Primer for ISO-TEI "Transcriptions of Spoken Language"

Thomas Schmidt, …

Content

Introduction

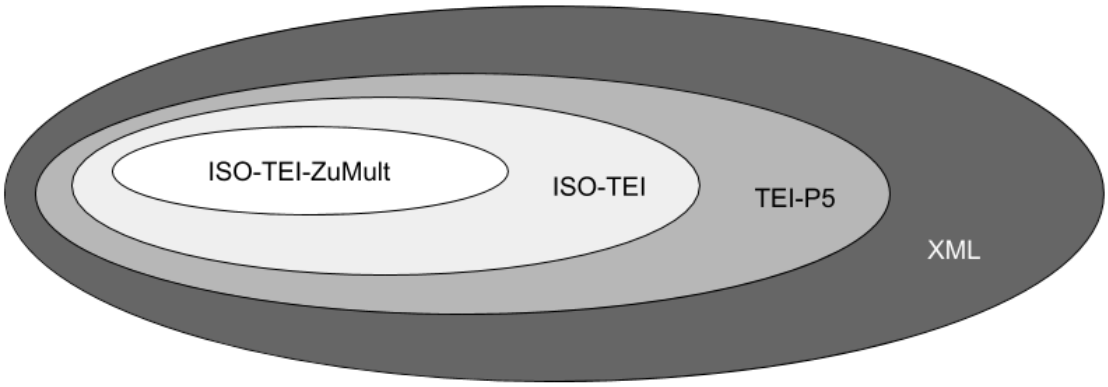
Related Work and References

ISO 24624:2016 Language resource management -- Transcription of spoken language

TEI-P5

CaC 2016

Specification of the format



TEI-P5 specifications

ISO specifications

[This is mostly a verbatim copy of parts of chapter 2 of Schmidt et al. (2017) with some changes and rearrangements for better readability]

The general focus of the standard is on orthography-based (i.e. not: IPA-based) transcription of recordings of authentic interaction (i.e. not: monologic "speech" or experiment data). Guiding design principles were the maxim to reuse as many elements as possible from chapter 8 of the existing TEI guidelines[1] and to orient their use towards interoperability with established tools. In particular, this meant a conscious limitation of choices in the numerous cases where the guidelines offer more than one solution for representing one and the same concept.

Basic building blocks for the document structure are

(a) one or more **<recording>** elements specifying the underlying audio and/or video file(s),

```
<sourceDesc>
    <recordingStmt>
        <recording type="audio">
            <media mimeType="audio/wav"
                url="file:/corpus/media/interaction_101.wav"/>
        </recording>
    </recordingStmt>
</sourceDesc>
```

(b) a **<particDesc>** defining the participants of the interaction, and

```
<profileDesc>
    <particDesc>
        <person xml:id="MJ" n="Mick" sex="1"/>
        <person xml:id="KR" n="Keith" sex="1"/>
    </particDesc>
    <!-- [...] -->
</profileDesc>
```

[1] See: http://www.tei-c.org/Vault/P5/1.0.0/doc/tei-p5-doc/en/html/TS.html.

(c) a <timeline> providing offsets into a recording.

```
<timeline unit="s" origin="#T0">
    <when xml:id="T0"/>
    <when xml:id="T1" interval="0.90663" since="#T0"/>
    <when xml:id="T2" interval="2.60121" since="#T0"/>
    <!-- [...] -->
</timeline>
```

The main part of the document is then made up of a sequence of <u> elements. They correspond to individual speaker contributions and contain the actual transcription text, references to points in the timeline and to the respective speaker (@who). As illustrated in Figure 3, the standard allows different levels of detail for the actual markup of the transcription text. In the simplest case, a plain text string can be used, which is temporally aligned via mandatory @start and @end attributes of the <u> element.

```
<u who="MJ" start="#T0" end="#T2">
    I ((cough)) see a door. I (0.3) want to paint it (black/blue).
</u>
```

Intervening temporal alignment can be added in the form of additional <anchor> milestone elements whose @synch attribute refers to a point in the timeline.

```
<u who="MJ" start="#T0" end="#T2">
    I ((cough)) see a door.
    <anchor synch="#T1"/>
    I (0.3) want to paint it (black/blue).
</u>
```

