Extending standoff annotation

Using XStandoff 2 for multi-layer annotations of multimodal and pre-annotated documents

Maik Stührenberg

Institut für Deutsche Sprache
Mannheim, Germany
stuehrenberg@ids-mannheim.de

Abstract

Information encoding is often complex. Textual information is sometimes accompanied by additional encodings (such as visuals). These *multimodal documents* may be interesting objects of investigation for linguistics. Another class of complex documents are *pre-annotated documents*. Classic XML inline annotation often fails for both document classes because of overlapping markup. However, standoff annotation, that is the separation of primary data and markup, is a valuable and common mechanism to annotate multiple hierarchies and/or read-only primary data. We demonstrate an extended version of the XStandoff meta markup language, that allows the definition of segments in spatial and pre-annotated primary data. Together with the ability to import already established (linguistic) serialization formats as annotation levels and layers in an XStandoff instance, we are able to annotate a variety of primary data files, including text, audio, still and moving images. Application scenarios that may benefit from using XStandoff are the analyzation of multimodal documents such as instruction manuals, or sports match analysis, or the less destructive cleaning of web pages.

Keywords: Standoff annotation, XStandoff, Complex documents

1. Introduction: Complex documents

Information is often encoded in multiple ways, take directions for example. If you want to explain someone the way to a specific location, you could use textual directions, a map, or a combination of both – which is often the preferred way (think of popular web services). Another example is an operating manual. These *multimodal documents* can be interesting objects of investigation for linguistics, since they often not only include alternative encodings of the very same information but sometimes encode parts only in a single way (which may provide clues about the importance). In addition, information items (regardless of their encodings) may be connected through relations which may be of interest in linguistic analysis.

Up to now, standards for annotating text and images exist, with each schema rather specialized. Creating a corpus of multimodal documents could benefit from a unified annotation format supporting different types of primary data and supporting multiple annotation layers. XML inline annotation, that is elements (and attributes) that enclose the corresponding units of text (the primary data) and that are hierarchically ordered in a tree-like structure, often fails when dealing with multiple annotation layers, since these tend to overlap. A classic example is the annotation of syllables and morphemes of the sentence "The star shines brighter today". If you try to combine both annotation layers in a single inline annotation, you will end up with overlapping markup, since the token "brighter" can be annotated as morpheme "bright" and "er" and syllables "brigh" and "ter". The resulting document violates XML's wellformedness constraints (Bray et al., 2008).[1]

Another form of complex documents are *pre-annotated documents*. In general, most NLP tools tend to use raw text as input data instead of already annotated documents (although there are exceptions to this rule). However, even if tools may accept documents of a given markup language, the addition of supplemental annotation layers may fail to the very same aforementioned reason.

Standoff annotation, that is the separation of primary data and markup, is a valuable and common mechanism to annotate multiple hierarchies and/or read-only primary data. The concept as such is not new at all, early descriptions can be found in the TIPSTER project (Architecture Committee for the TIPSTER Text Phase II Program, 1996) while Thompson and McKelvie (1997) provide additional reasons for separating primary data and annotations such as overlapping annotations, and copyright restrictions. Examples of newer standoff formats are XCES (Ide et al., 2000), the XML version of the former SGML Corpus Encoding Standard (CES), PAULA (Dipper and Götze, 2005), the Graph Annotation Format (GrAF) as part of the International Standard ISO 24612:2012 (Linguistic Annotation Framework, LAF), and XStandoff (Stührenberg and Jettka, 2009; Stührenberg, 2013). XStandoff was introduced as an enhanced version of the Sekimo Generic Format (Stührenberg and Goecke, 2008). While the first version was developed with the goal of storing multiple annotations for automatically resolving anaphora resolution, XStandoff (internally numbered as version 1.1) subsequently added the XSF Toolkit, a set of XSLT 2.0 stylesheets that not only allowed for the easy management of XSF instances but introduced visualizations of markup overlaps as well (Jettka and Stührenberg, 2011). XStandoff 2, discussed in this article, is the first version that breaks backwards-compatibility (at least in some cases) for the benefit of a massively extended segmentation module (see Section 2.2.).

