Gesprächsforschung - Online-Zeitschrift zur verbalen Interaktion (ISSN 1617-1837)
Ausgabe 18 (2017), Seite 274-296 (www.gespraechsforschung-ozs.de)

# Conversation Analysis and the XML method

**Christoph Rühlemann / Matt Gee**

**Abstract**
In this paper we introduce the XML method, a trio of technologies that can benefit conversation-analytic research. Specifically, we make a case for converting the center piece of CA research, the Jeffersonian transcript, into the format of the eXtensible Mark-up Language (XML). XML essentially turns documents into hierarchically ordered networks of nodes. As a network, an XML document can be exhaustively searched and any node or node set it contains can be extracted. We argue that the main benefit of formatting CA transcriptions in XML lies in the quantifiability that the format facilitates: CA-as-XML can provide precise "numbers and statistics" (Robinson 2007:65) thus helping to efficiently quantify observations and statistically substantiate claims about the 'generalizability' of observed practices of social action. We also introduce XPath and XQuery, two related query languages designed to exploit the XML format. Further, we describe XTranscript, a free online tool developed to convert completed CA transcripts to XML. Central to our approach is that the methodology be accessible to linguistics of varying levels of technical experience. Therefore, we also describe how this, and common concerns relating to the treatment of spoken data, have shaped our work in this area thus far.

*Keywords*: XML, Quantification, XTranscript, XPath/XQuery.

## 1. Introduction

CA is typically defined as a qualitative method (e.g., Stivers/Sidnell 2013:2). Recently, however, the reliance on the qualitative method alone, which, by implication, amounts to a rejection of quantitative methods for CA, has been questioned. First, a number of CA studies already include a quantitative component, as illustrated in the large number of studies using a mixed methods approach as cited by Stivers (2015:1-2). Second, and more importantly, Stivers (2015) notes that "for all its qualitative refinement, CA is arguably the most quantitative of the qualitative social science methods" (Stivers 2015:3). The key reason lies in Sacks's foundational assumption that "there is order at all points" (Sacks 1984:22). 'Order', in Sacks's sense, represents a "resource of a culture" (Sacks 1984:22). As such it is opposed to practices of action that are idiosyncratic, that is, of actions or action patterns merely typical of individuals or groups. In other words, Sacks's notion of order relates to *social practices* of action. These are the components of the 'machinery' which CA aims to find (Sacks 1984:26). They involve "communication rules that generate *regular patterns* of understanding and interactional organization" (Robinson 2007:65; emphasis in original). To be able to discern regular patterns, distributional evidence is indispensable: only if some behavior is more recurrent than some other comparable behavior, can it be deemed a candidate for a regular behavior, and hence a candidate for a social practice of action. Recurrence, in turn, involves quantification of some sort. Intriguingly, quantification has long been an inherent component of any CA research into practices in talk-in-interaction. As

Schegloff observes, quantification is represented in "the common use in some conversation-analytic writing of terms such as *massively*, *overwhelmingly*, *regularly*, *ordinarily*, and (as in the current sentence) *commonly* (Schegloff 1993:99; cf. also Stivers 2015; De Ruiter/Albert 2017). Quantification is, then, often expressed in *scalar* terms in CA research. Scalar expressions are inherently vague (de Ruiter/Albert 2017) leaving much important information unsaid. This missing information includes not only the exact extent of the observed distribution but also the crucial question, raised by Sacks (1984:23) in the context of speaking of 'order at all points', of "generalizability", i.e., the question whether the distribution observed in the small sample available to the researcher is the same as the distribution in the population of the phenomenon under investigation. This reliance on distributional evidence manifested in the use of scalar descriptors clearly belies "[t]he idea that any form of quantification sits in direct opposition to CA methods" (Stivers 2015:16) and reveals that reducing CA to an exclusively qualitative method "is a very restrictive view of CA" (Stivers 2015:16). Obviously, rather than being scalar and vague, quantification can also be more precise, "implicat[ing] numbers and statistics" (Robinson 2007:65).

In this paper, we wish to introduce the XML method, a trio of technologies that can benefit conversation-analytic research. The trio includes the (i) eXtensible Mark-up Language XML, (ii) XPath and XQuery, two programming languages to query XML databases, and (iii) XTranscript, an online tool we developed to automatically convert CA transcripts into the XML format. Specifically, we make the case that the XML method has the potential to advance the use of exact quantification in CA research. Quantification relies on coding, i.e., marking talk-in-interaction for some relevant analytical categories. As Stivers warns, given that "[c]oding, in any form, necessarily involves reduction from the intricate complexities of human behavior to broad and flattened categories" (Stivers 2015:2), "quantifying CA practices is not always appropriate, nor is it always analytically productive" (Stivers 2015:15). The coding we propose connects to the center piece of any CA research: the Jeffersonian transcript, which is to be "detailed enough to facilitate the analyst's quest to discover and describe orderly practices of social action in interaction" (Hepburn/Bolden 2013:58). CA transcripts are particularly suited to XML coding for two reasons: (i) they already contain a wealth of codings and (ii) the codings are very largely standardized and consensual: the categories and conventions for transcription developed by Jefferson (e.g. Jefferson [2004]) are universally adhered to in CA research. As is well known, CA transcripts feature codings for sequential aspects including overlap and latching, codings for temporal aspects such as inter- and intra-speaker pauses, speed-up and slow-down, codings for phonological aspects such as changes in pitch, intonation, intensity, as well as stretching, truncation, aspiration, stress, and smile voice, codings for the transcriber's comments, and, potentially, codings for gaze behaviour (e.g., Goodwin 1984). The XML format is perfectly applicable to any of these codings. Moreover, XML is equally applicable to coding for more specialized research foci. For example, in the case of studying resources deployed to mobilize response (cf. Stivers/Rossano 2010), XML mark-up could also be used to capture candidate features such as interrogative morpho-syntax and intonation, speaker gaze, recipient epistemic bias and so forth.

In the following we introduce the trio of technologies that constitute the XML method. We start off with characterizing XML and sketching out the advantages it

offers for CA research. We also introduce and illustrate how XPath and XQuery can be used to exploit XML documents. Further we describe XTranscript, a tool developed in the Research and Development Unit for English Studies (RDUES) at Birmingham City University to convert CA transcripts into XML.

## 2. What is XML?

In this brief introduction to XML we aim to provide merely a sketch of those properties of XML that are directly relevant to the issue of transforming CA transcripts to XML. For an accessible introduction for linguists see Hardie (2014); for a general introduction, see, for example, Watt (2002). We illustrate XML elements and their components based on a tagging scheme developed recently (cf. also Rühlemann, 2017) which is also underlying the XTranscript tool described further below; the full tagging scheme is given in the Appendix.

