A Generic Formalism for Encoding Stand-off annotations in TEI

Javier Pose, Patrice Lopez, Laurent Romary

A Generic Formalism for Encoding Stand-off annotations in TEI

Javier Pose Rodriguez
European Patent Office
jposerodriguez@epo.org

Patrice Lopez **Laurent Romary**
INRIA Saclay
patrice.lopez@inria.fr laurent.romary@inria.fr

Abstract

This article outlines a proposal for a consistent encoding of stand-off annotations in the frame of the TEI standard. The proposed encoding requires the extension of the current TEI schema with three additional elements, directly related to the encoding of stand-off annotations that provide a generic and flexible structure for encoding stand-off annotations in multiple layers or levels of annotations.

Keywords: *textual resources, TEI, annotations, stand-off annotations*

1. Motivation and Background

The Text Encoding Initiative (TEI) is the de facto standard for the representation of texts in digital form. It is maintained and developed by a consortium that is a non-profit membership organization composed of academic institutions, research projects, and individual scholars (http://www.tei-c.org/). The TEI Guidelines define encoding methods for machine-readable texts and provide an encoding scheme rendered in a formal markup language.

One field that has been very active in the last years and has gained the attention of many researchers deals with the encoding of annotations in XML. Corpus annotation is the practice of adding interpretative information to a text corpus, by coding added to the electronic representation of the text itself. The nature of the information can vary from simple notes or comments regarding certain parts of the text, to deep linguistic analysis, like morphologic, syntactic, semantic, discourse or lemma annotations. Thus, the annotations are basically metadata added post hoc that provide information about the different elements comprised in a document.

When encoding the annotations, two different approaches can be considered: inline annotations, where the markup elements of the annotations are directly added into the annotated data, and stand-off annotations, where the markup elements of the annotations reside in a location different from the location of the data being described by it.
Whereas the inline annotation has as main advantage its simplicity and easy maintenance due to the fact that the annotated elements are directly associated to the annotations, it also presents numerous drawbacks when implementing multiple types of annotations on the same source text. In such cases, the stand-off annotation mechanism has shown to be a better option (Dipper 2005).
The stand-off annotation mechanism is very flexible and allows the definition of new layers of information on the top of the source textual data without disturbing existing ones. Therefore, it seems to be the most obvious option for encoding multiple concurrent annotations on a source text.

Different text encoding standards have been proposed that allow the encoding of textual annotations: the standards developed in the frame of **ISO TC37 / SC4** (Ide and Romary 2006, Declerck 2008), **XCES** (Ide, Bonhomme and Romary 2000), **TIGER-XML** (Mengel and Lezius,

2000), **PAULA** (Dipper 2005), **XStand-off** (Stührenberg and Jettka 2009) or **TEI** (Burnard and Bauman 2008).

The Text Encoding Initiative (**TEI**) *is a de facto, constantly maintained XML standard for encoding and documenting primary data*. It has a very large base of users and *for primary data and metadata levels there is no real alternative to TEI* (Przepiórkowski and Banski, 2009). There have been different attempts to encode annotations, and in particular stand-off annotations, within the TEI (Boot 2009, Przepiórkowski 2009, Banski and Przepiórkowski 2009, Banski and Przepiórkowski 2010) but none of them offers a general framework for encoding such annotations. One detailed TEI implementation of a stand-off annotation can be also found in the TEIWiki page for "Stand-off use cases" (http://wiki.tei-c.org/index.php/Stand-off_use_cases). These attempts only disclose specific combinations of already existing TEI elements that allow encoding the annotations implemented in the corresponding projects.

Even if the TEI Guidelines [1] provides different elements for encoding stand-off annotations, it does not disclose any methodology for consistently encoding the stand-off annotations. Basically, the guidelines disclose the basic referencing mechanisms for linking pieces of the source document with the annotations (see the TEI Guidelines: Chapter 16.9, 17.4 and 20.4), but they do not provide any information or general guidance for encoding in a consistent manner the stand-off annotations, i.e. the linking information together with the annotation's content. Furthermore, it is also not clear in the TEI Guidelines where, in the whole TEI structure, the data corresponding to the stand-off annotations should be encoded.

Therefore, two basic problems arise when encoding stand-off annotations in TEI: where to encode the information of the annotation, i.e. the contents of the annotation and the linking information, and how to encode said information.

Since this type of annotations have become a very powerful tool for analysing and exploiting the encoded text, it seems to exist an urgent need of defining such type of general structure in the TEI schema for encoding stand-off annotations in a more comprehensive and coherent manner.

2. General encoding principles

The following requirements must be fulfilled when defining the general framework for encoding stand-off annotations in TEI:

- Expressiveness: the framework must be able to encode the heterogeneous types of stand-off annotation models. Thus, it should have enough flexibility to allow the encoding of multiple annotation layers, as well as the different content structures, references and segmentations that can be defined in the stand-off annotation models.

- Extensibility: the framework must be defined such that it can be used not only to encode stand-off annotations of textual information, but also of any type of source data, including audio, video, image, etc., providing a common formalism independently of the type of data.

[1] TEI: Text Encoding Initiative (http://www.tei-c.org/)

- Uniformity: the framework must provide a general mechanism that, even allowing different ways of encoding the annotation information, provides a generic structure that embraces all these possible encodings.

- Predictability: it should offer a general encoding frame such that any person not directly involved in the encoding of a specific type of annotation could understand the encoding structure and expect to have certain information at specific levels of the annotation's structure

- Clarity: the proposed framework should be clear and readable

- Flexibility: the encoding model should offer a high degree of freedom such that any type of stand-off annotation model can be encoded under the same and unique structure without the need of implementing big customizations

- Simplicity: the proposed frame must be simple to understand and to implement, not only for the human encoding the information, but also for the automatic tools creating and exploiting the annotations

3. The basic stand-off data structure

In order to identify what type of information should be considered when encoding the stand-off annotations in TEI, it is useful to analyse what is the nature of the data that can appear in a stand-off annotation. When analysing the contents of these annotations, the data can be grouped as follows:

- Stand-off metadata: comprises the data providing information about one or more aspects of the annotation. It should be stressed that this data do not provide information about the source text, but about the annotation itself, like for example author of the annotation, method of generation (manual/automatic), external ontology used, etc...

- Referencing data: comprises the data providing references to parts of the **source text**. Typically, references to the source text are implemented by referencing the id's of anchors or other elements of a basic segmentation inserted in the source text, or directly referencing parts of the text by making use of some offset mechanism, for example using character offsets. The referencing data can also be grouped to form groups of references.

- Annotation contents: comprises the data describing the information attached to the referenced source data.

