Separating the sheep from the goats: Clarifying corpus content using XML

Stefan Evert and Manuela Schönenberger
Cognitive Science Department / IMS
University of Osnabrück / University of Stuttgart
stefan.evert@uos.de / manuela@ims.uni-stuttgart.de

1 Introduction

In a study of the acquisition of verb-placement in Swiss German, two children were recorded over a period of 150 hours. The recordings were transcribed and stored as Word documents. After a laborious process of re-arranging and counting almost 6000 utterances, a striking pattern emerged: the children adopt verb-movement while adults do not. Only after age 5;0 does the children's verb-placement become target consistent (Schönenberger 2001).

Over time, further annotations were added to the Word documents and new hypotheses were formulated and tested. For each experiment, frequency tables had to be compiled manually, sometimes limiting options for the statistical analysis (e.g. the age of the speaker can be inferred from the recording dates with much finer granularity than listed in the frequency tables). Since this procedure is time-consuming and error-prone, the need for another solution quickly became obvious.

In order to make the data amenable to automatic processing, we transformed them into an XML-encoded corpus, combining diverse information sources such as annotation codes, group headings, and file names. The translation process was largely automatic, using a combination of XSLT stylesheets and Perl scripts. However, consistency checks carried out by the translation scripts detected a considerable number of mistakes and formatting errors that had to be corrected manually in the original Word documents.

The resulting XML corpus can be searched, re-formatted and tabulated (for frequency counts) with the help of standard software such as an XSLT processor and Web browser. In the future, we plan to enrich it with further annotations, which will not interfere with the existing markup (in contrast to the previous Word documents, where it became increasingly difficult to add new information without rendering the transcription lines cluttered and unreadable).

2 Description of the data

In a research project on language acquisition the question of how Swiss-German speaking children acquire verb placement was investigated. Swiss German, as German and Dutch and its related dialects, is a Verb Second (V2) language that displays the verb-final pattern in embedded clauses. Since verb placement in matrix clauses does not coincide with that in embedded clauses it seemed likely that a child's acquisition of target-consistent verb placement would be a relatively difficult and time-consuming task. To study this question, two children, Moira and Eliza, acquiring Lucernese Swiss German were observed from age 3;10 up to age 8;01. The findings were surprising: both children produce target-consistent V2 in matrix clauses, but move the finite verb in all embedded contexts before age 4;11, giving rise to a large number of verb-placement errors, which conflicts with the hypothesis that children set parameters governing word order very early (Wexler 1999).

In the acquisition literature the claim that German children use the verb-final pattern as soon

as they start to produce embedded clauses is widespread. This claim originates with Clahsen and Smolka (1985) and was later supported by a detailed study carried out by Rothweiler (1993), who analysed the natural production data of young German-speaking children, who correctly use the verb-final pattern in all embedded contexts. Rothweiler's corpus comprises over 800 embedded clauses produced by seven children, ranging in age from 2;09–5;06. Only in two of these embedded clauses does the verb placement not coincide with the target grammar. Gawlitzek-Maiwald, Tracy and Fritzenschaft (1992) provide some counter-evidence based on one monolingual child called Benny, who sometimes places the finite verb incorrectly. During this stage of development Benny also produces many embedded clauses with the verb-final pattern. Brandt and Diessel (2004) report that the two German-speaking children of their study produce verb-placement errors in relative clauses.

Little is known about the acquisition of verb placement in embedded clauses of Dutch. Krikhaar (1992) discusses early embedded clauses in Dutch in which the subordinating element is usually dropped, a phenomenon which has also been noticed in other languages (see e.g. Müller and Penner 1996). Although in these embedded clauses the subordinating element is non-overt, the finite verb occupies the verb-final position in agreement with adult Dutch.

Based on a longitudinal study of Bernese Swiss German, Penner (1996) shows that the child's linguistic development which he had been following for several years is U-shaped: embedded clauses start to appear shortly before the child's 2nd birthday. All of these embedded clauses show the verb-final pattern. However, at a later stage in development the child starts to vacillate between moving the finite verb and leaving it in the clause-final position. This period lasts for several months (3;02–3;11). Around age 4;0 the child finally switches to the verb-final pattern. Penner's analysis is based on a corpus which consists of over 1100 utterances, most of which, but not all, are embedded clauses. Some of the embedded clauses were produced during elicitation tasks and therefore cannot be classified as spontaneous production data.