Simply speaking, eXtensible Markup Language (XML) is a data architecture connecting meta-data and data. The architecture's defining feature is the hierarchical network of nodes. Every node in the XML structure is connected somehow to any other node; also, being a hierarchy, every node is either subordinate or superordinate to another node, as shown in the tree structure in Figure 1. Further, XML "provides a standard syntax for the mark-up of data and documents" (Watt 2002:1). The syntax along with the hierarchical network structure make XML documents exhaustively searchable and therefore useful for linguistic research.

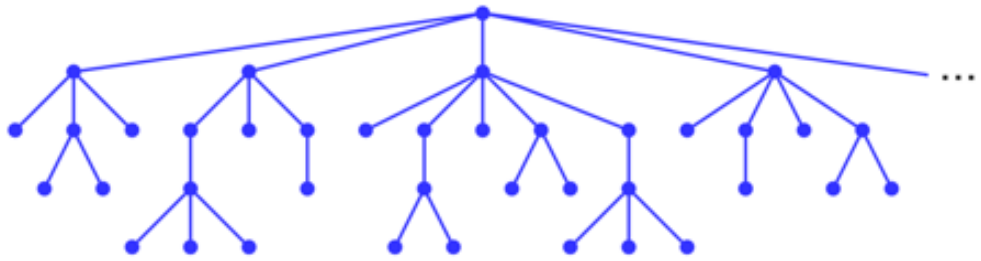

**Figure 1: XML tree structure**

The most frequent type of XML node is the 'element'. Elements have one of two structures, either a pair of 'tags' with content between them, is in example (1), or a single tag, as in (2), which is said to be an empty element. XML tags must start with an opening bracket < and end with a closing bracket >. The position of the forward slash / defines the type of XML tag:

1. No forward slash is used for an opening tag, as in example (1).
2. A forward slash before the name of the element denotes a closing tag, also shown in (1).
3. A forward slash at the end of a tag's contents denotes an empty tag, as in example (2).

Opening and empty tags (but not closing ones) may contain none, one or multiple attributes which are punctuated by an equals sign and quote marks. Attributes provide more information about the element.

**(1)**
<sequence type="overlap"> I don' know </sequence>

**(2)**
<voice intonation="continued" />

In (1), the element name 'sequence' is used to mark up sequential phenomena (cf. Appendix); the 'type' attribute serves to identify the kind of sequential phenomenon observed, while the value "overlap" specifies the sequential phenomenon as 'overlap'. A key requirement in XML is 'well-formedness'. To be well-formed, the element needs to be closed by the closing tag </sequence>. In (2), the 'voice' element is empty. To be well formed the empty element must end with a forward slash.[1]

In both (1) and (2), only one attribute is specified. However, elements can have multiple attributes (or no attributes). For example, in (3), the 'sequence' element contains two attributes and their values: beside the 'type' attribute, there is the 'n' (for 'number') attribute whose value "1" identifies the overlap as the first in the transcript.

**(3)**
<sequence type="overlap" n="1"> I don' know </sequence>

In (4), the 'timing' element identifies a pause, while the value "1.1" on the attribute 'duration' records the length of the pause. Again, the element is 'empty', as indicated by the ending forward slash, meaning that the element does not play host to another element and thus does not require a closing tag.

**(4)**
<timing type="pause" duration="1.1" />

The more common case where an element does contain another element is shown in (5): we now see that the overlap encountered in (3) is spoken with an intonation signalling continuation. However, *I don' know* just represents the transcriber's best guess due to unclear hearing. The possible hearing is wrapped into a 'comment' element, one level lower in the XML hierarchy than the enclosing 'sequence' element, with a 'hearing' attribute and the value "possible". The continued intonation on the candidate text is captured and specified in the 'voice' element (used for vocal properties of delivery); the value "continued" on the 'intonation' attribute, finally, specifies the intonation contour as incomplete. Both the 'comment' and the 'voice' element are hosted by the 'sequence' element.

[1] Another requirement for well-formedness is avoiding XML overlap, as in <a>  </a>  (cf., for example, Carruthers 2008).

**(5)**

```
<sequence type="overlap" n="1" >
        <comment hearing="possible">
                I don' know
        </comment>
        <voice intonation="continued" />
</sequence>
```

While in (5) there are just two hierarchical levels, with the 'sequence' element being 'parent' to the 'comment' and the 'voice' elements (or the two elements being 'children' to the 'sequence') it is not uncommon in XML documents to find a much more complex hierarchy with many more elements being 'descendants' to higher-order 'ancestor' elements. This complexity will inevitably become apparent in trying to capture the rich detail of CA transcripts in XML format. Consider for illustration (6), the first two lines from a CA transcript (see Section 3 for the full transcript), and (7), a possible rendition of the lines in XML:

**(6) ["Drained canal", BNC: KBD 1790-1801]**

| | | |
|---|---|---|
| 1 | Alan: | Well it's, it's (.) luck innit |
| 2 | | [( I don' know), ] |

**(7)**

```
<transcript id="BNC: KBD 1790-1801" >
        <u who="Alan" n="1" >
                Well it's
                <voice intonation="continued" />
                it's
                <timing type="pause" duration="." >
                luck innit
                <sequence type="overlap" n="1" >
                        <comment hearing= "possible" >
                                I don' know
                        </comment>
                        <voice intonation="continued" />
                </sequence>
        </u>
        <!-- u-elements omitted -->
</transcript>
```

In (7), the XML has grown considerably: we find four hierarchical levels, the 'transcript' element, which encloses the whole transcript (including not only the one utterance by Alan but also many others omitted in (7)), being the highest-level element and the 'comment' element as its most remote descendant. In between the two extremes are fitted the 'u' element, typically used to denote turns, as well as the 'voice', 'timing', and 'sequence' elements.

To the untrained eye, the XML transcript may look rather convoluted and it may not be obvious why it should have any advantage over the Jeffersonian transcript. We therefore specify benefits of the XML format in the next section.

## 3. Why is XML useful for CA?

As noted above, XML is a network structure where any node is connected somehow to any other node. Thus, using appropriate XML query tools such as XPath and XQuery any node or set of nodes can be addressed and extracted for further examination and processing. This addressability and extractability offers distinct advantages. We specify these advantages in the following section. We also showcase potential queries in XPath, a system for querying XML documents. Introducing XPath, as well as its 'big' sister XQuery, in good detail is far beyond our present aims; for a gentle introduction to these tools for corpus linguists see Rühlemann et al. (2015), for more comprehensive and technical descriptions see Watt (2002) and Walmsley (2007). XML offers four advantages for CA research.