The microstructure of the speaker contribution can be represented by inserting additional markup, most importantly <w> for word tokens, <pc> for punctuation characters, <pause> for pauses and <vocal> or <kinesic> for non-verbal phenomena like coughing or laughing (example 3).

```
<u who="MJ" start="#T0" end="#T2">
    <w>I</w>
    <vocal>
        <desc>cough</desc>
    </vocal>
    <w>see</w>
    <w>a</w>
    <w>door</w>
    <pc>.</pc>
    <anchor synch="#T1"/>
    <w>I</w>
    <pause dur="PT0.3S"/>
    <w>want</w>
    <w>to</w>
    <w>paint</w>
    <w>it</w>
    <unclear>
        <choice>
            <w>black</w>
            <w>blue</w>
        </choice>
    </unclear>
    <pc>.</pc>
</u>
```

Finally, segmentations of speaker contributions into units above the word level (the “sentence equivalents” of spoken language such as intonation units) can be represented by intervening **<seg>** elements (example 4). As we will discuss below, the additional markup below **<u>** is crucial for many automatic annotation methods. The examples in Figure 3 illustrate transcription proper, i.e. the direct representation in written form of what is heard or seen in the primary data.

```
<u who="MJ" start="#T0" end="#T2">
    <seg type="intonation-phrase" subtype="falling">
        <w>I</w>
        <vocal>
            <desc>cough</desc>
        </vocal>
        <w>see</w>
        <w>a</w>
        <w>door</w>
    </seg>
    <anchor synch="#T1"/>
    <seg type="intonation-phrase" subtype="falling">
        <w>I</w>
        <pause dur="PT0.3S"/>
        <w>want</w>
        <w>to</w>
        <w>paint</w>
        <w>it</w>
        <unclear>
            <choice>
                <w>black</w>
                <w>blue</w>
            </choice>
        </unclear>
    </seg>
</u>
```

Figure 1: <u> elements with different levels of internal markup

In order to represent additional annotations on that material, the ISO/TEI standard provides the possibility to introduce an arbitrary number of standoff annotation layers in **<spanGrp>** elements and to group these with the **<u>** element they belong to, using an **<annotationBlock>** element (see Banski et al., 2016). This mechanism is crucial also for storing annotations that result from automatic annotation methods in WebLicht. Figure 4 illustrates the annotation of an utterance with lemmas and part-of-speech tags.

```
<annotationBlock who="MJ" start="#T0" end="#T2" xml:id="ab1">
    <u xml:id="u1">
        <seg type="intonation-phrase" subtype="falling" xml:id="seg1">
            <w xml:id="w1">I</w>
            <vocal xml:id="voc1"><desc>cough</desc></vocal>
            <w xml:id="w2">see</w>
            <w xml:id="w3">a</w>
            <w xml:id="w4">door</w>
        </seg>
    </u>
    <spanGrp type="lemma">
        <span from="#w1" to="#w1">I</span>
        <span from="#w2" to="#w2">see</span>
        <span from="#w3" to="#w3">a</span>
        <span from="#w4" to="#w4">door</span>
    </spanGrp>
    <spanGrp type="pos">
        <span from="#w1" to="#w1">PPER</span>
        <span from="#w2" to="#w2">V</span>
        <span from="#w3" to="#w3">DET</span>
        <span from="#w4" to="#w4">NN</span>
    </spanGrp>
</annotationBlock>
```

Figure 2: <annotationBlock> grouping an <u> with standoff annotation (lemmatization and POS tagging) in <spanGrp>

With the same mechanism (figure 5), orthographically normalized forms can be assigned to transcribed forms when the latter do not follow standard orthography. This is the case, for instance, in many conversation analytic transcription systems that use "literary transcription" or "eye dialect" to represent actual pronunciations that deviate from the ones suggested by the orthographic form (as in "gotta" for "got to").

```
<annotationBlock who="CB" start="#T0" end="#T2" xml:id="ab1">
    <u xml:id="u1">
        <w xml:id="w1">sure</w>
        <w xml:id="w2">nuff</w>
        <w xml:id="w3">an</w>
        <w xml:id="w4">yes</w>
        <w xml:id="w5">I</w>
        <w xml:id="w6">do</w>
    </u>
    <spanGrp type="norm">
        <span from="#w1" to="#w1">sure</span>
        <span from="#w2" to="#w2">enough</span>
        <span from="#w3" to="#w3">and</span>
        <span from="#w4" to="#w4">yes</span>
        <span from="#w5" to="#w5">I</span>
        <span from="#w6" to="#w6">do</span>
    </spanGrp>
</annotationBlock>
```