An XStandoff instance consists of the `corpusData` root element, underneath zero or more `primaryData` elements, a `segmentation`, and an `annotation` element can oc-

---

[1] The discussion of this issue goes back to the days of SGML, including a large number of proposals for supporting overlapping markup not cited here due to space restrictions.

cur (see Listing 1).[2] The two latter elements define two base constructs of standoff annotation formats: (1) the identification of regions of primary data that will be used as anchors for one or more annotations (segments in XStandoff terms), and (2) the way in which annotations are stored.

Listing 1: Structure of an XStandoff instance

```
<corpusData xml:id="c1">
 <primaryData xml:id="p1">
  <!-- reference to textual primary data -->
 </primaryData>
 <primaryData xml:id="p2">
  <!-- reference to multimodal primary data -->
 </primaryData>
 <segmentation>
  <!-- segments -->
 </segmentation>
 <annotation>
  <level>
   <layer>
    <!-- annotation layer 1-->
   </layer>
   <layer>
    <!-- annotation layer 2 (optional) -->
   </layer>
  </level>
  <!-- additional annotation level -->
 </annotation>
</corpusData>
```

2. Segmentation

There are various kinds of segments depending on the type of primary data that serves as the basis for an annotation. We will discuss some preliminary considerations in the following subsection before we cover XStandoff's corresponding serializations in Section 2.2..

2.1. Segments in theory...

In case of *textual primary data*, segments can be identified by delimiting the character stream by means of tokenization methods (for example, splitting text into sentences by detecting sentence boundaries, sentences into token, and tokens into characters). In the example segmentation given in Figure 1, each character takes not a position but a span, e. g. the capital letter "T" starts at position 0 and ends at position 1.

```
 T  h  i  s     i  s     a     w  o  r  d  .
00|01|02|03|04|05|06|07|08|09|10|11|12|13|14|15
```
Figure 1: Delimiting textual data

However, this mechanism can be error-prone when using pre-annotated primary data because of the handling of significant and insignificant white-space in XML editors. In this case, it is more promising to use the element node tree of an existing annotation as an initial traversal for the selection of the respective textual part – similar to the minimal segmentation demanded by the Corpus Encoding Standard (Ide, 1998).

In case of *audio or video primary data*, one can use points in time (or frame positions) as start and end anchors for segments – alternative segmentation methods are defined in the International Standard ISO/IEC 15938-2:2002 (MPEG7:2). Segments in *spatial primary data* can be seen as two- or three-dimensional objects defined by a set of coordinates

in space (and an optional primitive shape). Examples for mouse-driven tools that allow the definition of such areas are the Image Markup Tool (IMT)[3] and the Text-Image Linking Environment (TILE)[4]. Combining both segmentation methods allows for the annotation of parts of videos (for example actions of a single character over a period of time).

2.2. ...and in XStandoff

XStandoff has a limited but powerful set of pre-defined segment types that has been extended during the development of the current version 2.

Segments are construed by using the `segment` element and may have overlapping boundaries without the need of finer-grained sub-segments (as in LAF). The type of segment is defined by several attributes, which are demonstrated below. A textual segment containing the word "This" of the text given in Figure 1 can be seen in Listing 2.[5]

Listing 2: Textual segment

```
<segment xml:id="seg1" primaryData="txt" type="char"
    start="0" end="4"/>
```

A spatial segment referring to a specific region of an image can be seen in Listing 3. There are already several specifications that allow the definition of (geo-)spatial regions, including ISOSpace (Lee, 2013), TEI's `surface` and `zone` elements (Burnard and Bauman, 2014), METS/ALTO (McDonough, 2006), or HTML's image map. While TEI's elements use explicit `ulx`, `uly`, `lrx`, and `lry` attributes to store the coordinates of the upper left and lower right position of the corners of a rectangle, HTML's image map uses `shape` and `coords` attributes as part of the `area` element.