*Exhaustive retrievability*
First, unlike commonly used formats such as MS Word whose search functionality works in a 'hop-on hop-off' fashion allowing retrieval of single instances at a time only, XML allows exhaustive retrieval of all target instances in one go. For example, if a researcher's focus is on overlap, a simple XPath query will address and retrieve all overlap instances. Using the above-mentioned tagging scheme, the XPath query could be this:

```
//sequence[@type="overlap"]
```

Here, the double slash initiates an iterative process repeated for each and every 'sequence' element while the square brackets specify a restriction to address only those 'sequence' elements that have the value "overlap" on the 'type' attribute.

*Large-scale analysis*
Second, XML is a machine-readable format. As such it does not 'care' as to how much data it is to process. The amount of data can be small or large - very large, indeed. To return to the above example of the XPath query for overlap: the same simple query would reliably retrieve all overlaps either from a single transcript or a corpus consisting of many thousands of transcripts. This is no doubt an eminent advantage. CA has traditionally worked with small amounts of data; not infrequently do researchers analyze just a handful of transcripts, or even less (e.g., Goodwin [1984] examines gaze in a single transcript), subjecting the data to rigorous qualitative analysis. Storing CA data in XML will allow CA researchers to examine 'big data'. As long as the details of the big data are true to CA principles, this new dimension will come at literally no cost, demanding no sacrifices, but offering a genuine gain.

*Unlimited filtering and combining capabilities*
Third, XML is well suited for highly specific research in that it can accommodate multiple restrictions and combinations. For example, the square bracket in the

XPath query above represents a filter: it specifies that not all types of sequential phenomena are to be retrieved but only that type which satisfies a certain condition (namely, in the above case, that the sequential feature is an overlap). In XML, there is, in fact, no limit to the number of such restrictions used in a single query. For example, assuming a researcher has a corpus of transcripts, he/she can extract all overlaps that are first, second, third, and so on in the transcripts, by exploiting the above-mentioned 'n' attribute:

```
//sequence[@type="overlap" and @n="1"]
```

Not only attributes of one and the same element can be used as filters, but filtering can also be done across elements, thereby combining restrictions from two (or more) different elements. For example, if a researcher is interested in retrieving only those overlaps that contain (in XML parlance:that is a 'parent' or 'ancestor' to) laughter, then the query would be this (laughter is tagged <laugh> in the tagging scheme):

```
//sequence[@type="overlap" and descendant::laugh]
```

If this is still not specific enough because the research exclusively focuses on overlaps that contain between-speech (free-standing) laughter, this condition could simply be added to the query using the attribute 'type' and its value "between-speech":

```
//sequence[@type="overlap" and descendant::laugh[@type="between-speech"]]
```

This line of code may already take some getting used to for XPath novices, but in fact, it is a simple line specifying only three restrictions. It is by no means uncommon to find code that is far more complex because a much larger number of filters are applied (in which case it is usual to use not XPath, which expresses the path in a single line of code, but XQuery, which offers a pre-defined structure, the FLWOR structure [cf. Walmsley 2007: Chapter six], to break long code into separate chunks).

*Enhanced quantifiability*

Fourth, XML provides enhanced quantifiability in that nodes can be counted and arithmetic operations can be performed in XPath and XQuery. The frequency counts thus obtained can be further processed using statistical software to produce, for example, visualizations for data inspection and/or perform statistical tests for significance.

A simple XPath function to count objects is count(). The following query outputs a single number for the frequency of between-speech laughter occurring in overlap:

```
count(//sequence[@type="overlap" and descendant::laugh[@type="between-
speech"]])
```

The next code illustrates the use of an arithmetic operator: using the 'div' operator

(for 'division'), the ratio of overlap affecting between-speech laughter out of all instances of overlap could be calculated thus:

count(//sequence[@type="overlap" and descendant::laugh[@type="between-speech"]]) div count(//sequence[@type="overlap"])

Often researchers are interested, not in overall frequencies, but in how data are spread over a unit. Consider, for instance, (8), the full transcript of Barry's story about going fishing.

**(8) ["Drained canal", BNC: KBD 1790-1801]**

| | | |
|---|---|---|
| 1 | Alan: | Well it's, it's (.) luck innit |
| 2 | | [( I don' know),] |
| 3 | Barry: | [ *I* remember ] once go:n' on, |
| 4 | | I got- (0.4) we got up 'bou' three three thirty 5 |
| | | in the morning ( ) |
| 6 | | went out to er (0.9) canal somewhere up |
| 7 | | (1.3) |
| 8 | | Dulga' area past Dulgate |
| 9 | | (1.3) |
| 10 | | we set up and we'd we'd been fishing for about 11 |
| | | two and half hours |
| 12 | | it's aba- about six thirty in the morning |
| 13 | | this old farmer comes up |
| 14 | | says er (1.1) ↑Aye aye lads, |
| 15 | | he said er (0.7) I wou' n' bother it |
| 16 | | they >>drained this area of the canal a few |
| 17 | | months aG(h)O<< Hhh::::, |
| 18 | | [ hh::: GGAeehh::: he ] he he |
| 19 | Alan: | [ huh huh huh huh huh .] |
| 20 | Barry: | S(h)*at* there watching our floats for hours |
| 21 | | *uh*heh heh: I mean *luck*ily you- you know, |
| 22 | | you'd gone on- with a car |
| 23 | | so it's a ma'er o' throw'n ev'ryth'n in th' back |
| 24 | | ['n' j's go:n'] |
| 25 | Alan: | [ ° ye:: ° ] |
| 26 | Barry: | s'm'ere else sort of (ay) |
| 27 | | (1.5) |
| 28 | | could've sat there all bleed'n' day! |
| 29 | | (1.00) |
| 30 | | °°an' not known anythin' about it°°. |
| 31 | | (4.4) |
| 32 | Alan: | aye |

It will be seen that the telling involves a number of pauses (nine, to be exact) and that the pauses differ considerably in length. The following XPath code extracts all durational values from the 'timing' elements that have the attribute 'type' and the value "pause":

//timing[@type="pause"]/@duration

The durations are depicted in a plot in Figure 2:

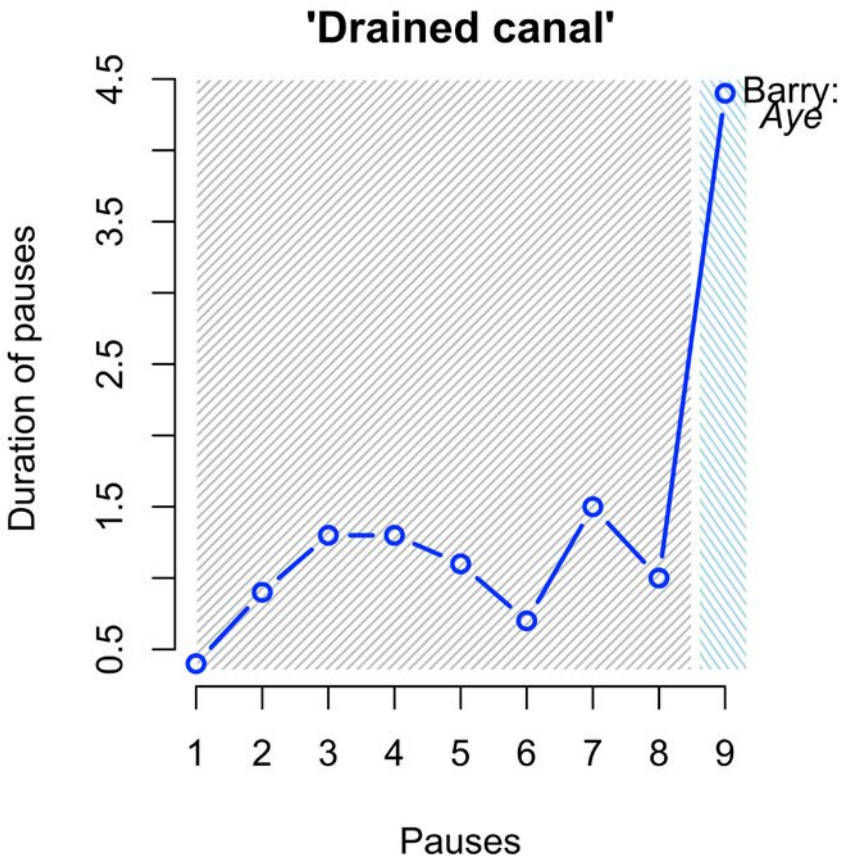


**Figure 2: Durations of pauses in story "Drained canal"**

Certainly the most striking pause is the very last in line 31: its duration is 4.4 seconds, far longer than any other pause in the telling. Looking at its position in the sequential context it becomes clear that this pause is not only an inter-speaker pause (rather than an intra-speaker pause, as all other pauses in the extract). More importantly, its extended length signals story completion - a signal readily taken up in Barry's "aye" in line 32, which can be seen as an instance of Hoey's (2017) sequence-recompletion. In this storytelling interaction, then, pauses do interact with storytelling structure, with the last pause interactively achieving the storytelling's closure.

As shown in the above examples, to be able to reap the benefits of XML, some mastery of XPath (and sometimes also XQuery) is necessary. To get CA transcriptions into the XML format, we developed XTranscript, an online tool that *automatically* converts CA transcripts into the XML format. This tool is introduced in the next section.

## 4. XTranscript: A tool for converting CA transcripts into XML

We have shown the benefits of XML, and how mature and powerful tools from software engineering (i.e. XPath and XQuery) can be leveraged for the analysis of XML encoded texts. However, the issue of creating useful XML files remains. In a research project, it may be possible to design your data collection and annotation

phases to use XML as your primary text format. However, the majority of CA researchers may prefer to work with traditional CA transcripts, or have already collected a considerable amount of CA annotated texts. XTranscript is an attempt to bridge this gap by converting CA transcripts into an XML schema including common CA annotations. In its current form, XTranscript works with the annotations detailed in the Appendix. We hope to add additional flexibility to future versions to enable other transcription conventions to be defined and converted into custom XML schemas.

XTranscript treats a transcript as a hierarchical structure: a text which consists of utterances, which in turn consist of tokens. Different transformation rules are defined at each level to match different types of annotations. For example, at the text level, a rule is defined for matching a whole utterance (a speaker ID followed by text, with an optional line number). From this a <u> XML tag is formed including the speaker ID as an attribute. The content of the utterance is then processed by the next level of rules. An overlap will be identified from it's opening and closing brackets (i.e. [ and ]) and can thus be recorded in the XML as <sequence type="overlap"> … </sequence>. Annotations like this, with opening and closing elements, are also added to a syntax checking system. Thus, if such an annotation is not closed within the utterance (or within another annotation) it can be closed automatically. The text within the utterance is also tokenised, and the final level of transformation rules are applied to each token in turn. This level includes word-specific annotations such as elongated sounds and in-word laughter. The tokenisation also ensures that the resulting XML tags are ordered correctly, thus preventing issues with overlapping elements (which are invalid in XML) and keeping the XML sufficiently clean for further processing.

Further to the CA annotations, XTranscript includes the option of part-of-speech (grammatical word class) annotation using the Stanford CoreNLP tagger (Manning et al. 2014). If part-of-speech tagging is enabled, each token is surrounded by a 'w' tag (an abbreviation for 'word') with attributes for the part-of-speech label (e.g. <w pos="VBD"> for a past tense verb) and the lemma of the lexeme to which the word belongs. The part-of-speech labels used by the Stanford tagger are defined by the Penn TreeBank tag-set (Santorini 1990:6).

In the final stage of processing, XTranscript performs a 'well-formed' check on the generated XML. XML must be well-formed to enable XPath and XQuery searches. If the check fails, the XML can still be edited to resolve the issue.

XTranscript can be accessed at http://rdues.bcu.ac.uk/xtranscript. It is an online service allowing users to upload their texts for conversion. Users can upload either a single file (in plain text, Microsoft Word, Open Document or PDF format) to receive the XML version, or upload a Zip file of documents to be converted, receiving a new Zip file containing the converted texts. The XTranscript website includes a description of the CA-as-XML schema and documentation to help with using XPath, XQuery and suitable search software.

XTranscript's usefulness for CA research has recently been demonstrated in a single-case study on gaze behavior in a multi-party conversation (Rühlemann et al., submitted). The study took advantage of the fact that XTranscript is able to process, not only annotations made to the spoken data, but also multi-modal information in the form of gaze annotation (cf. Goodwin 1984). All gaze changes by participants had previously been painstakingly measured and transcribed in the CA transcript

and were then reliably converted by XTranscript into XML. Thus gaze changes could be quantified, which in turn revealed a correlation between accelerating gaze changes and story progression (amongst other factors). What the case study demonstrates *methodologically* is that XML is capable of efficiently handling lots of quantitative data and that such an analysis can usefully complement the qualitative work typical of CA.