Figure 5: <annotationBlock> grouping an <u> with standoff annotation (orthographically normalized forms) in <spanGrp>

ZuMult specifications

Metadata

In the ZuMult architecture, the TEI **<header>** of transcripts is not the primary place to document metadata about recorded interactions, recordings, participating speakers (or even about the transcript itself). Properties of objects such as interactions or speakers do not directly pertain to the transcript, and experience has shown that they are therefore best modelled independently of the transcript. The transcript header can then reference these external objects as appropriate (see next section). Cases which make it obvious that the transcript header is not the right place to model interaction, recording, or speaker properties are:

- A corpus can contain recordings of interactions without corresponding transcripts. In these cases, a "dummy" (i.e. empty) transcript would have to be created just to have a place for the metadata.
- A corpus can contain 1:n relations between interactions and speakers. This is the case whenever speakers participate in more than one interaction, for instance in every longitudinal language acquisition study, but also in ethnographic field studies where the same (group) of person(s) is recorded in different situations. Modelling speaker properties in the transcript header would mean repeating one and the same information (e.g. a speaker's date of birth or gender) at several places, and this redundancy, if not properly managed, risks causing inconsistencies in the data.
- A corpus can contain recordings whose transcription is distributed over more than one document, for instance when technical reasons limit the maximal size of transcription documents or when there are two different transcriptions for the same recording. As in the previous point, the resulting redundancies can cause inconsistencies.

In ZuMult, we therefore limit information in the header to those elements which are required to work with the transcript itself, while all other metadata is recorded in separate files. Figure x is an example of a header for a transcript of a telephone conversation.

```
<teiHeader>
    <fileDesc>
        <titleStmt>
            <title>FOLK_E_00084_SE_01_T_01_DF_01</title>
        </titleStmt>
        <sourceDesc>
            <recordingStmt>
                <recording type="audio">
                    <media mimeType="audio/wav"
                        url="../../media/audio/FOLK/FOLK_E_00084_SE_01_A_01_DF_01.WAV"/>
                </recording>
            </recordingStmt>
        </sourceDesc>
    </fileDesc>
    <profileDesc>
        <particDesc>
            <person xml:id="SPK1" n="FR">
                <idno type="AGD-ID">FOLK_S_00477</idno>
            </person>
            <person xml:id="SPK2" n="EG">
                <idno type="AGD-ID">FOLK_S_00478</idno>
            </person>
        </particDesc>
    </profileDesc>
    <encodingDesc>
        <appInfo>
            <application ident="FOLKER" version="1.1">
                <label>FOLK Editor</label>
                <desc>Transcription Tool providing a TEI Export</desc>
            </application>
        </appInfo>
        <transcriptionDesc ident="cGAT" version="2014"/>
    </encodingDesc>
</teiHeader>
```

Figure 3

The following information is recorded in this example:

- A **<titleStmt>**, including the **<title>** of the transcript. This can be used in displays of the transcript
- A **<sourceDesc>** with a **<recordingStmt>** where the latter contains one or more **<recording>** elements specifying, in an embedded **<media>** element, the URL(s) of the audio and/or video file(s) on which the transcript is based. This information is necessary to

calculate multimedia representations of the transcript (e.g. a HTML page with embedded `<audio>`), but also for certain processing steps (e.g. an automatic aligner will require both the transcript text and the audio file).

- A `<profileDesc>` with a `<particDesc>` where the latter lists all the `<person>` elements describing the speakers occurring in the transcript. The attribute `@xml:id` is required so that `@who` attributes of `<annotationBlock>` can refer to it via an IDREF (see below). The value of the `@n` attribute can be used in displays of the transcript as the speaker's abbreviation. Note that `@xml:id` and `@n` can, but need not be, identical. A subordinate `<idno>` element is used to reference the speaker's metadata in an external documentation (see above and next section).
- An `<encodingDesc>` containing, first, an `<appInfo>` giving information about the tool with which the transcript was created, and second, a `<transcriptionDesc>` identifying the transcription system ("convention") which was used in transcribing. The latter can be a necessary piece of information for calculating displays of the transcript (as far as this information is not already encoded in a `@rend` attribute, see below).

