generalized structure, which can be applied to a number of different kinds of 'musical ASCII', each representing a different type of notation. For the text critical markup, a tagset may be used which is very similar to TEI markup, but containing some specialized extensions. The development of such a music representation is a considerable task. In the TMI project, no more than a prototype will be constructed, and then only for the notation forms that occur in the actual sources.

An important feature of TEI is that it can be extended by adding newly-developed tagsets, for example for music notation. In a similar manner, TEI may be tailored to fit a very wide range of source materials indeed. The consequence of such adaptability is that the researcher may be confident that the TEI will not enforce a prescriptive model upon the encoded document - and thus misrepresent it -, but that it will describe the document's intrinsic structure, in accordance with the researcher's requirements.

Sect. 5 Applying the TEI: Printers devices, emblem books and SGML

The printers devices project, in which the Royal Library and the department of Computers and Humanities cooperate, aims at the cataloging, description and study of 15th, 16th, and 17th century printers devices from the Northern Netherlands. We use material, mainly photocopies of title-pages, provided by the STCN (Short Title Catalogue of Books printed in the Netherlands) (Gruys83). Printers devices are small marks that printers and booksellers used to identify their work. They are often accompanied by a small motto.

When we started to work on the project, we assumed that each printer had a (one) printers device, and that one could use printers devices to identify printers, for example when no printer was named in the impressum. During the project it became clear that the use of printers devices was much more diverse or fluid than we had thought. In that light we decided to postpone, at an early stage, attributions of devices to printers during the project, and to simply register the names appearing on the title page or in the colophon. In our study of printers devices we concentrate on three basic questions:

- The attribution of devices: who used a device in a particular period?
- In the light of the first question: how precisely were printers devices used?
- The third question: What ideas were expressed with these printers devices?

It is this third question that leads us to the main theme of this section. What is depicted on printers devices and what did it mean in that time? The methodological question thus being: How can we study printers devices as an historical phenomenon?

Sect. 5.1 The 'emblematic game': printers devices and emblem books

It was Maurice Sabbe (Sabbe32) who first showed that 15th and 16th century printers devices from the Southern Netherlands became more and more 'emblematic' in character. His findings also apply to our material. During the 16th century we see, in the Northern Netherlands, printers devices transform from rather simple images depicting, in most cases, coats of arms combined with traditional commercial signs into richer, 'emblematic' images often accompanied by a motto. If we use our collection of printers devices to compare images and mottoes, we find that the words and images used, often refer to similar themes, but that they do so using all sorts of variations. We have called this playful way of constructing printers devices using combinations of words and images, the *emblematic game*.

The emblematic game, as we understand it, can be described as being about *copia*, that is abundance, riches, and wealth. In this we agree with, for example, Miede68 who asserts that at the heart of the emblematic genre lies embellishment, a concept encompassing activities like searching, explaining, changing, enriching, adapting, and paraphrasing. Practising *copia* one strives to produce *cornucopia*: something that is rich and full, but also adequate and suitable of form and content in a given context. This notion of the emblematic game as aiming for aptness and riches, does not accord with too restrictive a view of the emblematic genre. Were we to define that genre, we would have it embrace self-proclaimed emblem books, but also collections of imprese, devises, and adages, as well as the more theoretical or rhetorical tracts discussing the various sub-genres.

What is important to realize is that our study of printers devices, as described above, seeks to describe printers devices against the background of the *emblematic game*. A game in which all participants can play different roles at different times. The depiction of a lion can refer to a whole range of meanings: strength, *furor*, carefulness, cleverness, vigilance, braveness. A simple word used in a motto, like the Latin word *ingenio*, also has several layers of meaning, each with different connotations. Combinations of words and images can change or vary these meanings. Often, this playful way of combining words and images is inspired by already existing printers devices.

For our research it is important to have a firm grasp on the use of printers devices by printers and booksellers as well as the elements that were used to construct the contents of printers devices: words and (parts of) images.