## 5. But really, why XML?

A significant amount of work in digitizing and analyzing transcripts has been performed in linguistics, especially within the sub-field of Corpus Linguistics. Spoken corpora have formed part of seminal projects such as Collins/Birmingham University International Language Database (COBUILD), the British National Corpus (BNC), International Corpus of English (ICE) and the TalkBank corpora. In each case consideration must be given to the format by which to encode the spoken data. The same is true in regards to the quantitative analysis of CA transcripts. Three volumes stand out in which such issues are discussed: *Spoken English on Computer* (1995), *Developing Linguistic Corpora: a Guide to Good Practice* (2005) and *Compilation, transcription, markup and annotation of spoken corpora* (2016). This section highlights some of the key issues that we believe to be relevant to the XML method.

### 5.1. Formats other than XML

CA transcripts encode a lot of detailed information, but are ultimately designed to be read by humans. In order to make the leap to quantitative analysis, the transcripts must be in a form which is machine-readable. This goes beyond simply making them digital. After all, a PDF is digital, but the computer cannot understand the information within beyond drawing the letters and images on screen. To be machine-readable the format of the transcripts must follow predefined rules and software must exist which can interpret these rules. XML is such a case. Even more to our benefit is that XML is now ubiquitous in the digital world and so is software which can interpret it.

In CA, conventions have been developed for the annotation of spoken data. Studies such as Sacks, Schegloff and Jefferson (1974) and Jefferson (2004) have helped to embed these in the research area in what has come to be known as the 'Jeffersonian' system of annotation. The consistency that this has provided gives us a good starting point from which we can move towards machine-readable transcripts. To enable the quantitative analysis of CA transcripts, two possible approaches present themselves: 1. develop custom software to enable the analysis or 2. convert the transcripts into a format for which analysis software already exists.

We are not aware of software which works directly with Jeffersonian transcripts in a quantitative manner. However, other corpus building projects have developed custom software alongside a custom format, for example the TalkBank corpora (MacWhinney 2000). The TalkBank projects have successfully compiled a number of spoken corpora, sharing them online for others to reuse. The transcripts in the TalkBank projects conform to the CHAT format, developed for use with the CLAN

software. CHAT encodes many different elements from various sub-fields of linguistics into a common machine-readable format. This includes annotations for phonology, morphology, grammar, speech acts and CA. The CA annotations use a scheme which has some similarity to the Jeffersonian system, however the majority of symbols are different in order to avoid conflicts with the rest of CHAT's syntax. The CHAT system, then, including its CA transcriptions, is tied into the CLAN software for analysis. CLAN provides a number of search and summarization tools for use with transcripts. These include word frequency and co-occurrence patterning, calculating the length of utterances and calculating the length of pauses and overlaps. These functions don't make use of the CA annotations, but rather the utterance and time markers which also make up part of the CHAT format. When it comes to the CA annotations, the focus is still on the qualitative analysis. CLAN provides useful functions for displaying the annotations in clear and readable manner and aligning the transcript with the audio, but the quantitative functions in CLAN have been designed to work with the annotations it provides outside of the CA system.

CLAN is but one example, but highlights our desire to avoid being tied to a single software package. XML provides more flexibility. Multiple packages exist which can read XML and perform XPath queries, including desktop applications, command line programs and software libraries. We describe above how useful the combination of XML and XPath can be for extracting information from complex annotations and networks of nodes. We therefore choose the second option from before, to convert the CA transcripts into XML, and leverage the tools that already exist to process and query XML.

## 5.2. Which type of XML?

Until this point we have talked of the 'format' of XML and eluded to the schema we designed for the purpose of studying CA phenomena. XML, however, is not one format. Rather it is the syntax or punctuation of a machine-readable format which allows many custom versions (i.e. schemes) of XML to be developed. XML defines (amongst other things) that elements consist of tags and attributes, that tags start with < and end with > and that an attribute is written in the form *key="value"*. The names of the elements and attributes is left up to the designers of a XML schema. How precisely the XML schema needs to be defined is also left up to its creators. Indeed, XML allows a high degree of flexibility in the naming of elements and attributes. Thus, many different types of XML are in use and continue to be created by developers and researchers.

Tools exist to support the definition of XML schemas, notably Document Type Definitions (DTDs) and the more recent XML Schema. These can be understood as formal definitions of a type of XML. They dictate, for example, which elements can be contained within each other, which attributes can be used in conjunction with which elements and the data types that attribute values may take. The benefit of these tools is that the XML can be tested to see if the rules have been followed (a process called 'validation'). XML then can be checked for two levels of correctness: first, whether it is *well-formed* (for which the syntax must be correct) and second, whether it is *valid* (for which the rules defined in the schema must be followed). Ensuring XML is valid can be very useful as it helps eliminate mistakes; however,

writing XML Schema and performing validation requires an extra level of technical knowledge beyond the foundation we provide above. For our use case, validation is not essential and it is sufficient that the XML be well-formed in order to use XPath and XQuery. In linguistics, we see both ends of the spectrum. Major projects such as the Text Encoding Initiative have developed strictly defined rules for the encoding of a common XML format. Whereas a more recent trend has emerged for smaller research projects to use a lighter touch to XML, defining their own schemas in a more fluid manner.

The Text Encoding Initiative (TEI) (2018) is driven by a consortium of researchers and commercial partners from many fields working towards a common XML format for the sharing of texts. TEI is perhaps most notable as being the format in which the British National Corpus XML Edition (2007) (henceforth, BNC) is distributed. Thompson (2005) recommends TEI as a good format for use with spoken data in corpus linguistics, especially in cases where standardization, interchangeability and data sharing are important. To this end, the consortium provides DTD and XML Schema definitions of TEI, the latest version of which is TEI-P5, as well as tools by which the schema can be extended, and substantial written guidelines are provided as a more humanly accessible description of the TEI standard (TEI Consortium 2018).

TEI attempts to encode many different text-types, including both written and spoken modes, in XML. An entire section is dedicated to the encoding of *Transcriptions of Speech* in the latest version of the guidelines (Section 8). Many elements that are of relevance to CA are included in the schema. This includes *u* elements, which bound utterances and marks who the speaker is and information about transitions between utterances (e.g. latching, overlap, and pauses). It includes elements for in turn phenomenon, such as *pause, unclear, shift* (for changes in voice quality, e.g. tempo, pitch or volume) and *anchor* (for recording overlaps). Much of the components exist in TEI then that would enable CA transcripts to be encoded in XML. However, we observe some issues with the TEI schema which makes it unsuitable for our purposes.