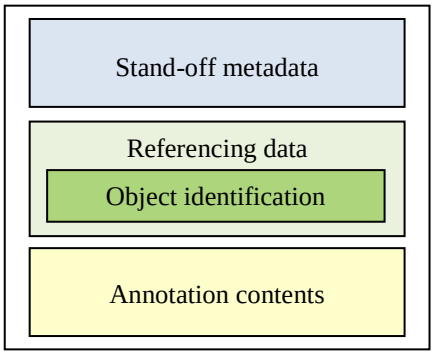

Fig. 1: The basic data

It should be noted that this classification of the stand-off data does not imply that the different types of data are always presented separated. For example, an annotation of a name in the source text could be encoded in different forms:

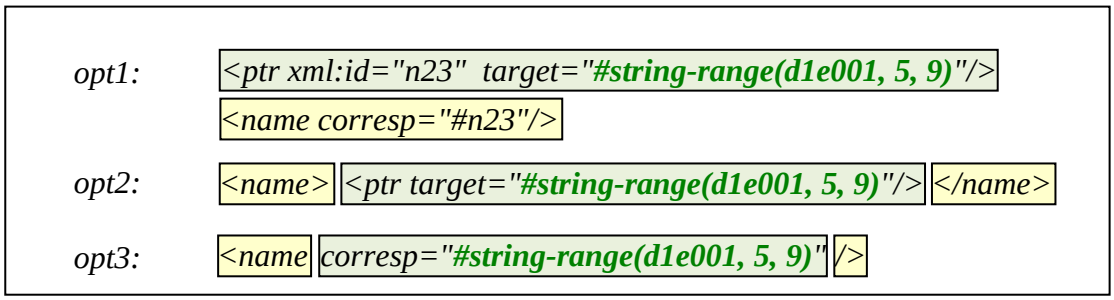

Fig. 2: Various TEI encodings of a simple standoff annotation

4. Encoding the basic stand-off data structure in TEI

In order to encode the information comprised in a stand-off annotation, it is proposed to extend the current TEI schema by adding a new element that can contain the data of a stand-off annotation.

In the present proposal, it is suggested to call this element **<stf>**, as an abbreviation of the world **stand-off**.

4.1. Encoding the stand-off metadata

Two are the main issues to consider when dealing with the encoding of the stand-off metadata:

- what to encode as metadata: in the present proposal, only a limited amount of information is suggested to be encoded as stand-off metadata:

  - id: indicating a unique identifier for the annotation
  - type: indicating the type of annotation, for example Part Of Speech (POS) annotations
  - category identifier: indicating the category associated to the type of annotation in a externally defined ontology, for example in ISOCat
  - author: indicating the person or software application responsible for the creation of the annotation
  - creation date: indicating the date when the annotation was generated or last modified

- where to encode the metadata: all these four basic pieces of information can be encoded as attributes in TEI using already existing attributes, so it sounds reasonable to encode this information as attributes of the element **<stf>**.

Therefore, the new element **<stf>** should have at least the following TEI attributes:

att.global (@xml:id):	for encoding the unique id
att.typed (@type):	for encoding the type of stand-off annotation
att.datcat (@datcat):	for encoding the data category registry
att.ascribed (@who):	for encoding the author of the stand-off annotation
att.datable.w3c (@when):	for encoding the creation date of the stand-off annotation

Therefore, a partial (only the minimum set of attributes) declaration of the **<std>** element is disclosed in the following figure:

```
element stf
{
   att.global.attributes,
   att.typed.attributes,
   att.datcat.attributes,
   att.ascribed. attributes,
   att.datable.w3c.attributes,
   ...
}
```

Fig. 3: Partial declaration of the element **<stf>**

4.2. Encoding the stand-off referencing data

As it was already indicated above, the referencing data comprises the data providing references to parts of the source text.
When encoding the referencing data in the different stand-off annotation models, there are two different approaches that can be followed:

- grouping of stand-off references: in this encoding approach, all the references to the source text are grouped and kept independent of the rest of the stand-off data. Therefore, in this case, specific elements are used for referencing the source text. The referencing data can also be further grouped to form subgroups of references inside the main grouping. The annotation contents make use of the defined stand-off references by referring to their id's.
  This approach is followed, for example, by the PAULA and XStand-off encodings.

  In the PAULA encoding ((Sonderforschungsbereich632 2008), all the references to the source text are encoded in the **<mark>** elements that are grouped with **<markList>** elements. The other elements use these references by using references to these elements.

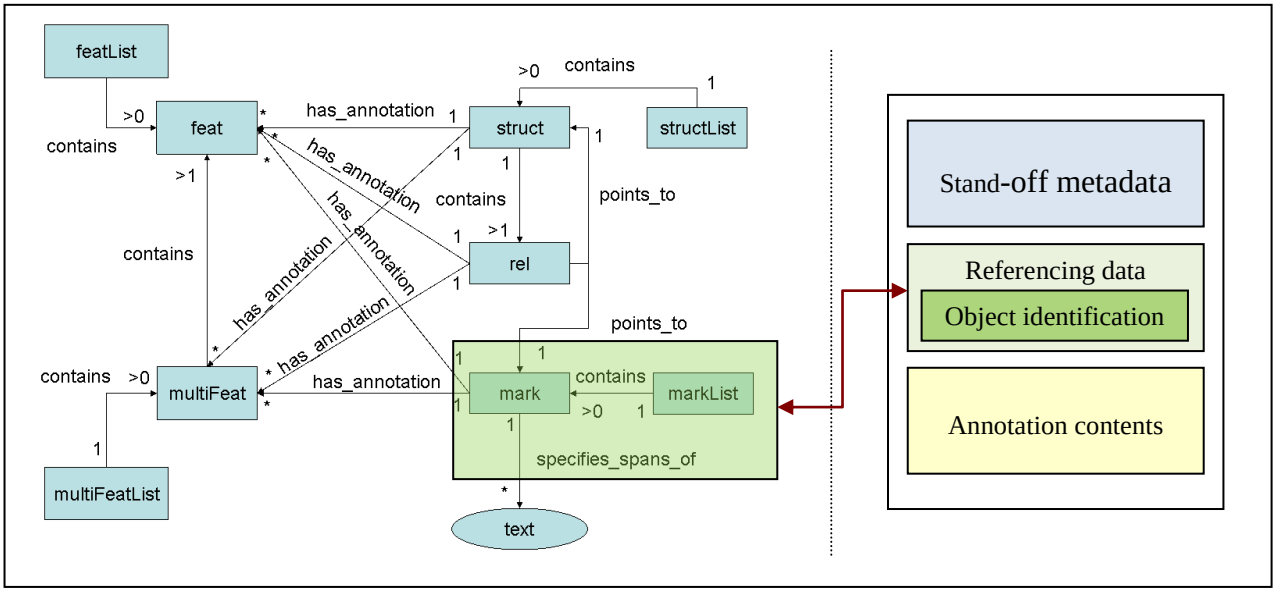

Fig. 4: Paula 1.0 Conceptual Structure (Data Model) with <multiFeat> and <...List> elements