Listing 3: Spatial segment

```
<segment xml:id="seg2" primaryData="img" type="spatial"
    shape="poly" coords="30,6 60,6 60,32 30,32"/>
```

The `coords` attribute is used to store a set of two- or three-dimensional coordinates in space (there is a `context` attribute which is not shown in the example and has a default value of *2d*) to create a polygon shape (a line virtually connects the last coordinate with the starting one)[6]. Note that the coordinates are stored in clockwise order, starting with the upper left coordinate. In this case we define a polygon shape starting at the coordinates x_1=30, y_1=6, x_2=60, y_2=6, etc. There is a virtual line from the last coordinate to the first, closing the polygon.

Segments in time over audio or video primary data files can be instantiated similar to those over textual primary data by

---

[3]See http://www.tapor.uvic.ca/~mholmes/image_markup/ for further details.

[4]See http://mith.umd.edu/tile/ for further details.

[5]The `primaryData` attribute connects the segment and the corresponding primary data file.

[6]In addition to the *poly* value of the `shape` attribute, the other allowed values are *circle* and *bezir*. While circles are defined by a single triple of values *x* (center coordinate, x-axis), *y* (center coordinate, y-axis), and *r* (radius), the definition of a cubic Bézir curve follows SVG's syntax (Dahlström et al., 2011, Section 8.3.6, "The cubic Bézier curve commands").

using the `start` and `end` attributes (see Listing 4).[7]

Listing 4: Multimedia segment

```
<segment xml:id="seg3" start="00:00:00" end="00:02:00"/>
```

By combining spatial and temporal segmentation it is possible to annotate parts of a moving image, for example a certain character in a movie (see Listing 5). In this example we want to annotate an object in space which stays at the same coordinates during the time period starting at time code 00:00:00 and ending at 00:02:00.

Listing 5: Combined spatial-temporal segment

```
<segment xml:id="seg4" type="spatial" shape="poly"
 coords="0,10 20,50 0,2" start="00:00:00"
 end="00:02:00"/>
```

Since objects usually change their position over time, we have to use a slightly different approach. XStandoff supports the instantiation of segments by referring to already available ones. We can use this mechanism to express the movement of spatial segments over time.

Listing 6: Segment referring to already-instantiated segments

```
<segment xml:id="seg5" type="spatial" context="2d" shape
    ="poly" coords="0,10,30 100,150,30 0,200,30
    0,100,30"/>
<segment xml:id="seg6" type="spatial" context="2d" shape
    ="poly" coords="10,10,40 110,150,40 10,200,40
    110,100,40"/>
<segment xml:id="seg7" type="seg" segments="seg5 seg6"
    name="Bob" mode="continuous" start="00:00:00" end
    ="00:01:15"/>
```

In this example, the segments "seg5" and "seg6" both depict spatial segments. By creating the segment "seg7" as a reference to both former segments (defined by the value *seg* of the `type` attribute), we can express that the object named "Bob" has moved during the timespan from 00:00:00 to 00:01:15 from the coordinates given in "seg5" to the coordinates given in "seg6".[8].

As an alternative, XStandoff's logging functionality, which has been changed compared to previous versions, can be used as well (see Listing 7). From XStandoff 2 onwards, it is possible to insert `update` and `delete` elements underneath `segment` element and elements of imported layers (see Section 3.).

Listing 7: XStandoff's update mechanism

```
<segment xml:id="seg8" type="spatial" context="3d" shape
    ="poly" coords="0,10,30 100,150,30 0,200,30
    0,100,30" start="00:00:00">
 <update>
  <segment coords="10,10,40 110,150,40 10,200,40
     110,100,40" end="00:01:15"/>
 </update>
</segment>
```

Although this construct may seem a bit awkward at first, it may be easier to realize than ISO-Space's `MOVELINK` element demonstrated in Lee (2013).

Segmentation of a pre-annotated primary data file can be seen in Listing 8. We use the XHTML file shown in Listing 8 as an example, from which we select the first `div` element.