In summary, none of the children discussed in these studies behave like Moira and Eliza, who generally move the finite verb in embedded clauses and do not produce the verb-final pattern before age 4;11.

3 The status quo

3.1 Data collection and transcription

The data of these children were extracted from audio recordings made by Moira's mother. A tape recorder was used to record the children while they were playing or when Moira interacted with other members of her family, mainly her mother. In total there are 100 tapes of 90 minutes each, amounting to 150 hours of recorded data. Most of these data pertain to Moira. Her corpus consists of over 5500 embedded clauses produced spontaneously and several hundred elicited embedded clauses. Eliza's corpus is much smaller, but still contains over 600 spontaneous embedded clauses and several hundred elicited embedded clauses.

To analyse the acquisition of verb placement, all embedded clauses produced by the children were transcribed. To increase accuracy in transcription, each tape was listened to at least twice and inconsistencies in transcription were corrected. The transcription of these embedded clauses was stored in Microsoft Word documents, which we refer to as 'overview files'. All embedded clauses were transcribed in the order in which they appeared on the tape,

i.e. in chronological order. For ease of reference, the tape number and position at which an utterance occurred was also written down to make it possible to find a specific utterance on a tape without having to listen to the whole tape again. An excerpt of the overview file is reproduced in Figure 1. As shown there, the date at which a recording was made is usually mentioned (by the mother) on the tape and has been included in the transcription. In the overview files, each utterance which was of interest, either because it contains an embedded clause or an interesting formulation, was transcribed. Text formatting was used to provide additional information for utterances with an embedded clause, where the subordinating element is highlighted in bold face and the finite verb underlined. The speaker, unless it is Moira, is indicated by an initial or in full, and precedes the utterance. If the speaker is an adult, an initial was usually used, e.g. G: stands for the name of Moira's mother. Sometimes small dialogues were transcribed, as in (14), (15), (18) and (21), for example. This was usually done when an embedded clause introduced by *wil* 'because' was used by the child. In the target grammar, clauses introduced by *wil* allow for two different verb-placement patterns: V2 or verb-final, which crucially depend on the context of utterance. By giving the context in which the child used a given pattern it could later on be determined whether the verb placement was felicitous or not.

The recordings were made over a period of more than 4 years, and the transcription (which started immediately after the first tape had been completed) took even longer.

13 February 95

(4)	G: Jezt tüemmer immer schön s'Datum säge	4B (222)
(5)	S'Muul isch de Güselchübel **wo** chammer alles drischtecke	4B (223)
(6)	**Wenn** isch ganz Summer denn chammer chalti Chleider aalegge	4B (233)
(7)	De tuet t'Sunne immer am glichlige Ort si **wo** sind t'Lüt, di tuet nämlich tuet nämlich alles ufwärme	4B (235)
(8)	T'Sunne tuet immer wärmer und wärmer mache	4B (238)
(9)	Und **wenn** isch's chalt muesch warmi Chleider aalegge	4B (240)
(10)	das grüene **wo** isch chüeler	4B (265)
(11)	und meh und meh **wil** de Geifer isch chalt	4B (280)
(12)	und de **wo** sind Puebe det gsi hämmer gluegt	4B (289)
(13)	De simmer abeglofe go luege sind Puebe det	4B (304)
(14)	G: Dä hät chöne selber flüge.Worum? M:**wil** er hät Flügel gha (G: wil er Flügel gha hät)	4B (331)
(15)	G: Worum hät er möge? M: **wil** er isch en grosse Hund gsi	4B (334)
(16)	doch no eini chlini Chatz	4B (338)
(17)	und **wenn** tüend's esse machet's wie t'Hünd	4B (414)
(18)	G: Worum hät er ned chöne schäle? M: **wil** s hät ka Händ gha	4B (420)
(19)	M: **wil** er hät nu chönne beidi Füess neh	4B (426)
(20)	G: T'Affe gfallet der ned? Worum? M: **wil** si händ so äs grusigs Fäli	4B (477)
(21)	G: Hüt nomitag gömmer i Poscht. M: I wot ned mitcho. G: Worum ned? M: **wil** isch mer langwilig	4B (550)
(22)	M: Aber Fernseh luege wot ich denn. G: Wenn? M: gosch du uf Poscht	4B (557)
(23)	G: Hüt isch Mäntig und du gosch ned is Zwergehuus. Worum?	4B (600)

Figure 1. Excerpt from overview file.