Pointers

In the TEI guidelines, pointers from one element in the document to another are typically realised via XPointers, for instance `<u who="#SPK0">` can point to `<person xml:id="SPK0"/>`. While the XPointer mechanism may have the advantage of being extendable to much more general cases, especially to the case in which an element points to something outside its "own" document, it also has two disadvantages:

- There is no easy way of checking via a schema whether the pointer's target actually exists.
- XPath's `id()` function cannot be used to retrieve a pointer's target

As we feel that these disadvantages outweigh the loss of flexibility (we currently see no need, for instance, for external pointers from the transcript body), we generally define document-internal pointers as IDREFs. Practically, this simply means that we omit the leading '#' in the pointers. Figure XY provides examples for the most important ID/IDREF pairs:

```
<!-- speaker references -->
<person xml:id="SPK0"/>
<!-- [...] -->
<annotationBlock who="SPK0"/>

<!-- time references -->
<when xml:id="T1"/>
<anchor synch="T1"/>
<annotationBlock start="T0" end="T1"/>

<!-- annotation references -->
<w xml:id="w3">Bobbele</w>
<spanGrp type="NamedEntities">
    <span from="w3" to="w3">Boris Becker</span>
</spanGrp>
```

Still, we need a mechanism to point from elements in the header to information encoded in external places (see above). Since ZuMult's architecture requires such information to have a unique ID (within its "institution"), the `<idno>` element, defined in the TEI guidelines as "(identifier) supplies any form of identifier used to identify some object, such as a bibliographic item, a person, a title, an organization, etc. in a standardized way" is appropriate for that purpose. The element is used:

- directly after the root element to record the ID of the transcript itself (which should be sufficient to retrieve any superordinate external documents such as the documentation of the corresponding interaction)
- inside `<person>` elements in the header (see above) to retrieve the documentation of the corresponding speaker

Structure of Transcription Text

The use of an `<annotationBlock>` element is optional according to the ISO specification – it can be omitted if there are no annotations for a given `<u>`, and speaker assignment, time alignment is then done via attributes on `<u>`. For the sake of simplifying data modelling, the ZuMult specification makes `<annotationBlock>` mandatory, i.e. it requires all `<u>` elements to be embedded in an `<annotationBlock>`, and it is always the `<annotationBlock>`, rather than the `<u>`, which carries attributes `@who`, `@start` and `@end`.

```
<annotationBlock xml:id="c43" who="CJ" start="TLI_48" end="TLI_49">
    <u xml:id="u_d5e1834">
        <!-- [...] -->
    </u>
</annotationBlock>
```

We also require that the content of `<u>` be always and fully tokenized. This means that no character data (and hence no mixed content) is allowed for `<u>`. The content of `<u>` has to be fully decomposed into `<w>`, `<pause>`, `<incident>`, `<pc>` and `<gap>` elements.

Furthermore, we require a unique `@xml:id` at a minimum for all `<w>` and all `<annotationBlock>` elements in the body. Other elements can also be given an `@xml:id` if a processing method requires this.

Some processing methods can greatly profit from the assumption that transcription data have a fixed hierarchy depth. A fixed hierarchy depth allows us, first, to define an unequivocal document position for each “token”, i.e. leaf element, which is a prerequisite for defining token distances in queries. Second, a fixed hierarchy depth guarantees that the `preceding-sibling` and `following-sibling` axes in XPath exclusively contain nodes of the same type and hierarchy level as the context node. A fixed hierarchy depth requires three additional restrictions to the ISO specification:

(1) The use of `<annotationBlock>` is mandatory (see above).
(2) The use of one and only one level of `<seg>` directly beneath `<u>` is mandatory. For transcription systems which do not have a level of segmentation between the token and the highest structural level (“turn”, “contribution” or similar), this means that a redundant `<seg>` element with an appropriate `@type` attribute must be used. For instance, for transcripts of the FOLK corpus which are transcribed according to the cGAT conventions for minimal transcripts, we introduce a `<seg type="contribution">` element, whereas for HIAT, the `<seg>` division of `<u>` is used to represent the “utterances” (“Äußerungen”) which the transcript system defines.
(3) The element `<seg>` can only contain leaf nodes. Specifically, this means that `<unclear>` and `<choice>` must not be used to indicate uncertainty in transcription or a choice between alternative transcriptions, respectively. We think that these elements are not important enough to sacrifice the fixed hierarchy depth for them. Instead, we propose to use an attribute `@type='uncertain'` on each token that would have been a child of `<unclear>`, and, in case of alternatives, to only keep the first alternative within the `<seg>` element while treating all others as annotations in a `<spanGrp>` with `@type='alternative'`.