Listing 8: Example XHTML instance

```
<html xmlns="http://www.w3.org/1999/xhtml">
 <head><title>Instance</title></head>
 <body><div>This is a word.</div></body>
</html>
```

In theory, two different mechanisms can be used, either XPointer's `xpointer()` scheme (DeRose et al., 2002) or XPath 2.0 (Berglund et al., 2010) expressions. The specification of the former has not been finalized and has been in the state of a W3C Working Draft since 2002. Since it is not broadly supported either, we have chosen XPath 2.0 expressions (Listing 9).

Listing 9: Using XPath 2.0 for pre-annotated documents

```
<segment xml:id="seg_html1" target="xhtml:html/xhtml:
    body/xhtml:div[1]"/>
```

If we want to extract substrings of the `div` element's textual content, we can use XPath 2.0's `substring()` function defined in Malhotra et al. (2010) as shown in Listing 10.[9]

Listing 10: XPath 2.0's `substring()` function

```
<segment xml:id="seg_html2" target="xhtml:html/xhtml:
    body/substring(xhtml:div[1],1,4)"/>
```

Since every `segment` element has a unique identifier, it can be used as an anchor for one or more annotation elements which are stored underneath a specific `layer` element.[10]

3. Storing Annotations

While some standoff formats (for example, the aforementioned PAULA) store primary data and annotation levels in separate files, others such as LAF/GrAF store annotations transformed as feature structures, that is, element names are converted to attribute values. In contrast, XStandoff tries to be as integrated and to stick as close to the original inline annotation format as possible. The names of the elements and attributes and the tree-like structure remain unchanged. However, textual content of elements is deleted since it can be referenced via the corresponding segment, and additional attributes are added.

Each annotation level (i. e. POS tagging) is stored underneath the `level` element's `layer` element (i. e. the output of Stanford's POS-Tagger), no restrictions apply about the numbers of annotation levels and layers. Since XStandoff supports multiple primary data files, textual and spatial annotations of multimodal documents such as web pages (one primary data reference for the HTML page and embedded images, respectively) is possible. It is even possible to

---

[7]The segmentation unit of a primary data file is provided via a `unit` attribute of the `primaryData` element. XStandoff supports a wide range of pre-defined units, including chars, bytes, seconds, minutes, frames, amongst others.

[8]Note that the movement defined here is a linear one, up to now there is no inherent mechanism to describe a non-linear movement, although it is possible to use the `segment` element's optional metadata in combination with the `mode` attribute's "disjoint" value, see (Stührenberg and Jettka, 2009).

[9]Note that the `substring()` function uses "1" as start value instead of XStandoff's "0".

[10]For a discussion of the differentiation between the conceptual level and the serialization layer see (Witt, 2004).

use a single annotation layer to annotate items present in the textual and/or multi-modal primary data as a very shallow form of text-to-image mapping by referring to multiple segments at once.[11] Listing 11 shows such an example in which we have two primary data files (a PNG image and a TXT), which both contain the word "the" (non-XStandoff elements and attributes are highlighted in red, while references between XML `xs:ID` and `xs:IDREF` data types are highlighted in blue). The original inline annotation over the textual primary data file which has been converted to the corresponding XStandoff annotation layer is as follows:

```
<word xml:lang="en" wordform="the">the</word>
```

Listing 11: XStandoff instance for annotation based on textual and spatial primary data

```
<corpusData xml:id="x1"
 xmlns="http://www.xstandoff.net/2009/xstandoff/1.1"
 xmlns:xsf="http://www.xstandoff.net/2009/xstandoff/1.1"
 xmlns:dcr="http://www.isocat.org/ns/dcr">
 <primaryData xml:id="img">
  <primaryDataRef uri="img.png" mimeType="image/png"
   width="968" height="429" horizontalResolution="72"
   verticalResolution="72"/>
 </primaryData>
 <primaryData xml:id="text" unit="chars" xml:lang="en"
  xml:space="preserve">
  <primaryDataRef uri="text.txt" start="0" end="15"
   mimeType="text/plain"/>
 </primaryData>
 <segmentation>
   <segment xml:id="spatial-seg1" type="spatial"
    shape="poly" coords="30,6 60,6 60,32 30,32"
    primaryData="img"/>
   <segment xml:id="text-seg1" type="char" start="0"
    end="3" primaryData="text"/>
 </segmentation>
 <annotation>
  <level>
   <layer xmlns:w="http://www.xstandoff.net/examples/
       word">
    <w:word xml:lang="en"
     xsf:segment="spatial-seg1 text-seg1"
     wordform="the"
     dcr:datcat="http://www.isocat.org/datcat/DC-1415" />
   </layer>
  </level>
 </annotation>
</corpusData>
```