In opposition to strictly defined XML schema, Hardie (2014) outlines an approach to using XML which researchers can take whilst only needing to understand the fundamentals of XML. As Hardie notes, the nature of corpus building has shifted away from large-scale projects with requirements to support a great number of researchers or a myriad of research questions. Instead it has moved towards smaller-scale studies in which individuals or small research teams collect data and build corpora for specific studies. The later approach is reminiscent of much practice in CA research. Hardie goes on to outline the minimal requirements for a researcher to understand and write XML, eschewing the technically complex elements mentioned above (such as XML Schema) and presenting something much more approachable for those with less technical knowledge. We find ourselves in agreement with Hardie's recommendations and encourage researchers unsure about XML to read the gentle introduction which the paper provides

Hardie (2014:102) makes several suggestions in relation to XML for linguistic purposes, including: 1) regions in a text (e.g. overlaps) should be marked by opening and closing tags, 2) points in a text (e.g. pauses) should be marked by empty tags, and 3) annotations or metadata should not be recorded as text, but rather as attributes. The latter point Hardie makes less forcibly, but we explain below how it

becomes important in the context of quantitative analysis. Together, these three suggestions combine to enable XPath queries such as accurately counting the number of words a speaker uses or retrieving the content of overlaps. The TEI schema, however, is defined such that many spoken features are marked as points by empty tags, rather than regions (this includes overlaps and changes in voice quality) making it a complex task to extract their content without the development of custom software. TEI also defines that descriptions (e.g. "reads aloud from newspaper" - part of a non-verbal incident) be encoded as text, and thus would be extracted as though part of a speaker's turn when, for example, extracting the text of the utterance in which this occurs or counting the number of words each speaker utters. XPath/XQuery expressions can be devised to remove this content, but the code required to do so quickly becomes complex. These are very fine-grained issues, but we see them as deal breakers given our goal of quantitative analysis of the transcripts and our desire to keep the XPath expressions clean and easy-to-use.

Another issue presents itself when dealing with strictly defined XML schemas: that of expanding the XML. One of the great advantages of XML is that new elements and attributes can be added to enable multiple levels of analysis. If we provided a strict definition of our XML using DTD or XML Schema, we would need to update these definitions in order to include our new features and for the XML to be valid. There are two ways of doing this, either revise the original XML Schema definition to include the new elements, or add namespaces to the XML documents (each element and attribute can be associated with a namespace and each namespace can have a separate schema definition). We are once again adding an extra level of complexity, which while maybe desirable in large projects where strict conformity is a must, may well be a significant barrier to the uptake of XML in a linguistic research context. Thus, we stipulate that the XML must be well-formed, as is required for its use with XPath, but do not enforce any further validation on the XML.

The XML schema produced by XTranscript and used within this paper is designed to fulfil the goal of presenting CA annotations for quantitative analysis. Ultimately, we are not attempting to define a XML schema which may become a standard for others to follow, but rather we are using XML and XPath as means to an end. In this context, we believe a fundamental understanding of XML sufficient.

### 5.3. Which features of spoken data to include?

The amount of detail that can be included within a transcript can vary greatly and this is typically led by the type of analysis required. A study in phonology must include representation of phonemes, whereas a study in lexicography may need no more than the plain text, without even reference to whom each turn belongs.

In our case, that of taking the CA (Jeffersonian) transcript as our starting point, these decisions have essentially already been made. We make an assumption that CA researchers will find sequences, temporal elements and properties of voice of interest as these are most likely to be encoded within the transcripts with which we are working. That is not to say that other phenomena cannot be studied. Indeed, once a transcript is in XML, its flexibility allows for the addition of custom annotations at many levels of granularity by adding custom elements and attributes.

Non-standard spellings and semi-lexical features, however, present a different problem. Anderson (2016) provides a detailed study of the conventions used across

multiple corpora and highlights issues with the multiple possible representations of these phenomena. For example, a word may be transcribed in its more colloquial form (e.g. *wanna*) or in a standardized form (e.g. *want to*). Similarly, filled pauses could be written in a myriad of ways (e.g. *er, ehm, uh, um*). It is desirable that such features be represented in a consistent manner for the purposes of quantitative study. Ideally, for quantitative analysis, researchers will decide on how best to represent these features at the onset of their studies. Therefore, Anderson (2016:343) recommends reducing the number of semi-lexical forms to a limited set, which can be defined by the researcher(s). When such guidelines are provided to transcribers consistency within a single project can be obtained. In the development of the XML schema we decided that flexibility was needed to allow transcribers to make these decisions depending on the goals of their projects. Thus, XTranscript does not attempt to normalize spellings of non-standard forms, but it does provide a mechanism by which a list of forms representing laughter, filled pauses, backchannelling and other ad hoc categories can be provided to the tool in an attempt to identify such features. A default list based on those identified by Anderson (2016) and Diemer et al. (2016) is also provided to the user. Thus, these features will be annotated in the XML so that they can be retrieved using XPath expressions (the XML tags are named either *laugh* or *semilexical*). It should, however, be noted that this method is never going to be exhaustive and in some cases may be inaccurate (for example, *huh* most likely represents backchannelling according to the summary given by Anderson (2016:332), but we've also seen it represent laughter in our test transcripts). Still this highlights a potential advantage of XML. Where such ambiguities exist in the original transcript, they can be resolved in the XML by editing the elements and assigning the correct designation. Specifically in regard to XTranscript, we hope that by making the specification of semi-lexical features customizable, the decisions researchers have made in the development of the CA transcripts can be better reflected in the XML versions.

## 6. Concluding remarks

In this paper we introduced the 'XML method' as an alternative method for analyzing CA transcripts. The method consists of three components: XML, the XML query languages XPath and XQuery, as well as XTranscript.

We outlined some basic characteristics of XML. Its network character was highlighted as its defining feature. We argued that working with CA transcripts formatted in XML has distinct advantages. These include the following. Any node or node set is exhaustively addressable and extractable from the network. This is where XPath and XQuery come into play as querying languages specifically designed to achieve data extraction from XML documents. Also, XML facilitates large scale analysis, whereas the method of analyzing single transcripts confines CA research to small sample investigation, which facilitates deep qualitative penetration but raises issues of generalizability. Third, XML offers unlimited filtering and combining capabilities for data extraction; that is, multiple numbers of specific target data can be addressed all in one go, thus enabling the analysis of how these data interact. Finally, XML supports distinctly enhanced quantification: any type and amount of (combinations of) nodes can be counted and extracted.

The third, and pivotal, element in the XML method is XTranscript, a recently-developed tool for automatically converting CA transcripts into the XML syntax. The XML elements provided by XTranscript are perfectly true to CA: XTranscript 'translates' the wealth of Jeffersonian codings that are standardly part of any CA transcription into the XML syntax with very little post-editing necessary.

We have discussed how previous projects have dealt with digital transcripts. Whilst these projects have made very valuable contributions to the study of transcribed speech, we noted that they do not provide us with the tools we require for the quantitative analysis of CA transcripts. Thus, we outlined our own approach. An approach which makes use of the essential features of XML as required for use with XPath, thus enabling the quantitative study of CA transcripts. Also, an approach which avoids the potentially very steep learning curve that the use of more complex XML technologies entails.