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE paula SYSTEM "paula_mark.dtd">
<paula version="1.1">
  <header paula_id="mycorpus.doc2_tok"/>
  <markList xmlns:xlink="http://www.w3.org/1999/xlink" type="tok" xml:base="doc2.text.xml">
     <mark id="tok_1" xlink:href="#xpointer(string-range(//body,'',1,2))"/>
     <mark id="tok_2" xlink:href="#xpointer(string-range(//body,'',4,5))"/>
   </markList>
</paula>
```

Fig.5: Identification of standoff alignments in a exemplary PAULA encoding

In the XStand-off encoding (Stührenberg and Jettka 2009) all the references to the source text are encoded in the **** elements that are grouped using **<segmentation>** elements. All the other elements make use of these references by using references to these elements.

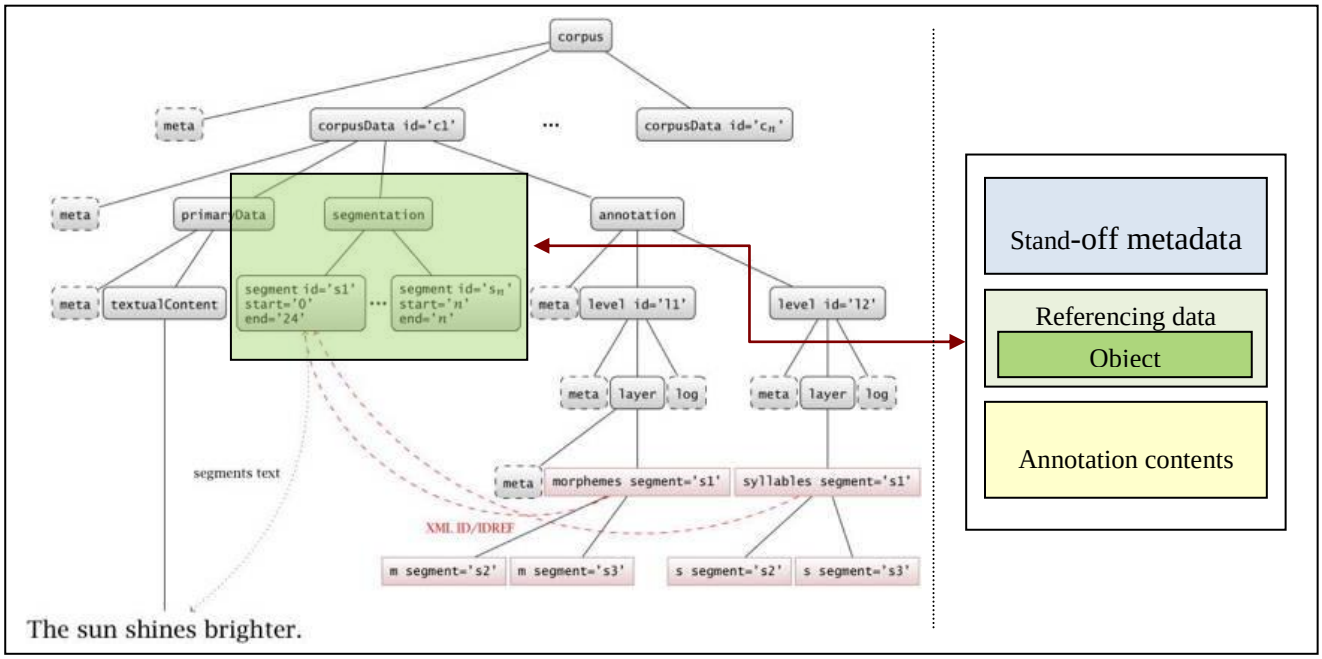

Fig. 6: XStandoff Conceptual Structure (Data Model)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsf:corpusData ...>
  <xsf:primaryData start="0" end="24" xml:lang="en">
    <textualContent>The sun shines brighter.</textualContent>
  </xsf:primaryData>
  <xsf:segmentation>
    <xsf:segment xml:id="seg1" type="char" start="0" end="24"/>
    <xsf:segment xml:id="seg2" type="char" start="0" end="3"/>
    <xsf:segment xml:id="seg3" type="char" start="4" end="7"/>
  </xsf:segmentation>
  <xsf:annotation>
    <xsf:level xml:id="l_morph">
      <xsf:layer xmlns:m="http://www.xstandoff.net/morphemes"...>
        <m:morphemes xsf:segment="seg1">
          <m:m xsf:segment="seg2"/>
        </m:morphemes>
      </xsf:layer>
    </xsf:level>
    <xsf:level xml:id="l_syll">
      <xsf:layer xmlns:s="http://www.xstandoff.net/syllables"...>
        <s:syllables xsf:segment="seg1">
          <s:s xsf:segment="seg2"/>
        </s:syllables>
      </xsf:layer>
    </xsf:level>
  </xsf:annotation>
</xsf:corpusData>
```

Fig.7: Identification of standoff alignments in a exemplary XStandoff encoding

- unconstrained use of stand-off references: in this encoding approach, the references to the source text are **not necessarily** grouped into a common place, but they can be used in different places depending on the encoding model. This approach is followed, for example, by the XCES encoding where no constraints are established regarding the location of the alignment information. For example, the following code shows multiple references associated with the annotation contents in the according to the xcesAna schema:

```
<cesAna xsi:schemaLocation="http://www.xml-ces.org/schema
   http://www.cs.vassar.edu/XCES/schema/xcesAna.xsd" version="1.0">
   <chunkList xml:base="exampleDoc.en.xml">
      <chunk xlink:href="#p1s1">
         <tok id="p1s1w1" xlink:href="#xpointer(string-range(id('p1s1'), '', 1, 5))">
            <lex>
               <base>it</base>
               <msd>Pp3ns</msd>
               <ctag>PPER3</ctag>
            </lex>
         </tok>
         <tok id="p1s1w2" xlink:href="#xpointer(string-range(id('p1s1'), '', 7, 5))">
            <orth>was</orth>
            <lex>
               <base>be</base>
               <msd>Vais1s</msd>
               <ctag>AUX1</ctag>
            </lex>
            <lex>
               <base>be</base>
               <msd>Vais3s</msd>
               <ctag>AUX3</ctag>
            </lex>
         </tok>
      </chunk>
   </chunkList>
</cesAna>
```