3.2 Annotation and quantitative analysis

In order to obtain quantitative data for verb placement, the transcribed utterances were rearranged into separate files by cut & paste, preserving the highlighting of the subordinator (bold face) and finite verb (underlined). First, they were allocated to different 'subcategorization files', depending on the type of subordinator. Clauses containing the complementizer *dass* 'that' were stored in the subcategorization file DASS, clauses introduced by *wil* 'because' in the subcategorization file WIL, relative clauses introduced by *wo* 'which'

referring to a subject in SUBJECT RELATIVE WO, etc. Thus the subcategorization file DASS contained all embedded clauses containing the complementizer *dass* produced by either of the two children. The data in each of these subcategorization files were further subdivided into the type of verb placement used (i.e. V2, V3, ambiguous, verb-final) and then listed under appropriate subheadings giving the age of the children, grouped by month**.** Since the children are only one week apart in age, Moira's age was used for grouping utterances from both speakers. This categorization of the data made it possible to compute, e.g., how many embedded clauses Moira and Eliza produced at age 4;10 with the complementizer *öb*, and in which the verb placement is V3 (i.e. sentences with the pattern *öb subject Vfin* …). For instance, by counting the utterances in the excerpt shown in Figure 2 , we find that eight embedded clauses with the pattern *öb subject Vfin* were produced at age 4;10 by Moira (the default speaker), and one by Eliza (whose name is highlighted in bold face). Two of Moira's utterances are preceded by the code 'U*', in which U stands for unclear verb placement and * stands for non-target-consistent. Thus, although it cannot be determined whether the verb placement falls into category V2 (*öb Vfin* …) or V3 (*öb subject Vfin* …) because the subject is missing, both alternatives are ruled out in the target grammar and hence verb placement is non-target-consistent.

	4;10 (26A(361) - 31B (070))	
	tüemer luege **öb** s isch richtig	26B (580)
	me müend doch luege **öb** mer händ Schnee i de Händ oder ned	28A (226)
	und etz dört wo-n-ich mischle und du törfsch ned gsee **öb** ich ha es Büsi	28B (028)
	zu de Eliza abe go chlöperle **öb** si chunt achli ufe	28B (322)
	aso nochether go-n-ich zu de Eliza abe go chlöperle **öb** si chunt achli ufe und de luege i de was ich chönt schpile mit ere	28B (390)
U*	Söllet mer luege **öb** häsch recht	26B (461)
U*	Also etz mues ich nur no ä Gans ha - **öb** isch die	28B (024)
	Eliza: mues halt luege **öb** ich find nomel öppis	29B (005)

Figure 2. Excerpt from subcategorization file for *öb* (subsection *öb subject Vfin* …).

Not only was the transcription of the data a very time-consuming and labour-intensive task, which alas cannot be avoided or automated, classifying the embedded clauses according to type of subordinator, verb placement and age was also rather tedious, since it had to be done manually by copying the relevant utterances from the overview files and then pasting them into the appropriate subcategorization files, in which they then had to be allocated to the appropriate section according to verb placement and age.

All of the data which occurred before age 6;01 (tapes numbered 1–75) were used by Schönenberger (2001) to argue against the hypothesis that acquisition of word order is basically error-free and to adduce evidence in favour of one of the many competing analyses of what is the underlying structure of Swiss German (and German). In Schönenberger and Evert (2002) the same data were subjected to a statistical analysis in order to determine at what age Moira switches from a verb-movement grammar to a non-verb movement (i.e. verb-final) grammar. The frequency counts for these analyses were obtained manually, by going through the subcategorization files and counting the relevant utterances.

As time progressed, more data were collected and transcribed, and more detailed questions concerning the children's verb placement arose, as, for example, whether there is an interaction between type of subject (pronominal vs. non-pronominal) and type of finite verb. To pursue such questions, additional coding was introduced in the subcategorization files to identify the verb-type (e.g. lexical verb, auxiliary-like verb *ha* 'have', modal verb) and subject-type (pronominal, non-pronominal, non-overt). Moreover, all embedded clauses were

enclosed in brackets, and information about the verb position provided by subsection headings was reproduced in the codes. This is shown in Figure 3 for an excerpt from the subcategorization file for the subordinator *wenn* 'when, if'. As can be seen from this excerpt, all utterances at age 4;05 involving the complementizer *wenn*, and in which the verb is in 3rd position (*wenn Subject Vfin …*) were produced by Moira. In three of these utterances a finite lexical verb is used (L), in two the main verb *ha* 'have' appears (Have) and in one an auxiliary verb (Aux). In all six utterances the subject is pronominal (-). Moreover, in two of the codes an exclamation mark appears, pointing out something of interest that is not necessarily related to verb position, verb-type, or subject-type. Here it indicates that the element intervening between the complementizer *wenn* and the misplaced finite verb contains a pronominal subject as well as a pronominal object (forming a clitic cluster).