Example in ISO specification	Corresponding ZuMult solution
`<!-- uncertain passage -->`	`<annotationBlock who="SPK0" start="T4" end="T6">`

```xml
<u who="#SPK0" start="#T4" end="#T6">
    <w>you</w>
    <unclear reason="background noise">
        <w>should</w>
    </unclear>
    <w>let</w>
    <!-- [...] -->
</u>
```

```xml
    <u>
        <seg type="contribution">
            <w>you</w>
            <w type="uncertain">should</w>
            <w>let</w>
            <!-- [...] -->
        </seg>
    </u>
</annotationBlock>
```

```xml
<!-- uncertain passage with alternatives for a single word-->
<u who="#SPK0" start="#T4" end="#T6">
    <w>you</w>
    <unclear>
        <choice>
            <w>should</w>
            <w>could</w>
        </choice>
    </unclear>
    <w>let</w>
    <!-- [...] -->
</u>
```

```xml
<annotationBlock who="SPK0" start="T4" end="T6">
    <u>
        <seg type="contribution">
            <w xml:id="w1">you</w>
            <w xml:id="w2" type="uncertain">should</w>
            <w xml:id="w3">let</w>
            <!-- [...] -->
        </seg>
    </u>
    <spanGrp type="alternative">
        <span from="w2" to="w2">could</span>
    </spanGrp>
</annotationBlock>
```

```xml
<!-- uncertain passage with alternatives for a sequence of words-->
<u who="#SPK0" start="#T4" end="#T6">
    <w>I</w>
    <w>kiss</w>
    <unclear>
        <choice>
            <seg>
                <w>the</w>
                <w>sky</w>
            </seg>
            <seg>
                <w>this</w>
                <w>guy</w>
            </seg>
        </choice>
    </unclear>
    <!-- [...] -->
</u>
```

```xml
<annotationBlock who="SPK0" start="T4" end="T6">
    <u>
        <seg type="contribution">
            <w xml:id="w1">I</w>
            <w xml:id="w2">kiss</w>
            <w xml:id="w3" type="uncertain">the</w>
            <w xml:id="w4" type="uncertain">sky</w>
        </seg>
    </u>
    <spanGrp type="alternative">
        <span from="w3" to="w4">
            <span>this</span>
            <span>guy</span>
        </span>
    </spanGrp>
</annotationBlock>
```

Note that there is one case where a fixed hierarchy depth cannot be enforced, namely when an **<anchor>** is needed inside a **<w>**, which typically happens when overlapping speech sets in inside a word of another speaker. In these cases, **<w>** will have mixed content, and the DOM tree one additional level of hierarchy.

```xml
<annotationBlock who="EP" start="TLI_1214" end="TLI_1217" xml:id="ab788">
    <u xml:id="c788">
       <seg type="contribution">
           <w xml:id="w3156">äh</w>
           <w xml:id="w3157">sissi</w>
           <pause type="micro"/>
           <w xml:id="w3158">du</w>
           <w xml:id="w3159">siehst</w>
           <w xml:id="w3160">fantas<anchor synch="TLI_1216"/>tisch</w>
           <w xml:id="w3161">aus</w>
    </seg>
    </u>
</annotationBlock>
```

Altogether, there is only a small set of elements which can occur as leafs of the document tree. These are:

- <w>

- <pc>
- <pause>
- <incident>
- <gap> (not used in our data)
- <anchor>

Solutions for specific annotation types

The ZuMult specification requires that normalized forms, lemmas and Part-Of-Speech tags – if they exist – be encoded as attributes `@norm`, `@lemma` and `@pos` on the respective `<w>` element, rather than as spans in a `<spanGrp>` of their own. This is mainly required for performance reasons: this type of annotation is almost ubiquitous in the corpora we work with, and it is much more efficient to retrieve a property of a token via its attributes than via a standoff annotation in a `<spanGrp>`. Still, switching between these two types of representation is relatively straightforward, and the schema does not explicitly preclude the option of representing the same information in `<spanGrp>`.

```
<annotationBlock who="CB" start="T0" end="T2" xml:id="ab1">
    <u xml:id="u1">
        <seg type="contribution">
            <w xml:id="w1" norm="sure">sure</w>
            <w xml:id="w2" norm="enough">nuff</w>
            <w xml:id="w3" norm="and">an</w>
            <w xml:id="w4" norm="yes">yes</w>
            <w xml:id="w5" norm="I">I</w>
            <w xml:id="w6" norm="do">do</w>
        </seg>
    </u>
</annotationBlock>
```