Fig.8: Identification of standoff alignments in a exemplary XCES encoding

Also with a XCES encoding, the following example, from the The Open American National Corpus, follows the same approach:

```
<?xml version="1.0" encoding="UTF-8"?>
<cesAna xmlns="http://www.xces.org/schema/2003" version="1.0.4">
  <struct type="cesDoc" from="0" to="65865">
    <feat name="xmlns" value="http://www.xces.org/schema/2003"/>
    <feat name="version" value="1.0.4"/>
  </struct>
  <struct type="text" from="1" to="65864"/>
  <struct type="body" from="2" to="65863"/>
  <struct type="div" from="3" to="65862">
    <feat name="type" value="article"/>
    <feat name="xml:lang" value="en-US"/>
  </struct>
  <struct type="p" from="4" to="719">
    <feat name="id" value="p1"/>
  </struct>
  ...
</cesAna>
```

Fig.9: Identification of standoff alignments in a exemplary XCES encoding from the ANC

Another example can be found in a TEI exemplary encoding of stand-off annotations disclosed in the TEI Wiki page related to the Stand-off use cases (http://wiki.tei-c.org/index.php/Stand-off_use_cases).

```
<?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>...</teiHeader>
  <text>
    <body>...</body>
    <back>
      <div type="persons">
        <listPerson>
          <person xml:id="tm_person_195364">
            <persName corresp="#string-range(//lb[@n='1'],3,6)"
              ref="http://www.trismegistos.org/name/3756">Κράτης</persName>
          </person>
          <person xml:id="tm_person_195365">
            <persName corresp="#string-range(//lb[@n='2'],0,9)"
              ref="http://www.trismegistos.org/name/2811">Διονύσιος</persName>
          </person>
        </listPerson>
      </div>
    </back>
  </text>
</TEI>
```

Fig.10: Identification of standoff references in a exemplary TEI encoding

Therefore, the proposed TEI encoding should be able to manage both approaches.

It should be noted that the attribute **@corresp** forms part of **att.global.linking** which is common to all the TEI elements through **att.global**. Therefore, independently of the exact encoding of the annotation content, the second approach could always be achieved by referencing the source data using the attribute **@corresp** of the specific annotation content element.
In order to provide a common place for encoding source references according to the first approach, it is suggested to add the elements **<linkGrp>** and **<ptr>** as childs of **<stf>**. The element **<linkGrp>** can contain the following elements:

- **<ptr>**: that allows to encode references to the source text
- **<link>**: that allows to combine groups of references to the source text

Thus, a partial (only the minimum set of attributes and a child element for encoding stand-off references) declaration of the **<std>** element is disclosed in the following figure:

```
element stf
{
   att.global.attributes,
   att.typed.attributes,
   att.datcat.attributes,
   att.ascribed. attributes,
   att.datable.w3c.attributes,
  (linkGrp ¦ ptr)*
   ...
}
```

Fig. 11: Partial declaration of the element **<stf>**

4.3. Encoding the stand-off annotation contents

In this case, the possibilities of possible encodings are as big as all the possible annotations, not only the linguistic ones that can be implemented on a text. Therefore, it is not reasonable to propose a general framework that would cover all the possibilities.

Nevertheless, to begin with, two are the possible TEI elements that are candidates to be child of the new element **<stf>**:

- **<label>**: this element contains any label or heading used to identify part of a text. It can be used for encoding very simple annotations on specific parts of the source text

- **<fs>**: (feature structure) represents a feature structure, that is, a collection of feature-value pairs organized as a structural unit. It can be used for encoding more complex annotations. The reasons for considering the feature structure as basis for encoding complex annotations are the following:

    - the TEI tag set for feature structures can be adopted to represent a heterogeneous set of linguistic corpora (Witt, Rehm, Hinrichs, Lehmberg and Stegmann 2009)

- the combination of feature structures with pointers has been already successfully used for implementing stand-off annotations in TEI (Przepiórkowski 2009, Banski and Przepiórkowski 2010)

- the feature structure has been already successfully used for encoding syntactic information in a way that maximizes the compatibility with other standards (Przepiórkowski 2009)

- the feature structure in TEI provides a mechanism for defining constraints on the structure and further information about the structure through the Feature System Declaration. This gives also the opportunity of further providing information about the annotation, additional to the information provided in the annotation metadata (see TEI Guidelines: 18. Feature Structures)

- the Feature System Declaration can be implemented in a stand-alone container <stdDecl>, independent of the <teiHeader> and the <text>. this would allow to clearly differentiate the annotation information from the information directly related to the source text, i.e. the header and the source text.

- the TEI feature structure allows to align both feature names and their values with standardized external data category repositories such as ISOcat (see TEI Guidelines: 18. Feature Structures)

- the TEI recommendations for feature structures have been adopted as ISO Standard 24610-1 Language Resource Management — Feature Structures — Part One: Feature Structure Representation

Therefore, it is suggested to add the elements **<fs>** and **<label>** as childs of **<stf>** for encoding the annotation contents.

A final declaration of the **<std>** element which would allow to encode all the basic information of the stand-off annotations as disclosed above would be:

```
element stf
{
   att.global.attributes,
   att.typed.attributes,
   att.datcat.attributes,
   att.ascribed. attributes,
   att.datable.w3c.attributes,
  ((linkGrp ¦ ptr), label, fs)*
}
```

Fig. 12: Final declaration of the element **<stf>**

5. Grouping the stand-off annotations: encoding the annotation's hierarchy

When dealing with the encoding of annotations, there is often a need of grouping the different annotations carried out on the source text. According to (Goecke, Lüngen, Metzing and Stührenberg 2010), a distinction should be made between two principally different ways of grouping units of information related to the annotations:

- Annotation level: refers to the conceptual level of information represented in markup, i.e. refers to a model involving theoretical concepts. Example of annotation level is, for example, the level of syntax, that analyses the source text according to different syntactic theories (e.g. Lexical Functional Grammar, Tree Adjoining Grammar, Categorical Grammar)
- Annotation layer: referring to the technical realisation of markup

In many other projects, the annotations are grouped in different levels. For example in (Banski and Przepiórkowski 2010), it is stated that:

> *"The following levels of linguistic annotation are distinguished in the project: 1) segmentation into sentences, 2) segmentation into fine-grained word-level tokens, 3) morphosyntactic analysis, 4) coarse-grained syntactic words (e.g., analytical forms, constructions involving bound words, etc.), 5) named entities, 6) syntactic groups, 7) word senses (for a limited number of ambiguous lexemes). "*

In other cases, it would be useful to group the annotations, not based on their conceptual meaning and the corresponding realization but based on other project specific requirements, like for example grouping the manually created annotations or the automatically created annotations.

Therefore, it seems clear that it would be useful to establish an encoding framework that would allow to group different sets of related annotations when encoding them under the TEI.