4;05 (14A (338) - 15B (407)

3L-	wott ich emol zueluege – [**wem**mer macht das]	14B (413)
3L-	ich meine [**wenn** si macht Ufgobe] wott i au ned ie	15B (316)
3L-	[**Wenn** si macht es Gschenk] de törf i ebe ned ine	15B (318)
3Have-!	oder zum Bischpil [**wem***mer's* hät ned gern]	14B (389)
3Have-!	oder [**wem***mer's* hät gern]	14B (390)
3Aux-	[**Wenn** ich heet jetz achli z'Trinke ineglert] was het de jetz das geh	14B (560)

Figure 3. Excerpt of subcategorization file for *wenn*, with coding of verb position, verb-type and subject-type.

The additional coding introduced in the subcategorization files was used to obtain some insight into a possible interaction between verb-type and subject-type. For this purpose, more elaborate frequency counts had to be performed, once again by going through the subcategorization files manually. Four separate data sheets were used for the main types of embedded clauses (wh-complements, relative clauses, clauses introduced by a complementizer, clauses introduced by *wil*) for each child. In each sheet, a separate frequency count was obtained for every combination of verb-type (lexical vs. non-lexical), subject-type (pronominal vs. non-pronominal) and age, in the following way. The code preceding each utterance in the individual subcategorization files was examined and a tally was kept in the appropriate box, as shown in Figure 4 for the data from Figure 3. The entries in the individual boxes were then summed up to obtain the final frequency counts.

Clauses introduced by a complementizer (Moira)

	lexical		non-lexical	
	pron.	non-pron.	pron.	Non-pron.
…				
4;04				
4;05	|||||		|	
4;06				
4;07				
…				

Figure 4. Excerpt of data sheet for clauses introduced by a complementizer

3.3 Problems of this approach

Obviously, the approach taken in this work to the annotation and quantitative analysis of the corpus data has a number of serious drawbacks. For the statistical analyses, it is essential to have accurate frequency tables for a cross-classification of all embedded clauses produced by one of the speakers, according to two or usually more features (e.g. by subordinator vs. subject-type vs. verb placement vs. age). The structure of the subcategorization files (i.e. how utterances are grouped together in these files) makes it relatively easy to obtain such tables for

the combination of subordinator, verb placement and age. However, any other combination of features (e.g. the classification according to subject-type and verb-type reported at the end of Section 3.2) necessitates a very laborious and error-prone manual counting procedure, in which the complex code strings have to be 'parsed' by the human annotator, requiring a high level of concentration. This counting procedure has to be repeated from scratch whenever a new potentially relevant interaction between features is hypothesized.

In addition to the difficulty of producing frequency counts, the current form of the corpus does not provide an easy way of refining and extending the existing annotations. New annotations can either be added to the subcategorization files by a further subdivision of the grouping (which is impractical, given that most groups already contain only a few sentences) or by extending the code string at the beginning of each line (as shown in Figure 3, where the codes have been extended to indicate verb-type). However, this will quickly lead to bloated codes that are difficult to read and edit. Some annotations have already been inserted at the end of the lines for this reason, e.g. codes that indicate the type of relative clause (with 'O' for object-relative clause, 'T' for temporal relative clause, etc.). It is much harder to locate and count such annotations accurately when they are not aligned with the beginning of lines. A particular problem of the extended annotation codes is that they must be unambiguous, i.e. a unique combination of characters has to be used for every annotated phenomenon and feature value. If the same character sequence were to be used for different phenomena, it would have to be disambiguated by its position in the code string. In any case, a single wrong keystroke during the annotation process may inadvertently falsify or even delete previous annotations.