Our main point has been to introduce and justify what we call the XML method, a trio of technologies consisting of XML, XPath/XQuery, and XTranscript, that have the potential to move CA research a little closer to freeing itself from the 'very restrictive view' of CA as a purely qualitative discipline (Stivers 2015:16) and to adding to its toolbox a decidedly quantitative component.

## 7. References

Andersen, Gisle (2016): Semi-lexical features in corpus transcription. In: International Journal of Corpus Linguistics 21(3), 323-347.

British National Corpus Consortium (2007): British National Corpus version 3 (BNC XML edition). Distributed by Oxford University Computing Services on behalf of the BNC Consortium. Available online from http://www.natcorp.ox.ac.uk/XMLedition/ (retrieved 1 February 2018)

Carruthers, Jane (2008): Annotating an oral corpus using the Text Encoding Initiative. Methodology, problems, solutions. In: Journal of French Language Studies 18, 103–119.

De Ruiter, J. P. / Saul, Albert (2017): An appeal for a methodological fusion of conversation analysis and experimental psychology. In: Research on Language and Social Interaction, DOI: 10.1080/08351813.2017.1262050.

Diemer, Stefan / Brunner, Marie Luise / Schmidt, Selina (2016): Compiling computer-mediated spoken language corpora. In: International Journal of Corpus Linguistics 21(3), 348-371.

Goodwin, Charles (1984): Notes on story structure and the organization of participation. In: J. Maxwell Atkinson / John Heritage (eds.), Structures of social action: Studies in conversation analysis. Cambridge: CU Press, 225-246.

Hardie, Andrew (2014): Modest XML for corpora: Not a standard, but a suggestion. In: ICAME Journal 38, 73-103.

Heath, Christian (1984): Talk and recipiency: Sequential organization in speech and body movement. In: J. Maxwell Atkinson / John Heritage (eds.), Structures of social action: Studies in conversation analysis. Cambridge: CU Press, 247-265.

Hepburn, Alexa / Bolden, Galina B. (2013): The conversation-analytic approach to transcription. In: Jack Sidnell /Tanja Stivers (eds.), The handbook of Conversation Analysis. Malden/MA and Oxford: Wiley Blackwell, 57-76.

Hoey, Elliott M. (2017): Sequence recompletion: A practice for managing lapses in conversation. In: Journal of Pragmatics 109, 47-63.

Jefferson, Gail (2004): Glossary of transcript symbols with an introduction. In: Gene H. Lerner (ed.), Conversation analysis. Studies from the first generation. Amsterdam/Philadelphia: John Benjamins, 13-31.

Kirk, John. M. / Andersen, Gisle (2016): Compilation, transcription, markup and annotation of spoken corpora. In: International Journal of Corpus Linguistics 21(3).

Leech, Geoffrey / Myers, Greg / Thomas, Jenny (1995): Spoken English on Computer: Transcription, mark-up and application. New York: Longman.

MacWhinney, Brian (2000): The CHILDES Project: Tools for Analyzing Talk. 3rd Edition. Mahwah, NJ: Lawrence Erlbaum Associates.

Manning, Christopher D. / Surdeanu, Mihai / Bauer, John / Finkel, Jenny / Bethard, Steven J. / McClosky, David (2014): The Stanford CoreNLP Natural Language Processing Toolkit. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 55-60. Available online from (retrieved March 2018): https://nlp.stanford.edu/pubs/StanfordCoreNlp2014.pdf

Robinson, Jeffrey D (2007): The role of numbers and statistics within conversation analysis. In: Communication Methods and Measures 1(1), 65-75.

Rühlemann, Christoph (2017): Integrating corpus-linguistic and conversation-analytic transcription in XML. The case of backchannels and overlap in storytelling interaction. In: Corpus Pragmatics 1(3), 201–232. DOI: http://link.springer.com/article/10.1007/s41701-017-0018-7.

Rühlemann, Christoph / Bagoutdinov, Andrej / O'Donnell, Matthew B. (2015): Modest XPath and XQuery for corpora: Exploiting deep XML annotation. In: ICAME Journal 39, 47-84.

Rühlemann, Christoph / Gee, Matt / Ptak, Alexander (submitted): Multi-directional gaze in multi-party storytelling.

Sacks, Harvey (1984): Notes on methodology. In J. Maxwell Atkinson /John Heritage (eds.), Structures of social action. Cambridge: CU Press, 21-27.

Sacks, Harvey / Schegloff, Emanuel A. / Jefferson, Gail (1974): A simplest systematics for the organisation of turn-taking for conversation. In: Language 50(4), 696-735.

Santorini, Beatrice (1990): Part-of-Speech Tagging Guidelines for the Penn Treebank Project (3rd Revision).

Schegloff, Emanuel A. (1993): Reflections on quantification in the study of conversation. In: Research on Language & Social Interaction 26(1), 99-128. doi:10.1207/s15327973rlsi2601_5.

Schegloff, Emanuel A. (2000): Overlapping talk and the organization of turn-taking for conversation. In: Language in Society 29, 1-63.

Schmidt, Thomas / Wörner, Kai (2014): EXMARaLDA. In: Jacques Durand, Ulrike Gut, and Gjert Kristoffersen (eds.), The Oxford Handbook of Corpus Phonology. Oxford: Oxford University Press, 402-419.

Stivers, Tanja (2015): Coding social interaction: A heretical approach in conversation analysis? In: Research on Language and Social Interaction 48(1), 1-19. [L SEP]

Stivers, Tanja / Rossano, Federico (2010): Mobilizing response. In: Research on Language and Social Interaction 43(1), 3-31.

Stivers, Tanja / Sidnell, Jack (2013): Introduction. In: Jack Sidnell / Tanja Stivers (eds.), The handbook of Conversation Analysis. Malden/MA and Oxford: Wiley Blackwell, 1-8.

TEI Consortium, eds. 2018. TEI P5: Guidelines for Electronic Text Encoding and Interchange. Version 3.3.0. Last modified: 31 January 2018. TEI Consortium. Available online from (retrieved: 1 February 2018): http://www.tei-c.org/Guidelines/P5/

TEI Consortium, eds. 2018. Transcriptions of Speech. TEI P5: Guidelines for Electronic Text Encoding and Interchange. Version 3.3.0. Last modified: 31 January 2018. TEI Consortium. Available online from (retrieved: 1 February 2018): http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TS.html

Thompson, Paul (2005): Spoken Language Corpora. Developing Linguistic Corpora: a Guide to Good Practice. In: Martin Wynne (ed.), Developing Linguistic Corpora: a Guide to Good Practice. Oxford: Oxbow Books, 47-58. Available online from (retrieved March 2018): http://ota.ox.ac.uk/documents/creating/dlc/

Walmsley, Priscilla (2007): XQuery. Sebastopol, CA: O'Reilly.