In order to accomplish this requirement, two further TEI elements are proposed:

- **<stand-off>**: parent element that works as a general container for all the stand-off annotation. The element **<stand-off>** should, at least, have the following attribute:
  - att.global (@xml:id): for encoding the unique id

- **<stfGrp>**: element for grouping a set of **<stf>** elements
  The element **<stfGrp>** should, at least, have the following attributes:
  - att.global (@xml:id): for encoding the unique id
  - att.typed (@type): for encoding the type of stand-off annotations encoded under the element, for example morphosyntactic, semantic, structural, relational
  - att.datcat (@datcat): for encoding the data category registry associated to the type of annotation specified in @type
  - att.ascribed (@who): for encoding the author of the group of stand-off annotations

- att.datable.w3c (@when): for encoding the creation date of the group of stand-off annotations

The element **<stand-off>** works as a general container for the stand-off annotations that are encoded under the child elements **<stf>**. The element **<stfGrp>**, also a child of **<stand-off>**, allows to group a set of stand-off annotations encoded under **<stf>**.
In order to provide maximum flexibility, similar to other TEI grouping elements, **<stfGrp>** is also a child of **<stf>**, allowing to create multiple hierarchies of stand-off annotations.

Therefore, the declaration of the newly proposed TEI elements is:

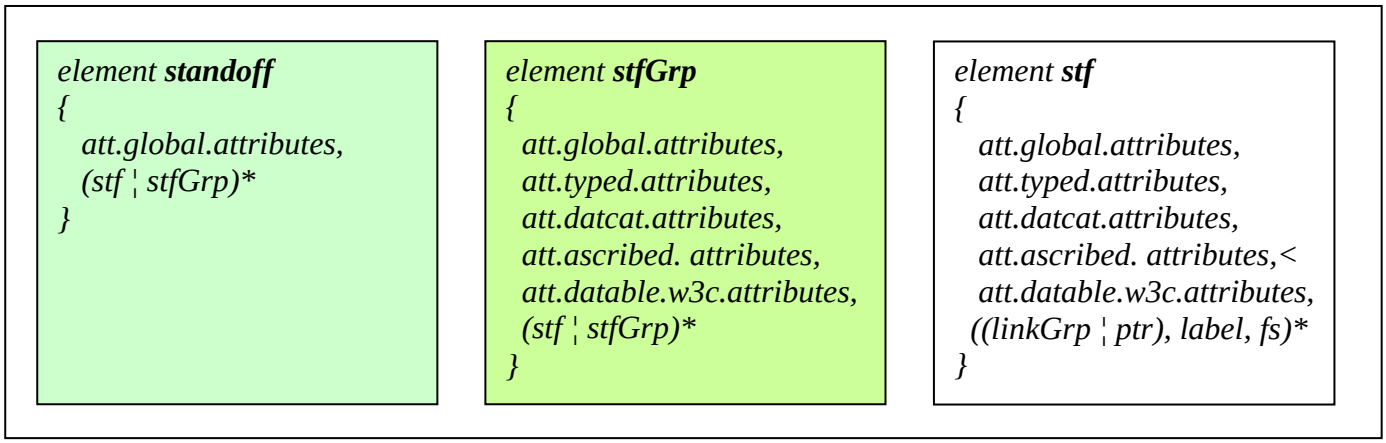

Fig. 13: Final declaration of the elements **<standoff>**, **<stfGrp>** and **<stf>**

The definition of these three elements allows to freely define any hierarchy of annotations when encoding them in the TEI:

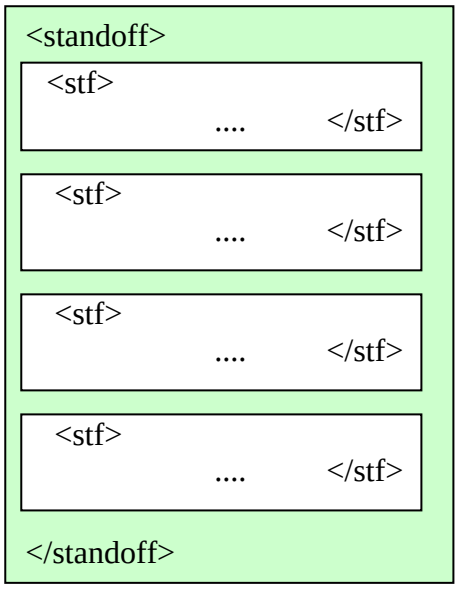

Fig. 14: Example of annotation hierarchy

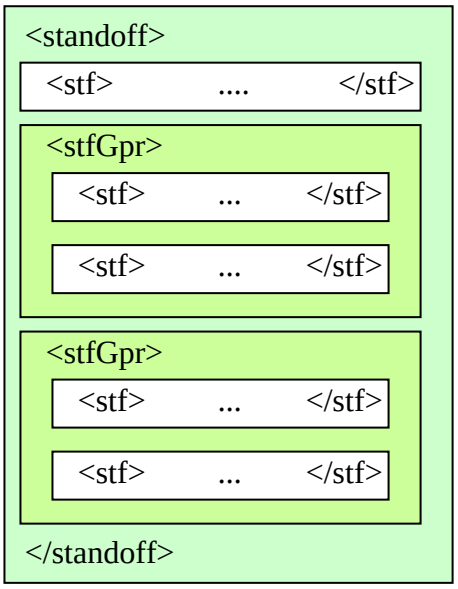

Fig. 15: Example of annotation hierarchy

5.1. Practical application of annotation hierarchies

In order to illustrate the flexibility of the proposed annotation framework, we will apply it to one specific case. In (Bański 2010), there are three different stand-off systems disclosed. In all of them, multiple layers of annotations are implemented on the source text. For simplicity, we will select the one used by the Open-Content Text Corpus (OCTC).

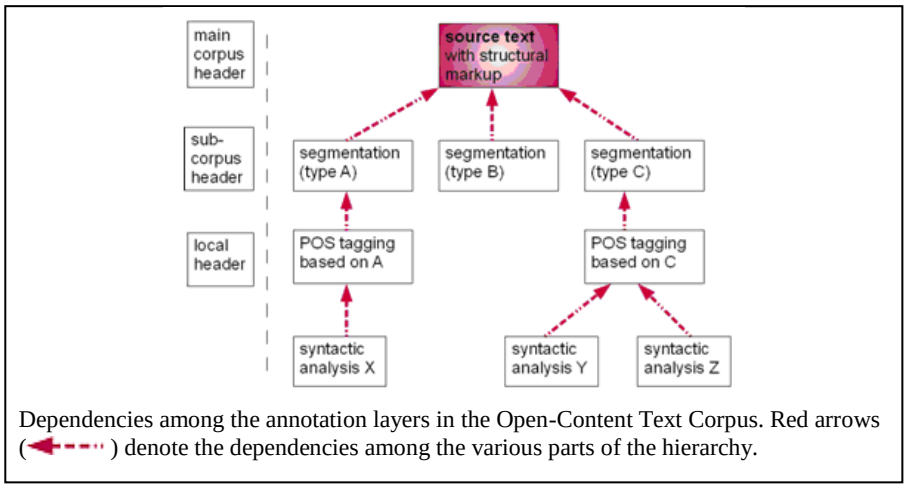

Fig. 16: Open-Content Text Corpus (OCTC) (Bański 2010)