Since the naming of the elements underneath the `layer` element is not restrained, it is highly recommended to use the `datCat` and `valueDatCat` attributes borrowed from TEI's current version, which are added to the imported annotation layer via XStandoff's import mechanism to link the respective element to a data category in ISOcat[12], following the (ISO/TC 37/SC 3, 2009) standard and providing semantic concepts. By using already-established markup languages as annotation layers, one is able to adapt these to the primary data file types supported by XStandoff (e.g. using the TEI to annotate video primary data). Multiple annotation layers (and levels respectively) can be added to an existing XStandoff instance any time by using the accompanying XSF Toolkit (see Stührenberg and Jettka (2009)

for a detailed discussion about the transformation of inline annotation to XStandoff).[13]

4. Alternative standoff approaches

As already mentioned in Section 1. the standoff approach is a proven concept that has been used in a variety of formats, especially when dealing with multiply annotated corpora. Apart from XCES, which has been adapted to be fully compatible with the current version P5 of the TEI in the form of IDS-XCES (Lüngen and Sperberg-McQueen, 2012) which is used for the German Reference Corpus (DEREKO, Deutsches Referenzkorpus, see Kupietz and Lüngen *in this volume*), another prominent candidate for a standoff serialization format supporting multiply annotations is the Graph Annotation Format, described as a pivot format of the international standard ISO 24612:2012. While the former does not provide any segmentation mechanism over multimodal primary data, GrAF allows different segmentation types in principle but the handling of the string values used to define the anchors of a segment is left to the processing applications. Although pre-annotated primary data is supported, the preferred primary data format is raw text. When dealing with pre-annotated primary data files, one may choose to include the markup as part of the character stream when calculating character positions, or to use a combination of an XPath 2.0 expression (to select the element containing the text) and an offset to select the corresponding part of the character string. XPath 2.0's `substring()` function shown in Listing 10 is not officially supported, although this may be application-dependent.

The annotation has to be converted into a feature structure format that is similar to the serialization defined in ISO 24610-1:2006 and Chapter 18 of the TEI P5 (Burnard and Bauman, 2014).[14] Although converting existing inline annotation is possible, it is a more complex task and the resulting subtrees may become quite large (see Stegmann and Witt (2009) for a discussion of using TEI feature structures as serialization format for multiple annotated XML instances).

The TEI itself has made substantial progress in supporting standoff annotation, and current development driven especially by the TEI LingSIG (the special interest group "TEI for Linguists") tend to further improve this support. But as (Bański, 2010) already sketched out, there are still a number of issues, including technical problems and social issues.

A more elaborated discussion on some of the formats already mentioned can be found in Przepiórkowski and Bański (2011).

5. Application scenarios and conclusion

While the example shown above may seem arbitrary, it can be used to demonstrate possible application scenarios for

---

[11]It is recommended to provide more detailed relationship information as an additional annotation layer. Note that we do not make any assumptions about the differences between a facsimile and its various electronic representations (either image or text) in the example due to space constraints.

[12]See http://www.isocat.org for further details.

[13]XStandoff's document grammar and the Toolkit files are freely available under the GNU Lesser General Public License (LGPL v3) online at http://xstandoff.net.

[14]While the standardized format should be used to describe complex feature structures, simpler ones can be serialized by a GrAF-internal format. For a more elaborate discussion about the TEI's influence on current standards see Stührenberg (2012).