Utterances also have to be annotated in the order in which they are listed in the subcategorization files, where the codes are inserted. This makes it nearly impossible to check new annotations against the original sound recordings (e.g. to find out whether pauses and intonation support a hypothesized syntactic analysis), which would require constant rewinding and changing of the tapes. Sometimes, it would also be helpful to group the utterances in a different way during the annotation procedure, e.g. when the codes for a certain phenomenon are refined. This is in fact what was done for the study by Schönenberger and Evert (2002). When examining the data it became clear that up to a certain age pronominal subjects and non-pronominal subjects do not pattern alike in clauses introduced by a complementizer with verb movement. The subcategorization file COMPLEMENTIZER was therefore re-examined, and each utterance was coded with respect to subject-type, adding symbols '+' for non-pronominal subjects and '-' for pronominal ones. To make manual counting easier, utterances were rearranged within each section, so that those with non-pronominal subjects would precede the ones with pronominal subjects, which in turn would precede those in which a subject pronoun was omitted. By regrouping the utterances, counting was made easier and mistakes could be discovered more easily, but the rearrangement was a very labour-intensive manual process.

These problems have prompted us to transform the collection of overview and subcategorization files into a 'real' electronic corpus that makes all the information coded in the Word documents directly available for automatic processing, and that can easily be extended with new annotations. The main objectives for the new corpus were the following:

1. The corpus should be encoded in a standard format that can be processed automatically with existing software (especially with respect to frequency counts, annotation updates, and the selection and grouping of utterances) and is suitable for long-term storage.
2. All annotated information should be made explicit in the corpus, rather than being hidden in the group headings (age of the speakers) or even in the names of the

subcategorization files (as it is the case for the type of subordinator and default speaker). The complex annotation codes, which conflate information for many different and largely unrelated phenomena (e.g. verb placement and subject-type), should be factored out into separate feature-value pairs for each information type. These requirements are essential for automatic processing and also make the coding more legible for human readers.

3. The corpus should make it possible to add new annotations independently from existing ones. In particular, there should be no danger that old annotations are falsified or deleted by accident.
4. Ideally, the new corpus should also support the validation and correction of annotations. This can be done fully automatically (such as testing that recording dates are consistent, or ensuring that all utterances have complete annotations), by focusing on a single phenomenon at a time, perhaps supported by colour-coding and other formatting, or by rearranging utterances automatically into homogeneous groups where any structure that is different from the rest of the group will stand out.

4 Transformation into an XML-encoded corpus

4.1 Rationale and XML encoding

In addition to requirements such as those listed at the end of Section 3.3, the choice of a suitable encoding for a corpus is ultimately determined by the data structures that have to be represented. As a first step it is therefore necessary to describe the abstract data model underlying the corpus. In our case, the corpus consists of a sequence of transcribed utterances with some syntactic markup (embedded clause, subordinator, and main verb), as seen e.g. in Figure 3. Each utterance is uniquely identified by tape number (including A or B side) and position. It is annotated with a set of feature-value pairs that collect information from the different sources (annotation codes, speaker indications, section headings and file names) about the date of recording, speaker, type of subordinator, verb placement, type of finite verb, type of subject, etc. Feature values are atomic (rather than complex feature structures), i.e. they can be represented as simple character strings. Although there are often several features relating to the same type of information (e.g. day, month and year for the date of recording), there is no need to make these connections explicit in the formal data model. We decided to indicate groupings by choosing suitable feature names, using a path notation such as `date/month` and `date/year`. Figure 5 shows the abstract representation of utterance 15B (316), whose original entry in the appropriate subcategorization file can be seen in Figure 3. At the current stage, we use the respective parts of the annotation code `3L-` as feature values for verb placement, verb-type and subject-type. We plan to replace them with more explicit labels, but the precise inventory of feature values has not been decided yet.

`15B (316)`	
`date/day`	10
`date/month`	8
`date/year`	1995
`speaker`	Moira
`subord/type`	comp
`vpos/place`	3
`verb/type`	L
`subj/type`	-
ich meine [$_{CLAUSE}$ [$_{SUBORD}$ wenn] sie [$_{VERB}$ macht] Ufgobe] wott I au ned ie	

Figure 5. Abstract representation of utterance 15B (316) from Figure 3 with annotations.