Using the proposed encoding framework, there are different approaches that can be followed for encoding the stand-off annotations. We will disclose three of them for illustrative porpoises:

- "Dependency" hierarchy: in this case, the annotations are grouped based on their dependencies

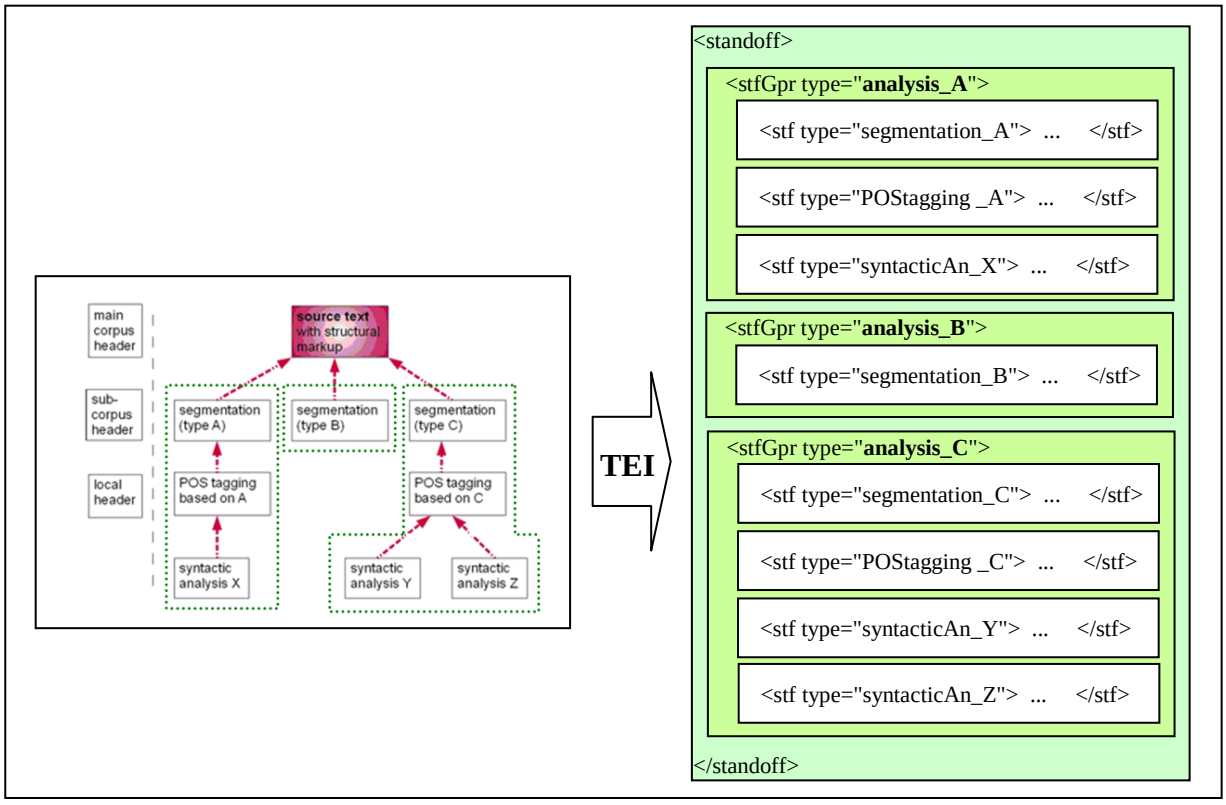

Fig. 17: "Dependency" hierarchy

- "Functional" hierarchy: in this case, the annotations are grouped based on their function

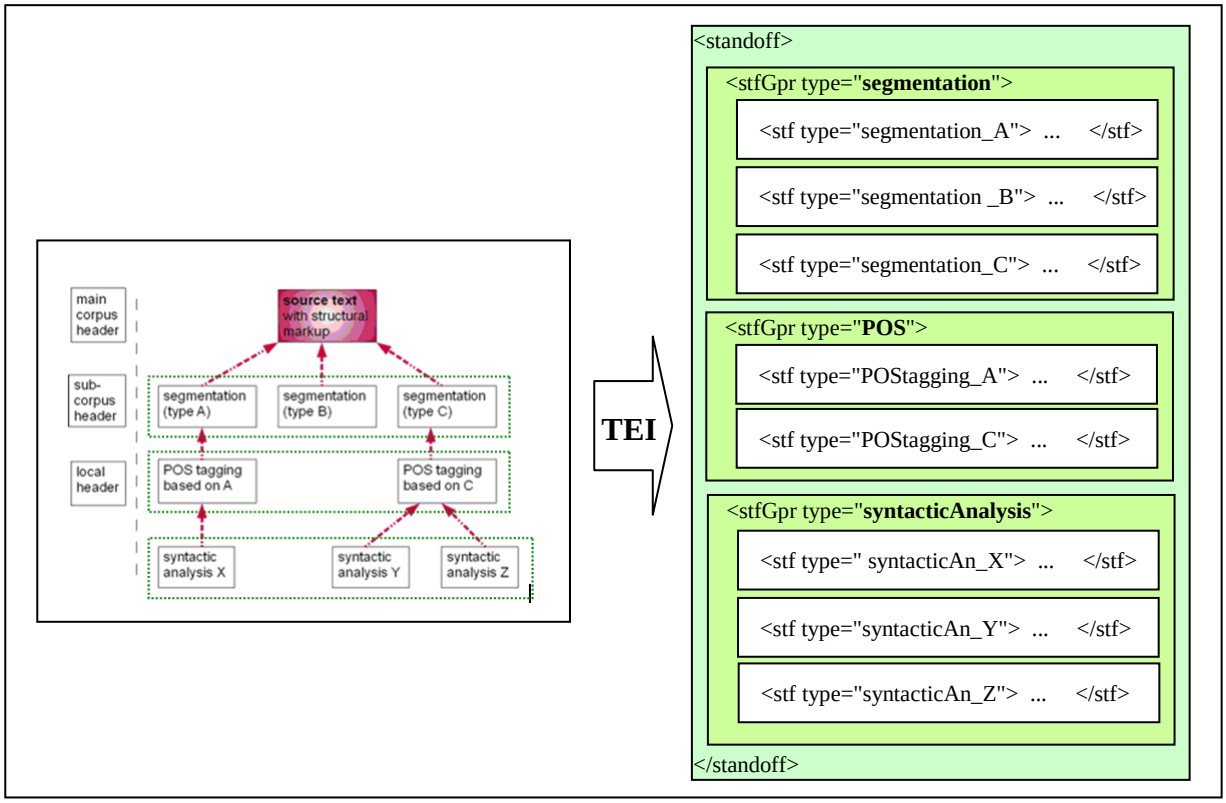

Fig. 18: "Functional" hierarchy

- No-hierarchy: in this case, the annotations are directly encoded under without defining any hierarchy