Sect. 5.2 The corpus of printers devices

So how do we go about the study of printers devices? First of all, we construct an electronic corpus of printers devices. This corpus contains the data taken from the photocopies: name of printer and/or bookseller, description of the device (in plain text as well as with ICONCLASS notations), the motto, the year(s) of usage in a structured text file, as well as references to the record of that printer or bookseller maintained by the STCN.

> *Note:* This record contains the following data about printers and booksellers: name and variants of that name used, places and periods of activity, etc.)

The corpus of printers devices is kept as a text file, in plain ASCII format. It contains records that adhere to the following, rather simple, conventions: fields are identified by way of a field label, multiple entries per field are identified by (a) special character(s), records (groups of fields forming a unity) are identified by a special character. Such use of simple conventions like the ones mentioned above, make it possible to load the text-file into a database; for example to generate a particular output (all devices a certain printer is associated with) or to control the entries of a certain field. Instead of loading the file into a database, one can also use small programs, written in AWK or PERL, to perform the same operations. Why use a text-file and not enter the data directly into a database? First it is easier to use one's favorite editor, with all its extras like macro's, spell-checker, etc., to prepare the text. In the second place, one can use the structured text to change it, without any trouble, into something else, a SGML file for example.

With the help of this electronic corpus we can study printers devices in a very detailed way in order to assess what themes are predominant, what elements were used to construct these themes, and what devices were based on other devices. But in order to find out what people could think during the 16th and 17th centuries when confronted with these images and words, we also have to rely on sources outside the corpus of printers devices. Sources we use predominantly are the Adages of Erasmus and several 16th century emblem books.

Sect. 5.3 SGML: electronic editions of emblem books

The relationships between the elements contained in the emblem books and those of the printers devices are, seen in the light of the *emblematic game* as discussed above, complex (relations between words and words, words and images) and varied (paraphrase, the use of synonyms, juxtaposition, etc.). Therefore the need arose, during this project, to have some of these texts in an electronic format. On the side of words, the electronic text, the computer and software (i.e. search routines) give us the opportunity to query texts in detail, save queries, compare the results of queries and keep track of links between results of searches and their contexts in the sources. Images, embedded in the texts, are described with ICONCLASS notations. These notations give detailed as well as more general access to the iconography of the images.

> *Note:* This has to do with the hierarchical structure of the ICONCLASS system. To explain the structure of ICONCLASS in detail here, would be beside the point. We will use one example to clarify the mechanism. A 'spade' can be described with the ICONCLASS notation `47I15(SPADE)`. In that notation the part `47I1` stands for 'agriculture'. A query with the term `47I1*` leads us to all scenes that depict something that has to do with agriculture, including all images that depict a spade.

The direct context of the image is often its surrounding text, therefore we have to preserve the relation between text and image of the original source in some manner.

For our kind of research the availability of electronic texts is fundamental. Not only because we want to

query a large body of material that consists of words and images in detailed and varied ways, as we explained in the previous paragraph, but also because we want to use similar sources, like the emblem books and collections of adages as background, linking various elements in order to document the use of words and images. These considerations, together with the idea that it might well be that our sources contain information that was not used by us, but that is of interest to other researchers, brings us to two practical questions: what additional texts do we want to have electronically and in what format do we keep our electronic texts?

Our choice of emblem books has been determined by several pragmatic factors, such as the number of emblems that could at first sight be related to printers devices, and the availability of an original edition at the Royal Library

Note: So far, we have digitized in whole or in part:

Alciatus	Andreas Alciatus, Emblematum libellus. (Latin editions: Christian Wechel, Paris, 1534; Mathias Bonhomme, Lyon, 1551; François Raphelengius, Leiden, 1591) The last includes a commentary by Claude Mignault.
Junius	Hadrianus Junius, Emblemata. (Latin edition: Christopher Plantin, Antwerp, 1565; Dutch translation: ibidem, 1575)
Giovio	Paolo Giovio, Dialogo dell'imprese militari et amorose & Gabriele Simeoni, Imprese heroiche et morali. (Guillaume Rouville, Lyon, 1574)
Paradin	Claude Paradin, Devises heroïques. (French edition: widow of Johannes Steelsius, Antwerp, 1563; Dutch translation: Princeliicke Deviisen, François Raphelengius, Leiden, 1615) (Both editions furthermore contain a translation, in French and Dutch respectively, of Gabriele Simeoni, Imprese heroiche et morali)
Sambucus	Johannes Sambucus, Emblemata (Latin edition: Christopher Plantin, Antwerp, 1569; Dutch translation: ibidem, 1566)
Whitney	Geoffrey Whitney, A Choice of Emblemes and other devises (François Raphelengius, Leiden, 1586).

As to the format, we have chosen for SGML. SGML itself is explained elsewhere in this article in more detail, so we will limit ourselves here to an account of the reasoning behind the choice. SGML encoded electronic texts are independent of existing hard- and software. This means that the exchange of electronic documents between computer platforms is possible without loss of information. To take a very simple example. Within a SGML encoded document or within the DTD associated with that document one formally declares what conventions were used to encode the Greek characters, or the mathematical characters, or the e with an acute accent (é), in short all those characters that are implemented differently on different platforms, or in different software programs. SGML handles these characters, but also references to external graphics, etc. in a uniform and standardized way.

SGML encoding of electronic documents is centered on the encoding of the (structure of the) information contained in these documents. Our SGML encoding of the emblem books, for example, structures the texts in the following way (simplified):

- BEGIN OF EMBLEM (indicated by the tag <div0>)
- BEGIN OF MOTTO (<div1 type='motto'>): Text goes here

- END OF MOTTO (</div1>)
- IMAGE (<figure>): Reference to an image and its description kept elsewhere
- BEGIN OF EXPLANATION: Text goes here
- END OF EXPLANATION
- END OF EMBLEM

Part of a real-life example, taken from the Dutch edition of Claude Paradin's book on devises, looks like this:

```
<div0 ID="emblem1" N="1" type='emblem'>
<xptr ID='ppaf1' doc='paf' from='ID emblem1'>
<head>I</head>
<div1 ID="motto1" N="1" lang=lat TYPE='motto'>
<p>Nullis praesentior aether.</p></div1>
<div1 id='koppel1' n='1' type='koppel'>
<p>Voorwaer <hi>gheen menschen zijn den hemel naeder comen</hi>,
Dan die ons Heeren Cruys te draghen heur bevromen.</p></div1>
<figure id='im1' n='1' entity='fig11'>
<figDesc><xptr id='pd1' doc='padIco' from='ID pad001'></figDesc></figure>
<fw type='sig'>A 6</fw><fw type='catch'>In als</fw>
<pb id='p12' n='12'>
<figure id='pf12' entity='fig12'></figure>
<fw type=pag>12</fw><fw type='header'>Princeliicke</fw>
<div1 id='uitleg1' n='1' type='uitleg'>
<p>In als gheluckich is den mensch die hem soo draecht,
Dat hem niet meer op eerd' in sijnen sin behaecht,
Dan volghen dese. Vaen, die met dit Teecken heylich
Allomentom verciert den me[n]sch hout vrij en veylich.</p>
```

The SGML codes used to structure the text (according to the rules laid down in the DTD) can be formally recognized as not being part of the source encoded, but at the same time they give powerful possibilities to manipulate that source:

- One can restrict queries to parts of the source. For example: the word *Ingenio* but only when it is used within a motto.
- One can easily extract structural elements from a source. For example: list all personal names. We use ICONCLASS notations as attributes of the <persname> element in order to be able to extract all female historical persons, because they are encoded as <persname type="61BB2">, or all authors from classical antiquity that are mentioned in a text, etc.
- One can maintain links between structural elements within a text and between elements from different texts. In the example above one can see that within <div0>, the division used for an emblem, there is an element <xptr> that points to the same emblem in the french edition of Paradin.
- One can have different structural views on one source at the same time. For example: one can code the beginning of every page of a book as well as use the coding scheme showed above. This means that when presenting the electronic document one can give users the choice to jump from page to page or from emblem to emblem.