Watt, Andrew (2002): XPath essentials. New York: John Wiley.

Wittenburg, Peter / Brugman, Hennie / Russel, Albert / Klassmann, Alex / Sloetjes, Han (2006): ELAN: a Professional Framework for Multimodality Research. In: Proceedings of LREC 2006, Fifth International Conference on Language Resources and Evaluation.

Wynne, Martin (ed.) (2005): Developing Linguistic Corpora: a Guide to Good Practice. Oxford: Oxbow Books. Available online from (retrieved 1 February 2018): http://ota.ox.ac.uk/documents/creating/dlc/

PD Dr. Christoph Rühlemann
Institut für Anglistik und Amerikanistik
Wilhelm-Röpke-Straße 6d
35032 Marburg

chrisruehlemann@googlemail.com

Mr. Matt Gee
School of English
Birmingham City University
The Curzon Building
4 Cardigan Street
Birmingham B4 7BD
United Kingdom

matt.gee@bcu.ac.uk

Veröffentlicht am 12.4.2018


## 8. Appendix: Tagging scheme underlying XTranscript as of March 2018

| Category | XML element | Sub-category | XML attributes & attribute values | CA symbol | Description |
|---|---|---|---|---|---|
| Sequential aspects | <sequence> | overlap: | <sequence type="overlap"> | **[ ]** | overlapped/over-lapping speech |
| | | | <sequence n=" "> | | id number of overlap |
| | | | <sequence part="1/2/..."> | | position of the overlap in a sequence of overlaps |
| | | | <sequence from=" " *or* to=" "> | | overlap in mid-word |
| | | latching: | <sequence type="latching"> | **=** | one turn latched on to next turn with less-than-usual or no gap at all |
| | | | <sequence position="start" *or* "end" *or* "within"> | | the position within the turn of the latch |
| Temporal aspects | <timing> | pauses: | <timing type="pause" duration=" "> | **(.)** or **(1.2)** | short or longer pause |
| | | speed-up: | <timing speed="faster" degree="much" *or* "more" *or* "most"> | **> a <** | increase in speed |
| | | slow-down: | <timing speed="slower" degree="much" *or* "more" *or* "most"> | **< a >** | decrease in speed |

| | | | | | |
|---|---|---|---|---|---|
| Phonological aspects | <voice> | intonation: | <voice intonation="rise"> | **?** | question(-like) rise |
| | | | <voice intonation="halfrise"> | **¿** or **?,** | rise stronger than a comma but weaker than a question mark |
| | | | <voice intonation="weakrise"> | **¿** | weakly rising intonation |
| | | | <voice intonation="fall"> | **.** | falling intonation |
| | | | <voice intonation="continued"> | **,** | continued intonation |
| | | | <voice intonation="level"> | **–** | level intonation |
| | | | <voice intonation="animated"> | **!** | animated tone, not necessarily an exclamation |
| | | pitch change: | <voice pitch="up"> | ↑ or ^ | sharp rise in pitch |
| | | | <voice pitch="updown"> | ↑ ↓ | sharp risefall in pitch |
| | | | <voice pitch="down"> | ↓ or \| | sharp fall in pitch |
| | | volume: | <voice volume="high"> | bold formatting | loud voice |

| | | | | | |
|---|---|---|---|---|---|
| | | | <voice volume="low" degree="much" *or* degree="more" *or* degree="most" > | **°a** | soft voice; three degrees |
| | | stretching: | <voice stretch=" " degree="much" *or* degree="more" *or* degree="most" word=" | **a::** | lengthened sound; three degrees; stretched letter and |
| | | stress: | <voice stress=" " degree="much" *or* "more" *or* "most"> | ***a*** or ***a*** or bold formatting | stressed or heavily stressed or very heavily stressed |
| | | realization: | <voice realization=" "> | | deviant realization of word |
| | | truncation: | <voice truncation=" "> | **-** | cut-off in mid-word |
| | | aspiration: | <voice aspiration="inhale" *or* aspiration="exhale"> | **.h** or **h.** | inhalation or exhalation |
| | | | <voice form="h" *or* "hh" *or* "hhh"> | **hh** | extent of aspiration |
| | | smile voice: | <voice quality="smile"> | **£** | talk produced while smiling |
| | | creaky voice: | <voice quality="creaky"> | ***** or **#** | words pronounced with a creak |
| | | tremulous voice: | <voice quality="tremulous"> | **~** | tremulous speech |
| Laughter | <laugh> | within-speech: | <laugh type="within-speech" word=" "> | **a(h)a** | laughing within words |

| | | | | | |
|---|---|---|---|---|---|
| | | | <laugh volume="high" *or* volume="low"> | **(H)** or **(h)** | loud or soft within-speech laughter |
| | | between-speech: | <laugh type="between-speech" form=" "> | e.g. **h**, **ha**, **ho**, **heh** | laughing between words |
| | | | <laugh volume="high" *or* volume="low"> | **H** or **h** | loud or soft between-speech laughter |
| Comments | <comment> | on hearing: | <comment hearing="unclear"> | **( )** | unclear hearing |
| | | | <comment hearing="possible"> | **( a )** | possible hearing |
| | | | <comment hearing="alternative" alternative=" "> | **( a / b)** | alternative hearings; specified in 'alternative' attribute |
| | | on event: | <comment event=" "> | **(( ))** | extra-linguistic event |
| | | on other: | <comment other=" "> | | other types of comment |
| Gaze | <gaze> | direction: | <gaze to=" " duration=" "> | **Xname 1.3** | gazed-at participant; and duration |
| | | | <gaze to="down" duration=" "> | **X↓ 1.3** | downward gaze; and duration |
| | | | <gaze to="up" duration=" "> | **X↑ 1.3** | upward gaze; and duration |

| | | | | | |
|---|---|---|---|---|---|
| | | | <gaze to="side" duration=" "><br><br><gaze to="shift" duration=" "> | **X← 1.3** or **X→ 1.3**<br><br>**X 1.3** | sideways gaze away from participant(s); and duration<br><br>shifting gaze; and duration |
| Gesture | <gesture> | hand:<br><br>facial: | <gesture type="hand" description=" " duration=" "><br><br><gesture type="face" description=" " duration=" "> | | description and duration of hand gesture<br><br>description and duration of facial expression |