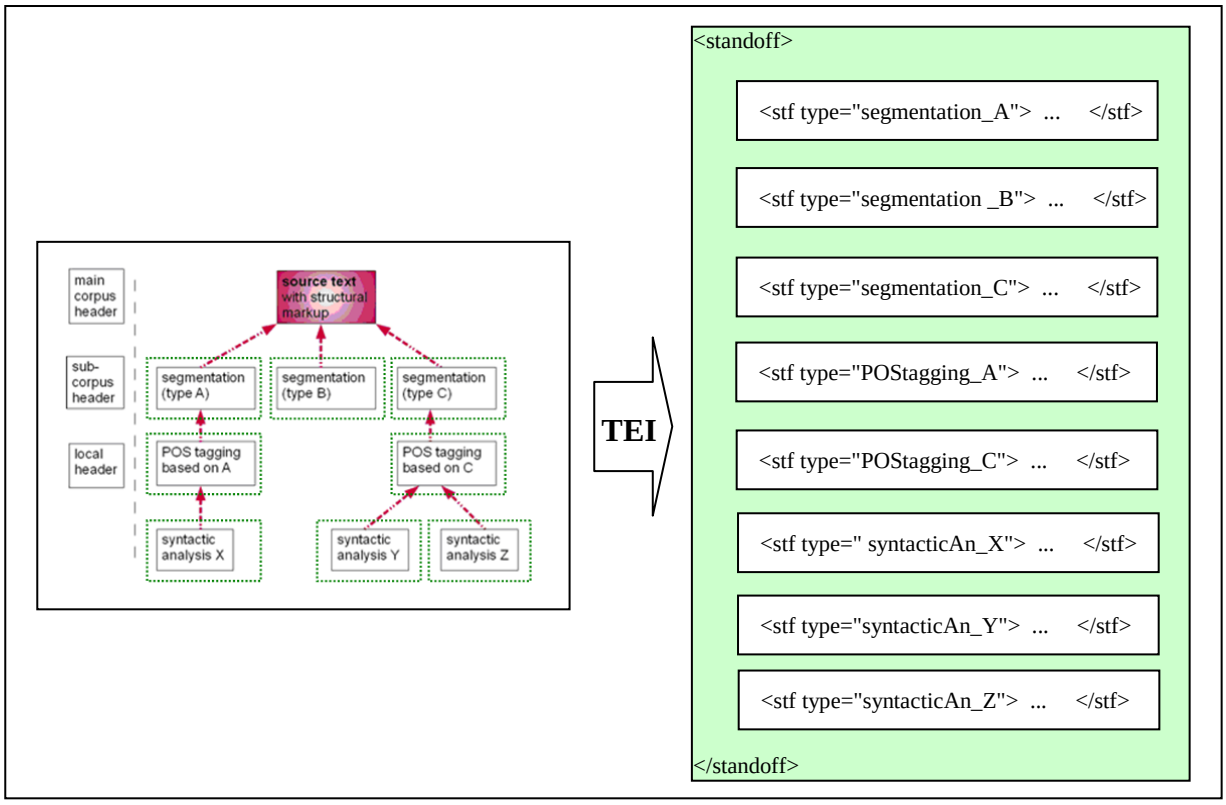

Fig. 19: No hierarchy

6. Using non-TEI schemas for encoding stand-off annotations

There are, in addition to the TEI, other text encoding standards, most of them providing already mechanisms for encoding annotations, like the standards developed within ISO TC 37 / SC4 -LAF, MAF, SynAF- (Ide and Romary 2006, Declerck 2008), GrAF (Ide and Suderman 2007), XCES (Ide, Bonhomme and Romary 2000), TIGER-XML (Mengel and Lezius, 2000), XStand-off (Stührenberg and Jettka 2009) or PAULA (Dipper 2005).

All these standards already provide a very well defined framework for encoding stand-off annotations. Therefore, one possible way of encoding stand-off annotations in TEI would be simply using the desired schemas inside the **<stf>** element making use of the corresponding namespaces. The ODD customization mechanism could be used for importing the corresponding non-TEI schema into a customized TEI schema.

The following figure discloses one case where the TIGER-XML standard has been used for encoding stand-off annotations on the source text:

```
<stf>
  <nt id="nt2" xmlns=" http://www.ims.uni-stuttgart.de/projekte/TIGER/public/TigerXML.xsd">
    <edge label="head" idref="#t6"/>
    <edge label="nonhead" idref="#nt20"/>
    <edge label="nonhead" idref="#nt21"/>
  </nt>
</stf>
```

Fig. 20: Using external schemas

7. Location of stand-off annotations in the TEI structure

Once the general framework for encoding stand-off annotations has been defined, it needs to be established where in the whole TEI structure said annotations (i.e. **<stand-off>**) must should be encoded.
This issue has been discussed and is still open in the TEI community (see http://sourceforge.net/p/tei/feature-requests/378/).

There are basically three alternatives:

- In the **<teiHeader>**: in this case the stand-off annotations would be encoded in the header of the TEI document
- In the **<text>**: in this case the stand-off annotations would be encoded in a specific container in the <text>, probably a ** ** element or in the **<back>** element. This last encoding approach was followed by the TEI exemplary encoding of stand-off annotations disclosed in the TEIWiki page related to the Stand-off use cases (http://wiki.tei-c.org/index.php/Stand-off_use_cases).
- In a new place between the **<teiHeader>** and **<text>**: in this case the stand-off annotations would be encoded as an independent "entity" between the <teiHeader> and the <text>.

In the present proposal, it is suggested to use the last option for introducing the new elements in the TEI document structure.