With the last point we have entered an altogether different area, that of presentation of SGML encoded documents. Before we will address that topic, there is something else we have to discuss: our choice for the TEI DTD.

Our choice for the TEI DTD stems, again, from pragmatic considerations. The TEI DTD is an elaborate

construction that targets users of SGML working within the field of the Humanities. We found it a powerful tool that, being well documented, is suited to structure the texts of our 16th century emblem books, with all their idiosyncrasies, like one book with two title pages, or two books in one.

> *Note:* We had to adjust the TEI DTD in a very slight manner, in order for it to cope with one or two peculiarities of our project. To give an example. We photograph an emblem book from cover to cover. The photographs are digitized, and links are made in the electronic text from every page to its digitized image. The tag one should use, which is <figure>, is declared in the TEI DTD to be part of paragraph content. In our case the tag <figure> is used throughout the document after a tag that indicates a page break, therefore we declared it valid for the element <text>.

Due to the modular approach of the TEI DTD, we, or users of our files, can easily expand the mark-up of texts under investigation. For example, suppose we want to use one of the mentioned emblem books to prepare a critical edition, we can simply add the TEI text critical module to our DTD and from that moment on we can use the specialized tags this module defines in our document.

Sect. 5.4 Working with SGML encoded documents

Apart from the advantages of using SGML mentioned in the paragraphs above, we have found the most important advantage to be the creation and maintenance of links between our articles and the sources they are based upon. Remember our data file with the descriptions of printers devices. We can easily change that file into a SGML file and thus refer to the data contained in that file from the articles we write about the printers devices. We have found it stimulating to show the practice of the *emblematic game* within the realm of the printers devices by weaving webs between words and images, and thus describing and analyzing the use of words and images in printers devices against the background of the *emblematic game*.

To give one example of the possibilities of making this web, we will use a printers device used by Pieter I van Waesberge (shown below).

Printers device used by Pieter I van Waesberge

This printers device depicts a lion holding a column; at the foot of the column we see a man spading, in the background we see two persons (discussing?). The device carries the motto: *Ingenio superatur*. Within the corpus of printers devices the record describing this printers device is linked to other records:

- To all the other records that describe the same device used by other printers and/or booksellers;
- To all the other records that describe other devices used by the same printer or bookseller.

The record is also linked to other sources. For example to a file prepared by the STCN that lists the shop-names of printers and booksellers. Combining the information after the traversal of these links learns us that, besides Pieter I van Waesberge, other members of this family used the same device. Furthermore that they had a shop in Rotterdam, at the *Steiger* near the *Korenmarkt* called *Inde de gecroonde Leeuw*. But there is more. From our database we want to make a reference to the corpus of printers devices from the

Southern Netherlands (published in printed format as Vande93), because a printer from Antwerp named Soolmans used as a device the image of a lion holding a column with the motto *Ingenio superatur*.

This is an important link, because it shows that motto did make sense in combination with the depiction of lion and column. The device used by Pieter I van Waesberge is richer in its imagery. We use ICONCLASS notations to link our printers device to other devices that depict a spading man, with of without the two additional persons. In fact, it turns out that the combination of spading man with two other persons is a printers device of its own, albeit in reversed form: the two men discussing are depicted in the foreground and the spading man is depicted in the background. The device, used by among others by Aelbrecht Hendricksz (see ill. below), is an illustration of Matthew 13:44. The spading is connected with the *Koninkrijk der hemelen, dat is gelijk een schat verborgen in de akker*. Just a spading man was used as a device by Joannes Maire, printer and bookseller in Leiden, sometimes with the motto *Fac et spera*, in other cases with the motto *Labore*. But we made still another link with a source outside the corpus of printers devices. An emblem book by Schoonhovius contains an interesting visual parallel to our printers device. A man puts up a column with the help of a crane. The motto is: .... In our case it is a lion, known for its cleverness that illustrates the victory over something heavy, something hard to accomplish (the column), but once accomplished, something that is strong and durable. The image of the spading man with the two man talking drags in several connotations: *Labore*, looking for the treasure hidden in the field (God's kingdom) and, through the association of the spading man with Adam after the expulsion from Paradise: man's mortality.