Most other annotations will be represented inside `<spanGrp>` elements. So far, we have identified the following annotations to be relevant for the data in ZuMult:

(a) Annotation of token spans

One of the most frequent annotation patterns is the assignment of a category to a contiguous stretch of tokens within a `<u>`. For instance, parts of the GeWiss data are annotated for "Diskurskommentierungen" (discourse commenting) in this way. The annotations are represented in `<span>` elements inside a `<spanGrp>` with an appropriate `@type` attribute. Attributes `@from` and `@to` point to the IDs of the first and last `<w>` element to which the annotation applies.

```
<annotationBlock who="LKC_0243" start="TLI_7" end="TLI_52" xml:id="ab3">
    <u xml:id="c3">
        <anchor synch="TLI_7"/>
        <w xml:id="w11">meine</w>
        <w xml:id="w12">sehr</w>
        <w xml:id="w13">verehrten</w>
        <w xml:id="w14">damen</w>
        <w xml:id="w15">und</w>
        <w xml:id="w16">herren</w>
        <anchor synch="TLI_8"/>
        <pause dur="PT0.43S"/>
        <anchor synch="TLI_9"/>
        <w xml:id="w17">ich</w>
        <w xml:id="w18">danke</w>
        <anchor synch="TLI_10"/>
        <pause dur="PT0.42S"/>
        <anchor synch="TLI_11"/>
        <w xml:id="w19">für</w>
        <w xml:id="w20">die</w>
        <w xml:id="w21">freundliche</w>
        <w xml:id="w22">einführung</w>
        <anchor synch="TLI_12"/>
        <pause dur="PT0.7S"/>
```

```
        <anchor synch="TLI_13"/>
        <w xml:id="w23">und</w>
        <pause type="micro"/>
        <w xml:id="w24">möchte</w>
        <w xml:id="w25">gleich</w>
        <w xml:id="w26">vorausschicken</w>
        <anchor synch="TLI_14"/>
        <pause type="micro"/>
        <w xml:id="w27">dass</w>
        <w xml:id="w28">ich</w>
        <w xml:id="w29">ihnen</w>
        <w xml:id="w30">heute</w>
        <w xml:id="w31">sehr</w>
        <w xml:id="w32">sehr</w>
        <w xml:id="w33">viel</w>
        <w xml:id="w34">zumute</w>
    </u>
    <!-- spanGrps for normalisation, lemma and pos as usual -->
    <spanGrp type="DK">
        <!-- DK stands for 'Diskurskommentierung' -->
        <!-- the GEWISS data are annotated in part for this -->
        <span from="w17" to="w22">D2_Anfang</span>
        <span from="w23" to="w26">D1_Sprechhandlung-A</span>
    </spanGrp>
</annotationBlock>
```

(b) Annotation of non-contiguous tokens

If an annotation refers to more than one token, and there are intervening tokens to which the annotation does *not* refer, `@from` and `@to` cannot be used in the way described above. Instead, we allow an attribute `@corresp` ("points to elements that correspond to the current element in some way" according to the TEI guidelines) with the referenced IDs separated by a space. For example, a lemmatisation of separable particle verbs (Batinic & Schmidt 2018) can be expressed as follows.

```
<annotationBlock who="EP" start="TLI_1214" end="TLI_1217" xml:id="ab788">
    <u xml:id="c788">
        <w xml:id="w3156">äh</w>
        <w xml:id="w3157">sissi</w>
        <pause type="micro"/>
        <w xml:id="w3158">du</w>
        <w xml:id="w3159">siehst</w>
        <w xml:id="w3160">fantas<anchor synch="TLI_1216"/>tisch</w>
        <w xml:id="w3161">aus</w>
    </u>
    <spanGrp type="super-lemma">
        <span corresp="w3159 w3161">aussehen</span>
    </spanGrp>
</annotationBlock>
```

(c) Language of transcript, utterances, tokens etc.