Together with the requirements listed at the end of Section 3.3, the data model exemplified in Figure 5 made XML, the extensible markup language (Yergeau *et al.* 2004), an obvious and highly suitable choice for the encoding of our corpus. Feature-value pairs can naturally be represented by XML elements or attributes, and the transcribed utterance with markup is a typical example of the mixed content model used by narrative-oriented XML documents (Harold and Means 2001: 85–97). XML has become a firmly established standard, which means that a corpus encoded in this format will remain accessible without having to rely on a specific (commercial) software product. In principle, any text editor is sufficient to access the information stored in an XML document. A wide range of software is available for processing XML data, most notably the XSLT stylesheet transformation language (Clark 1999) that can be used to modify XML files, extract specific information from the files, and translate them into other formats such as HTML or plain text. XML parser libraries, which are available for many programming languages (including Java and Perl), allow XML documents to be read and created with ease. Since XML is designed to be extensible, new annotations can be added to the corpus that do not interfere with existing ones. Validation of the corpus format and the annotations is possible by specifying a document type definition (DTD, see Harold and Means 2001: 26–57) or in a more sophisticated way by using the XML Schema language (Thompson *et al.* 2004; Biron and Malhotra 2004).

When devising an XML document format, one inevitably has to strike a balance between conceptual clarity, validatability, and ease of use. For our purposes, we found simple and convenient access to the data to be the most important criterion. This led us to the unusual solution of encoding features names as element paths, i.e. as nested XML elements with the same names. In this way, the XPath expressions (Harold and Means 2001:147–167) used to access feature values are identical to the names of the respective features. An example of our XML format is shown in Figure 6, for the same utterance as in Figure 5 (edited slightly for clarity).

```
<U id="015B (316)">
  <F>
    <speaker>Moira</speaker>
    <date>
      <day>10</day>
      <month>8</month>
      <year>1995</year>
    </date>
    <vpos>
      <place>3</place>
    </vpos>
    <verb>
      <type>L</type>
    </verb>
    <subj>
      <type>-</type>
    </subj>
    <subord>
      <type>comp</type>
    </subord>
  </F>
  <TEXT>
    ich meine <CLAUSE> <SUBORD>wenn</SUBORD> si <VERB>macht</VERB>
    Ufgobe </CLAUSE> wott i au ned ie
  </TEXT>
</U>
```

Figure 6. XML representation of utterance 15B (316).

Our XML encoding also uses short and uppercase element names for the ‘backbone’ structure to make XPath access more convenient. For instance, the value of the `verb/type` feature is selected by the XPath expression `F/verb/type`, and the embedded clause in the transcription of the utterance by `TEXT/CLAUSE` (both relative to the `<U>` element

representing the entire utterance). Each `<U>` element is uniquely identified by its `id` attribute, so that a specific utterance can easily be extracted from the corpus.

The XML format shown in Figure 6 has some disadvantages. The encoding of feature names as element paths requires that their components (e.g. `verb` and `type` for `verb/type`) must be valid XML element names, and care has to be taken to avoid conflicts with other element names in the corpus. It is also impossible to validate such XML files against a generic document type definition, because the DTD has to list all possible element names explicitly. When new attributes are added to the corpus, it is no longer valid until the DTD is updated to include the new element names (thereby breaking the validity of earlier versions of the corpus). Furthermore, the corpus cannot be processed with generic XSLT stylesheets that do not include explicit feature names.

We believe that these drawbacks are tolerable, since the corpus will mostly be used as a read-only resource for search, inspection and data extraction. We do not envisage direct editing of the XML corpus or automatic modifications with generic XML processing tools. All updates and extensions of the corpus annotation should be made by specialized scripts in order to ensure consistency and avoid loss of information. These scripts can also perform a thorough validation of the corpus format without the need for an external DTD. When necessary, such a DTD can always be generated automatically from an inventory of all the features present in the current version of the corpus.

4.2 Automatic transformation procedure

In a first step, the original Word documents were regularized to simplify the translation process. We made sure that all utterances include complete annotation codes, tape positions and transcriptions with syntactic markup, and that the speaker is always indicated clearly. Particular care was taken to separate transcriptions from tape numbers and annotation codes by tab stops. Many of the utterances contain several embedded clauses. These complex utterances had to be split in the subcategorization files, so that each entry would contain only a single embedded clause to which the annotation codes refer. The fact that an embedded clause is part of a more complex utterance was indicated with an ellipsis '…' in the subcategorization file, as in utterance 24A (284) in Figure 7. In some other cases, fragments of preceding utterances were included to give some context. Highlighting and bracketing for these fragments was removed in order to avoid confusion, as in the example of 27A (159), where the wh-complement is highlighted but not the preceding *wil*-clause.