Device used by Aelbrecht Hendricksz.

Combining these threads we would conclude that Van Waesberge's device puts forward the message that hard work and man's ingenuity will overcome difficult tasks. Is this interpretation too far-fetched? In our forthcoming publication on Dutch printers devices we will show that there are similar devices that carry a similar message and that the theme of hard work was predominant in many devices. The example is meant here to show how we use the SGML files to link all sorts of information in order to get a grip on what we have called the *emblematic game*.

What software does one need to prepare SGML documents? Actually surprisingly little. Some members of our team use Softquad's *Author Editor* in a Microsoft Windows environment. Softquad's products have some problems with the modularity of the TEI DTD, but since these problems and their remedies are thoroughly discussed on the WWW, they can be easily overcome. Others use an add-on for Emacs running under Linux called PSGML made by Lennart Staflin, ftp: ftp.lysator.liu.se/pub/sgml, which is free of charge and does an excellent job. Validating SGML encoded documents is done by using a validating parser, we use SGMLS by James Clark. What software can one use to distill information from SGML encoded documents? Here things become a little bit tricky. Most commercial software available is, apart from being expensive, more or less tailored to accommodate the presentation of SGML encoded documents. Querying the data contained in the files is rather sober, and thus, seen from the research perspective presented in the

above, tends to be inadequate. Another drawback of commercial products is the way in which the hypertext links are implemented. We rely, as explained above, heavy on this feature when we prepare the results of our research. Sometimes a minimal set of HyTime links is supported together with the TEI extended pointer syntax (as in Softquad's *Panorama Pro*), sometimes a product (Softquad's *Explorer*) just offers minimal HyTime support.

On the presentation side, most commercial products do offer some useful features: navigation tools, stylesheets, and the possibility of making personal hypertext links that exist on top of the original documents.

Sect. 6 Conclusions

SGML produces device independent documents that explicitly record textual information, without depending on a particular presentational form. SGML compliant tools currently available cover all aspects of regular document production, including editors, browsers, converters, and document management systems.

The TEI, as an SGML application, does not restrain the power of the underlying SGML. The TEI conventions cover to a large extent the sources researchers within the Humanities work with. The conventions are flexible and can be adapted to particular needs. A TEI document can be presented reasonably well using an SGML browser, allowing for several distinct styles to be applied, hypermedial connections to be activated, and personal webs to be inspected. Equivalently, the document can easily be converted to presentation formats such as HTML or TeX.

The use of TEI provides an environment particularly suited for the study of material described in the previous sections, that consists of diverse informational parts that are connected in varied and often obscure ways. An important point here is that, since data used in research is available in an information-oriented, electronic format, other researchers can use these data either to verify any conclusions drawn from it, or to use the data for their own purposes.

We want to emphasize that no system can be powerful enough to describe every textual source. The TEI is not yet 'complete', and will for this reason *never* be complete. This consideration is imminent to the design, and a strategy is therefore offered for extending and altering the DTD's to suit local needs without altering the basic TEI guidelines.

As a final remark, we would like to point out that one has to be prepared to spent a considerable amount of time to get 'things right' (tailoring DTD's, preparing structured electronic documents with SGML, getting familiar with the software) in order to use the electronic source as part of a research project, or to make these sources available to the public. Whether or not a project is worth the extra time and effort can only be assessed at the end. This may require the scholar to take a visionary standpoint, and not be satisfied with immediate, less reusable results.