We require the `<text>` element which contains all transcription text to have an `@xml:lang` attribute specifying "the" language of the transcript. Following the recommendations at https://tools.ietf.org/html/bcp47, the value of that attribute will usually be a (lower-case) two letter code (ISO639-1) if it exists, and a three letter code (ISO639-2) otherwise. Further mechanisms for extending the scheme (e.g. region subtags or script subtags) can be applied if required.

```
<text xml:lang="de">
```

Whenever a transcript contains speech tokens in more than one language, `@xml:lang` attributes on elements on lower levels of the document hierarchy can be used to annotate, for instance, code switches, code mixes, loan words or similar phenomena. The following example is from the Australian German corpus.

```
<text xml:lang="de">
    <!-- [...] -->
    <annotationBlock who="EK" start="TLI_96" end="TLI_100" xml:id="ab54">
        <u xml:id="c54">
            <w xml:id="w1163">Äh</w>
            <pc xml:id="pc12">,</pc>
            <w xml:id="w1164">meine</w>
            <w xml:id="w1165">äh</w>
            <w xml:id="w1166">Großeltern</w>
            <pc xml:id="pc13">,</pc>
            <w xml:id="w1167">mein</w>
            <w xml:id="w1168">Großvater</w>
            <w xml:id="w1169">ist</w>
            <w xml:id="w1170">ausgewandert</w>
            <!-- [...] -->
            <w xml:id="w1184">wir</w>
            <w xml:id="w1185">haben</w>
            <w xml:id="w1186">keine</w>
            <w xml:id="w1187">äh</w>
            <pc xml:id="pc16">,</pc>
            <w xml:id="w1188" xml:lang="en">what</w>
            <w xml:id="w1189" xml:lang="en">would</w>
            <w xml:id="w1190" xml:lang="en">you</w>
            <w xml:id="w1191" xml:lang="en">say</w>
            <w xml:id="w1192" xml:lang="en">there</w>
            <pc xml:id="pc17">?</pc>
            <w xml:id="w1193">Keine</w>
            <pc xml:id="pc18">-</pc>
            <w xml:id="w1194">es</w>
            <w xml:id="w1195">ist</w>
            <w xml:id="w1196">nirgends</w>
            <w xml:id="w1197">könn'</w>
            <w xml:id="w1198">Se</w>
            <w xml:id="w1199">finden</w>
            <w xml:id="w1200">wo</w>
            <w xml:id="w1201">meine</w>
            <w xml:id="w1202">Eltern</w>
            <pc xml:id="pc19">.</pc>
        </u>
    </annotationBlock>
</text>
```

The general assumption is that `@xml:lang` specifications on lower levels will override `@xml:lang` specifications on higher levels, or, conversely, when an element on a lower level does not have an `@xml:lang` attribute, it "inherits" the language of the hierarchically nearest element which does have one.

A language annotation via `@xml:lang` can also be applied to `<spanGrp>` elements, most importantly when the annotations are translations (of an utterance or parts thereof).

```
<text xml:lang="de">
    <!-- [...] -->
    <annotationBlock who="SPK1" start="T11" end="T18" xml:id="ab5">
        <u xml:id="u5">
            <seg xml:id="seg5" type="utterance" subtype="declarative">
                <w xml:id="w47">Ach</w>
                <anchor synch="T12"/>
                <w xml:id="w48">so</w>
            </seg>
            <anchor synch="T13"/>
            <seg xml:id="seg6" type="utterance" subtype="declarative">
                <w xml:id="w49">Mhm</w>
                <pc xml:id="pc4">,</pc>
```

```
                <anchor synch="T14"/>
                <pause dur="PT1.1S"/>
                <anchor synch="T16"/>
                <w xml:id="w50">also</w>
                <w xml:id="w51">kein<anchor synch="T17"/>en</w>
                <w xml:id="w52">Führerschein</w>
            </seg>
        </u>
        <spanGrp type="translation" xml:lang="en">
            <span from="T11" to="T13">Oh, okay. </span>
            <span from="T13" to="T18">Mhm, no driver's license then. </span>
        </spanGrp>
    </annotationBlock>
</text>
```

(d) References to external items

A frequent special case is an annotation of tokens, utterances, sequences of utterances, annotation blocks for references to external items, mostly in elicitation tasks such as:

- Prompts in a translation task (Wenker sentences and similar)
- Pictures in a picture naming task
- Objects of a map in a map task