	4;9 (23B (393) - 26A (361)	
2L-	und wotsch achli lose oder [**was** säget's] ...	24A (284)
2M-	... und er weiss ned [**wo** chammer jo lüter aalo] denn ghörsch de Ton grad ned, de bellet eifach dri	24A (284)
	4;10 (26A (361) - 31B (070)	
2M-	G: Worum häsch gwüsst weli Nummere? M: wil Tabea hät mir gseit [**weli** mues ich ufmache]	27A (159)

Figure 7. Excerpts from the modified subcategorization files.

After the initial clean-up, we used the OpenOffice suite (http://www.openoffice.org/) to convert the Word documents into a manageable XML representation (compared to the XML and HTML output of MS Word, which is completely overloaded with internal formatting information). For this purpose, we saved the documents in the native OpenOffice format,

extracted the file `contents.xml` from the compressed archive, and used an XSLT stylesheet to reduce it to the relevant information (style definitions and ‘paragraph’ markings, which delimit lines in the overview and subcategorization files). A detailed analysis of these simplified XML files was then performed with separate Perl scripts for the overview and subcategorization files. For the overview files, recording dates given on separate lines had to be recognized (using a date parser module from CPAN), the speaker had to be identified for each utterance (in bold face at the beginning of a line, and followed by colon), and an ID had to be computed from the tape number and position. For the subcategorization files, the main task was to recognize and parse the annotation code at the start of each line, as well as to computed an ID that can be used to relate the annotated utterance to the corresponding entry in the overview files.

The next step was to merge the information from subcategorization files (annotation code) and overview files (date of recording). The transcription itself as well as speaker identification is present in both types of files and was used for consistency checks. Matching was based on the computed IDs, and most of the information was taken from the subcategorization files, adding only date information from the overview files. The combined data were then transformed to the XML format described in Sec. 4.1 and saved in a disk file. This file contains 5908 embedded clauses (5357 by Moira, 551 by Eliza) and has an uncompressed size of 4.6 Mbytes.

A fundamental problem in the automatic translation process is that multiple utterances may share the same ID because of the low resolution of the tape position counter. This problem was compounded by utterances like 24A (184), which had to be split because they contain multiple embedded clauses. In such cases, the IDs were extended automatically in order to be unique in the corpus. For example, the ID of the second line in Figure 7 was changed to 24A (184)a. Since utterances with the same tape position may be spread across different subcategorization files (for different speakers) and since utterances were split in the subcategorization files but not in the overview files, the resulting IDs were not consistent between the two types of files. Therefore, ‘fuzzy’ matching had to be used for the merging process, based on a comparison of tape number and position, speaker, as well as the full transcription (which had to account for minor variations and split utterances in the subcategorization files).

During the translation process, extensive consistency checks were carried out in order to ensure a high quality of the resulting XML corpus and detect errors introduced by the translation scripts. In addition to matching speaker names and transcriptions between the overview and subcategorization files, we validated speaker names and annotation codes against exhaustive lists of possible values. We also made sure that syntactic markup in the transcriptions is well-balanced (especially the brackets marking subordinate clauses), which is a prerequisite for the XML encoding, and we tested that higher tape numbers correspond to later recording dates. All problems detected by the translation script were recorded in log files with a detailed description of their place of occurrence, so that they could easily be corrected in the original Word documents. The most frequent error in the overview files concerned unmatched brackets, in particular closing brackets that had been forgotten. Another frequent error involved brackets that were accidentally highlighted in bold face or underlined, a problem that would have been difficult to detect by eye. This process was iterated until no more errors were detected.

5 Working with the XML corpus

5.1 Presentation and data extraction

The primary uses of the new XML corpus are (i) searching for utterances with specific annotations, (ii) presentation of search results with highlighting and sorted or grouped in a meaningful way, and (iii) extraction of data tables as a basis for statistical analysis. All three tasks can easily be accomplished with XSLT stylesheets. XPath conditions can be used to select individual utterances according to various criteria. For instance, the expression

```
U[F/speaker = "Moira" and F/vpos/type = "amb"]
```

matches all utterances by Moira where verb placement is ambiguous. Such XPath conditions can be embedded in XSLT stylesheets or applied directly by an application program using a standard XML parser library. The results of a corpus search can be sorted and reformatted with an XSLT stylesheet for presentation to the user. It is easiest to generate HTML output in this way, which can then be imported into other applications such as MS Word. Figure 8 gives an example of HTML output for the utterance 15B (316) from Section 4.1 (compare this to the the much less readable XML representation in Figure 6).