> *Note:* For further information, please contact Arjan Loeffen (SGML, TEI and related applications), Peter van Huisstede and Hans Brandhorst (Printers devices) or Frans Wiering (TMI).

References

- Alsch95 L. Alschuler: **ABCD... SGML. A user's guide to structured information**. London [etc], 1995. (International Thomson Computer Press)
- Bradl97 N. Bradley: **The concise <SGML> companion** Harlow, England [etc], 1997 (Addison-Wesley)

- Colby96 M. Colby, D. Jackson: **Special edition. Using SGML**. [Indianapolis], 1996 (QUE Corporation)
- Donov97 T. Donovan: **Industrial-strength SGML. An introduction to enterprise publishing.** Upper Saddle River, N.J., 1997 (Prentice Hall PTR)
- Goldf90 C. F. Goldfarb: **The SGML handbook**. Edited and foreword by Y. Rubinsky. Oxford, 1990 (Clarendon press)
- Gruys83 J.A. Gruys, P.C.A. Vriesema & C. de Wolf, 'Dutch national bibliography 1540-1800: the STCN' In: **Quaerendo**, 13 (1983), p. 149-160.
- Heery96 R. Heery: 'Review of metadata formats'. In: **Program** 30 (4) 1996. Also available as http:www.ukoln.ac.uk/metadata/review.html.
- Ide95 N. Ide, J. Véronis (ed.): **Text Encoding Initiative. Background and context.** Dordrecht (etc.), 1995. (Kluwer academic publishers)
- Iso10179 **Information technology -- text and office systems -- document style semantics and specification language (DSSSL).** International standard, 1996. ISO/IEC 10179.2:1994. (Draft International Standard published in 1994 as ISO/IEC 10179.2:1994).
- Iso10744 **Information technology -- Hypermedia/Time-based structuring language (HyTime).** [Geneva], 1992 (ISO/IEC 10744:1992)
- Iso8879 **Information processing -- Text and office systems -- Standard generalized markup language (SGML).** 1986, with amendment 1, 1988 (ISO 8879-1986/A1:1988 (E)).
- Loeff96 A. Loeffen: 'Toward semantic specifications for SGML encoded documents' In: **Informatiewetenschap 1996**. Proceedings of the 4th interdisciplinary conference on information science. Delft, 1996. Pp.73-97.
- Maler96 E. Maler, J. El Andaloussi: **Developing SGML DTDs. From text to model to markup.** Upper Saddle River, 1996 (Prentice Hall PTR)
- Miede68 Hessel Miedema, 'The Term 'Emblema' in Alciati', in: **Journal of the Warburg and Courtauld Institutes**, 31 (1968) pp. 234-50.
- Pool95 Otto Pool, 'The *Apollo* Project: Software for Music Analysis using *DARM*S', **Computing in Musicology** 10 (1995-96) 123-30.
- Sabbe32 M. Sabbe, 'Le symbolisme des marques typographiques', in: **Gulden Passer**, 10 (1932).
- Sperb94 C. M. Sperberg-McQueen, L. Burnard [ed.]: **Guidelines for electronic text encoding and interchange**. (ACH-ACL-ALLC report TEI P3) Oxford, 1994
- Sperb95 M. Sperberg-McQueen, L. Burnard: 'The design of the TEI encoding scheme'. In: **Computers and the Humanities** 29 (1995), pp. 17-39. Reprinted in Ide95.
- Travi95 B. Travis, D. Waldt: **The SGML implementation guide. A blueprint for SGML migration.** [Berlin etc], 1995. (Springer)
- Vande93 F. Vandeweghe & B. Op de Beeck, **Drukkersmerken uit de 15de en 16de eeuw binnen de grenzen van het huidige België**, Nieuwkoop, De Graaf, 1993.