In this case @target can be used on an appropriate to “specif[y] the destination of the reference by supplying one or more URI References” (definition from the TEI guidelines). More specifically, the target can be addressed as some element in an external XML document.

```
<annotationBlock who="SPK1" start="T0" end="T1" xml:id="au2">
    <u xml:id="u2" xml:lang="pt">
        <seg xml:id="seg1" type="utterance" subtype="not_classified">
            <w xml:id="w1">Se</w>
            <w xml:id="w2">ele</w>
            <w xml:id="w3">puder</w>
            <w xml:id="w4">resolver</w>
            <w xml:id="w5">esse</w>
            <w xml:id="w6">problema</w>
            <pc xml:id="pc1">,</pc>
            <w xml:id="w7">ele</w>
            <w xml:id="w8">é</w>
            <w xml:id="w9">muito</w>
            <w xml:id="w10">inteligente</w>
        </seg>
    </u>
    <spanGrp type="kaufmann-sentences">
        <!-- this is saying that #seg1 is a prompt for Kaufmann's sentence 16 -->
        <span from="seg1" to="seg1" target="kaufmann-sentences.xml#s16">prompt</span>
    </spanGrp>
</annotationBlock>
<annotationBlock who="SPK2" start="T2" end="T3" xml:id="au4">
    <u xml:id="u4" xml:lang="pdt">
        <seg xml:id="seg3" type="utterance" subtype="not_classified">
            <w xml:id="w13">Wann</w>
            <w xml:id="w14">hei</w>
            <w xml:id="w15">kann</w>
            <w xml:id="w16">det</w>
            <w xml:id="w17">Problem</w>
            <w xml:id="w18">löse</w>
            <w xml:id="w19">dann</w>
            <w xml:id="w20">is</w>
            <w xml:id="w21">her</w>
            <w xml:id="w22">en</w>
            <w xml:id="w23">sehr</w>
            <pause dur="PT0.7S"/>
            <w xml:id="w24">äh</w>
            <w xml:id="w25">intelligenter</w>
            <w xml:id="w26">Mensch</w>
        </seg>
    </u>
```

```
    <spanGrp type="prompt-reference">
        <!-- this is saying that #seg3 is a Plautdietsch translation of Kaufmann's sentence 16 -->
        <span from="seg3" to="seg3" target="kaufmann-sentences.xml#s16">translation</span>
    </spanGrp>
</annotationBlock>
```

No specific assumption is made about the target document except that it should be well-formed XML and that the address given in target can be identified via an @xml:id attribute. For the above example, the target document ("kaufmann-sentences.xml") might look as follows.

```
<prompt-list>
    <!-- [...] -->
    <prompt xml:id="s16">
        <s xml:lang="pt">Se ele puder resolver esse problema, ele é muito inteligente.</s>
        <s xml:lang="en">If he can solve this problem, he's a genius.</s>
        <s xml:lang="es">Si él puede resolver este problema, es muy inteligente.</s>
    </prompt>
    <prompt xml:id="s17">
        <s xml:lang="pt">Se ele realmente matou o homem, ninguém pode ajudar ele.</s>
        <s xml:lang="en">If he really killed the man, nobody can help him.</s>
        <s xml:lang="es">Si realmente mató al hombre, nadie lo puede ayudar.</s>
    </prompt>
    <!-- [...] -->
</prompt-list>
```

Miscellaneous

According to the TEI guideline, `@interval` on a `<when>` "specifies a time interval either as a number or as one of the keywords defined by the datatype" and `@since` "identifies the reference point for determining the time of the current when element, which is obtained by adding the interval to the time of the reference point". This leaves somewhat unclear how to proceed if the main function of `@interval` is to provide an offset into a recording, as is the case for transcriptions of spoken language. In the ISO specification, the very first `<when>` element is left without `@interval` and `@since` attributes. Instead, an `@absolute` attribute is proposed for the case where the exact date and time of the start of the recording are known, and it is implicitly assumed that this `<when>` element corresponds to an interval of 0.0 seconds since the start of the recording. For reasons of consistency and better processability, we prefer to make this assumption explicit by always specifying `@interval='0.0'` for the first element, and to let its `@since` attribute point to itself. We disallow `@absolute` on `<when>` because it serves no real purpose, assuming that the date and time of the recording can be documented in the metadata.

```
<timeline unit="s">
    <when xml:id="TLI_0" interval="0.0" since="TLI_0"/>
    <when xml:id="TLI_1" interval="2.354" since="TLI_0"/>
    <when xml:id="TLI_2" interval="2.633" since="TLI_0"/>
    <when xml:id="TLI_3" interval="2.992" since="TLI_0"/>
</timeline>
```

@rend Attributes

Practical Information for Transcribers

FOLKER/OrthoNormal

EXMARaLDA

Other tools

- Transcriber
- ELAN
- Praat
- CLAN/CHAT

TEI Drop

CLARIN Webservices

Practical Information for Developers

Example data

Schema

XSL Stylesheets

- For XML to XML transformation
- For XML to HTML transformation

Code