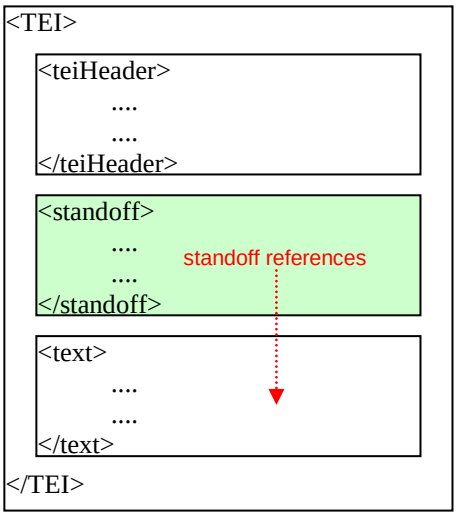

Fig. 21: Standoff annotations in the TEI document structure

Most of the reasons for this have been already indicated in the discussions carried out inside the TEI community (http://sourceforge.net/p/tei/feature-requests/378/).
Summarising, the reasons for this TEI document structure where the stand-off annotations are encoded as an independent "entity" not directly linked to the <teiHeader> or the <text> are the following:

- it allows to clearly differentiate between the source text and the annotations made on said text, that are clearly of different nature (probably different author, creation dates, meaning, function...)
- it facilitates the encoding and processing of the TEI documents because all the stand-off annotations are encoded in a clearly defined location inside the TEI structure
- It is inline with similar mechanisms embedding representations external to the text proper. The stand-off annotations, that are not a piece of the text itself, should be stored in a "separate" place, different than the text. This makes much more clear the nature of the information
- encoding stand-off annotations in <teiHeader>, or <back>, is contrary to the original porpoise of these elements and introduces confusion on the general encoding

8. Open issues

The present proposal represents only a first approach to the issue of encoding stand-off annotations in the TEI. There are a number of issues that should be discussed and agreed in the TEI community. Some of the open issues are the following:

- Mechanism for encoding the stand-off metadata: in the present proposal, only a very limited metadata information has been encoded, basically using the attributes of the TEI elements. Nevertheless, there is a lot of metadata information that could also be useful to encode and, probably, it would not be possible to do it by using only the attributes. Therefore, it could be

useful to think to what extend it could be defined a new mechanism for encoding stand-off metadata associated to the annotations.

- Elements for encoding stand-off contents: as in the previous case, the present proposal only discloses a minimal set of elements that could be useful for encoding the contents of the stand-off annotations. Nevertheless, there are some cases where this set of elements are not enough for encoding the contents. Two possible candidates for be included as child of **<stf>** would be: **<figure>** for annotating not only the textual information, but also of images and formulas and **<graph>** for encoding annotations according to graph theories.

9. Conclusion

The motivation for the present document was to propose a general framework for encoding the stand-off annotations in the TEI. Currently there is an urgent need in the TEI Guidelines to provide general guidance for encoding stand-off annotations. As it stands now, the TEI Guidelines provide the key elements for encoding the annotations, i.e. the linking mechanism and the analysis elements, but still lacks of a general guidance of how to use said elements in a consistent and predictable manner. In some way is like if the Guidelines provide the basic products for making a meal but does not provide the recipe for cooking it.

In order to provide such a general frame for encoding the stand-off annotations, three are the main proposals disclosed in the present document: (1) the introduction of three new TEI elements specifically defined for encoding stand-off information, (2) a general frame for allowing hierarchical encoding of the stand-off annotations, and (3) a container for the stand-off annotations independent of the header and the text of the TEI document.

References

[Bański 2010] P. Bański (2010). Why TEI stand-off annotation doesn't quite work: and why you might want to use it nevertheless. In Proceedings of Balisage: The Markup Conference, 2010. Vol. 5 of Balisage Series on Markup Technologies

[Bański and Przepiórkowski 2009] P. Bański and A. Przepiórkowski (2009). Stand-off TEI Annotation: the Case of the National Corpus of Polish. In Proceedings of the Third Linguistic Annotation Workshop, ACL-IJCNLP 2009, pp 64–67

[Bański and Przepiórkowski 2010] P. Bański and A. Przepiórkowski (2010). TEI P5 as a Text Encoding Standard for Multilevel Corpus Annotation. In Digital Humanities 2010 Conference Abstracts, pp. 98–100

[Boot 2009] P. Boot (2009). Towards a TEI-based encoding scheme for the annotation of parallel texts. In Literary and Linguistic Computing 24(3), pp. 347–361

[Burnard and Bauman 2008] L. Burnard and S. Bauman (2008). TEI P5: Guidelines for Electronic Text Encoding and Interchange. Oxford. http://www.tei-c.org/Guidelines/P5/

[Dipper 2005] S. Dipper. (2005). Stand-off representation and exploitation of multi-level linguistic annotation. In Proceedings of Berliner XML Tage 2005 (BXML 2005), pp. 39–50

[Goecke, Lüngen, Metzing and Stührenberg 2010] D. Goecke, H. Lüngen, D. Metzing and M. Stührenberg (2010). Different Views on Markup: Distinguishing Levels and Layers. In Linguistic Modeling of Information and Markup Languages, Text, Speech and Language Technology, pp. 1-21

[Ide, Bonhomme and Romary 2000] N. Ide, P. Bonhomme and L. Romary. (2000). XCES: An XML-based Encoding Standard for Linguistic Corpora. In Proceedings of the Second Language Resources and Evaluation Conference (LREC), Athens, pp. 825-830

[Ide, Romary and de la Clergerie 2003] N. Ide, L. Romary and E. de la Clergerie. (2003). In International standard for a Linguistic Annotation Framework, Proceedings of HLT-NAACL'03 Workshop on The Software Engineering and Architecture of Language Technology, Edmonton.

[Ide and Suderman 2007] N. Ide, K. Suderman (2007). GrAF: A Graph-based Format for Linguistic Annotations. In Proceedings of the Linguistic Annotation Workshop 2007, Prague, 1-8

[Mengel and Lezius, 2000] A. Mengel and W. Lezius. (2000). An XML-based encoding format for syntactically annotated corpora. In LREC 2000, pp 121–126.

[Przepiórkowski 2009] A. Przepiórkowski (2009). TEI P5 as an XML Standard for Treebank Encoding. In Proceedings of the Eighth International Workshop on Treebanks and Linguistic Theories (TLT8), pp. 149-160

[Przepiórkowski and Bański, 2009] A. Przepiórkowski and P. Bański (2009). "Which XML Standards for Multi-level Corpus Annotation?. In LTC 2009, pp. 400-411

[Sonderforschungsbereich632 2008] Sonderforschungsbereich/SFB 632 (2008). The PAULA Stand-off Format. Version 1.0. http://www.sfb632.uni-potsdam.de/en/paula.html

[Stührenberg and Jettka 2009] M. Stührenberg and D. Jettka (2009). A toolkit for multi-dimensional markup – the development of SGF to XStand-off. In Proceedings of Balisage: The Markup Conference 2009, Montreal.

[Witt, Rehm, Hinrichs, Lehmberg and Stegmann 2009] A. Witt, G. Rehm, E. Hinrichs, T. Lehmberg and J. Stegmann (2009). SusTEInability of linguistic resources through feature structures. In Literary and Linguistic Computing 2009