XStandoff 2.0, namely the analysis of multimodal documents (Baldry and Thibault, 2006; Bateman, 2008), that is, documents that contain information encoded both in textual and pictorial form. In this emerging field neither texts nor images are analyzed in an isolated way but together with respect to their very own communicative aspects and methods. Apart from the examples already mentioned in the Introduction, another one is the analysis of a sports match such as the one presented at `http://spielverlagerung.de/category/english/`. Using XStandoff for these kind of documents, it is possible to annotate relationships between textual and multimedia-based information.[15] Based on this representation one may differentiate between information available only in textual or graphical form by filtering for specific segment types. Additional annotation layers could store explicit co-referential relations between textual and graphical elements, which in turn may be an entry point for analyzing multimodal documents.

For web corpora, standoff annotation can be a means to a less destructive cleaning of primary data files, especially when they support pre-annotated files as input instances (Stührenberg, 2014). Similar to that, XStandoff can be helpful in comparing annotation layers (by annotating the same primary data with different markup languages and including them as different annotation layers in the same XStandoff instance) or judging the quality of NLP tools such as parsers and taggers.[16] Such instances can be easily evaluated by using standard XPath or XQuery queries.

In the meantime, XStandoff may serve as a link between standoff annotation and already established annotation formats in the fields of Linguistics and Digital Humanities. While the format as such is quite stable, further development has to be undertaken regarding an integrated web-based annotation tool capable of supporting various types of primary data and opening up a new way of collaboratively editing standoff annotation.

6. References

Architecture Committee for the TIPSTER Text Phase II Program. (1996). Tipster text phase ii architecture concept. Technical report, National Institute of Standards and Technology.

Baldry, A. and Thibault, P. J. (2006). *Multimodal Transcription and Text Analysis: A multimodal toolkit and coursebook*. Equinox Textbooks and Surveys in Linguistics. Equinox, London.

Bateman, J. A. (2008). *Multimodality and Genre. A Foundational for the Systematic Analysis of Multimodal Documents*. Palgrave Macmillan, New York, NY.

Bański, P. (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*, volume 5 of *Balisage Series on Markup Technologies*, Montréal.

Berglund, A., Boag, S., Chamberlin, D., Fernández, M. F., Kay, M., Robie, J., and Siméon, J. (2010). XML Path Language (XPath). Version 2.0 (Second Edition). W3C Recommendation, World Wide Web Consortium (W3C).

Bray, T., Paoli, J., Sperberg-McQueen, C. M., Maler, E., and Yergeau, F. (2008). Extensible Markup Language (XML) 1.0 (Fifth Edition). W3C Recommendation, World Wide Web Consortium (W3C), 11.

Burnard, L. and Bauman, S., editors. (2014). *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. Text Encoding Initiative Consortium, Charlottesville, Virginia, 1. Version 2.6.0. Last updated on 20th January 2014, revision 12802.

Dahlström, E., Dengler, P., Grasso, A., Lilley, C., McCormack, C., Schepers, D., and Watt, J. (2011). Scalable Vector Graphics (SVG) 1.1 (second edition). W3C Recommendation, World Wide Web Consortium (W3C).

DeRose, S. J., Maler, E., and Daniel, R. J. (2002). XPointer xpointer() scheme. W3C Working Draft, World Wide Web Consortium (W3C).

Dipper, S. and Götze, M. (2005). Accessing heterogeneous linguistic data – generic xml-based representation and flexible visualization. In *Proceedings of the 2nd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pages 206–210, Poznan.

Ide, N. M., Bonhomme, P., and Romary, L. (2000). XCES: An XML-based Encoding Standard for Linguistic Corpora. In *Proceedings of the Second International Language Resources and Evaluation (LREC 2000)*, pages 825–830, Athen. European Language Resources Association (ELRA).

Ide, N. M. (1998). Corpus Encoding Standard: SGML Guidelines for Encoding Linguistic Corpora. In *Proceedings of the First International Language Resources and Evaluation (LREC 1998)*, pages 463–470, Granada. European Language Resources Association (ELRA).

ISO/IEC JTC 1/SC 29. (2002). Information technology — Multimedia content description interface – Part 2: Description definition language. International Standard ISO/IEC 15938-2:2002, International Organization for Standardization, Geneva.

ISO/TC 37/SC 3. (2009). Terminology and other language and content resources — Specification of data categories and management of a Data Category Registry for language resources. International Standard ISO 12620:2009, International Organization for Standardization, Geneva.

ISO/TC 37/SC 4/WG 1. (2006). Language resource management — Feature Structures – Part 1: Feature Structure Representation. International Standard ISO 24610-1:2006, International Organization for Standardization, Genf.

ISO/TC 37/SC 4/WG 1. (2012). Language resource management — Linguistic annotation framework (LAF). International Standard ISO 24612:2012, International Organization for Standardization, Geneva.

Jettka, D. and Stührenberg, M. (2011). Visualization of concurrent markup: From trees to graphs, from 2d to 3d. In *Proceedings of Balisage: The Markup Conference*, volume 7 of *Balisage Series on Markup Technologies*, Montréal.

---

[15]An example can be found in Stührenberg (2013).

[16]Examples can be found at `http://xstandoff.net/examples/`.

Kupietz, M. and Lüngen, H. (to appear). Recent developments in DeReKo. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*.

Lee, K. (2013). Multi-layered annotation of non-textual data for spatial information. In Bunt, H., editor, *Proceedings of the 9th Joint ISO - ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 15–23, Potsdam.

Lüngen, H. and Sperberg-McQueen, C. M. (2012). A TEI P5 Document Grammar for the IDS Text Model. *Journal of the Text Encoding Initiative*, (3), 11.

Malhotra, A., Melton, J., Walsh, N., and Kay, M. (2010). XQuery 1.0 and XPath 2.0 Functions and Operators (Second Edition). W3C Recommendation, World Wide Web Consortium (W3C).

McDonough, J. (2006). METS: standardized encoding for digital library objects. *International Journal on Digital Libraries*, 6:148–158.

Przepiórkowski, A. and Bański, P. (2011). Which xml standards for multilevel corpus annotation? In Vetulani, Z., editor, *Human Language Technology. Challenges for Computer Science and Linguistics – 4th Language and Technology Conference, LTC 2009, Poznan, Poland, November 6-8, 2009, Revised Selected Papers*, volume 6562 of *Lecture Notes in Computer Science*, pages 400–411. Springer, Berlin and Heidelberg.

Stegmann, J. and Witt, A. (2009). TEI Feature Structures as a Representation Format for Multiple Annotation and Generic XML Documents. In *Proceedings of Balisage: The Markup Conference*, volume 3 of *Balisage Series on Markup Technologies*, Montréal.

Stührenberg, M. and Goecke, D. (2008). SGF – an integrated model for multiple annotations and its application in a linguistic domain. In *Proceedings of Balisage: The Markup Conference*, volume 1 of *Balisage Series on Markup Technologies*, Montréal.

Stührenberg, M. and Jettka, D. (2009). A toolkit for multidimensional markup: The development of SGF to XStandoff. In *Proceedings of Balisage: The Markup Conference*, volume 3 of *Balisage Series on Markup Technologies*, Montréal.

Stührenberg, M. (2012). The TEI and Current Standards for Structuring Linguistic Data. *Journal of the Text Encoding Initiative*, (3), 11.

Stührenberg, M. (2013). What, when, where? Spatial and temporal annotations with xstandoff. In *Proceedings of Balisage: The Markup Conference*, volume 10 of *Balisage Series on Markup Technologies*, Montréal.

Stührenberg, M. (2014). Less destructive cleaning of web documents by using standoff annotation. In *Proceedings of the 9th Web as Corpus Workshop (WAC-9)*, Gothenburg, Sweden.

Thompson, H. S. and McKelvie, D. (1997). Hyperlink semantics for standoff markup of read-only documents. In *Proceedings of SGML Europe '97: The next decade – Pushing the Envelope*, pages 227–229, Barcelona.

Witt, A. (2004). Multiple hierarchies: New Aspects of an Old Solution. In *Proceedings of Extreme Markup Languages*, Montréal.