015B (316): Moira, 10.08.1995

verb placement: 3 / verb type: L / subject type: – / subordinator: comp

ich meine **[wenn** si macht Ufgobe **]** wott i au ned ie

Figure 8. HTML presentation of utterance 15B (316).

Quantitative data for statistical analysis can be extracted in the form of tables that list the values of selected features (as columns) for all or some utterances (as the rows of the table) in the corpus. From such a table, frequency tables for cross-classifications of the corpus data with respect to different feature combinations (which are the basis for the application of hypothesis tests and statistical models) can easily be computed with statistical software packages, e.g. Gnu R (R Development Core Team 2005). All that is required is thus a translation of the feature annotations into a tabular text format, which again can be achieved with an XSLT stylesheet. Figure 9 shows an example of such a statistical table, listing type of subordinator, verb placement, verb-type, subject-type and date of recording (as a real number measured in years, so 1995.0301 stands for January 11th, 1995) for some of Eliza's utterances.

subord	verb.place	verb.type	subj.type	date
comp	1	M	-	1995.0301
rel	2	M	-	1995.0301
wh	2	Tun	-	1995.0301
comp	1	Aux	+	1995.0301
comp	2	Have	-	1995.0301
comp	2	Have	-	1995.0301
comp	2	Have	-	1995.0301
wh	2	L	-	1995.0301
wh	2	Tun	-	1995.0301

Figure 9. Example of statistical table extracted from the XML corpus.

5.2 Enriching and improving the annotation

Another goal of our efforts is to add new or more detailed annotations to the corpus. Examples are 2nd person singular pronouns, which can appear in three different forms: the pronoun *du*, the clitic *t*, or *pro* (the non-overt pronominal form), as well as the specification of

the auxiliary verb (*ha* 'have' or *si* 'be') used. The interface used for this purpose should be easy to operate for non-technical users, not be tied to a specific computing platform, and preferably not require an internet connection during the annotation work. Our strategy is to export (part of) the corpus to a tabular text format (similar to that of Figure 9), which can be loaded into a standard spreadsheet application (e.g. MS Excel or the OpenOffice spreadsheet) and annotated there. The new annotations are then merged back into the corpus based on utterance IDs, with the help of a Perl script that performs extensive validation and records all changes in a log file.

We also plan to make annotations more explicit by further expansion of the compact annotation codes into their 'component meanings' (e.g. # implies omission of 2nd person singular pronoun, while ## implies omission of non-2nd person singular pronoun), and substituting single-letter codes and symbols with meaningful values. This is particularly important when corpus searches are performed with XPath expression, due to the limited support for string manipulation and regular expression matching in current implementations of the standard. Such automatic expansions may be ambiguous and will need to be checked and clarified manually.

Over time, the quality of the corpus will be further improved by repeated error checking. One approach is to arrange the utterances in homogeneous groups and display them in a form that makes incorrect annotations easy to spot (with heavy use of text formatting and colour coding). Another possibility is a targeted search for errors, using queries to find unlikely combinations of feature values (e.g. unambiguous verb placement when the subject has been omitted) and inconsistent syntactic markup (e.g. a marked subordinator or finite verb outside an embedded clause). The results of such queries can then be perused by a human expert. One specific quality measure planned for the near future is to compare the markup for the same utterance in the respective overview and subcategorization files so and resolve any inconsistencies found in this way.

6 Conclusion and future work

In this paper, we have shown how a collection of hand-written Word documents can be transformed into a highly useful XML corpus by a largely automatic translation procedure. Although a certain amount of manual corrections and tidying was unavoidable, most of this work was concerned with the inconsistencies and format errors detected by the translation scripts in the original files. The XML corpus supports convenient search and data extraction with standard software applications (all that is needed is an XSLT processor, a Web browser, and a spreadsheet or statistical software package), enabling research that goes far beyond the initial objectives of the data collection.

The first questions that we plan to address with this new resource are the following: (i) In an early phase (age 3;10–4;04), verb movement in Moira's embedded clauses results in the order: *subordinator Vfin subject* … . After age 4;04, Moira starts to produce a second non-target-consistent order in clauses introduced by a complementizer, i.e. *complementizer pron subject Vfin* … . Since the second pattern only occurs with pronominal subjects, the question arises whether the overall number of pronominal subjects in the early phase is significantly different from that in the later phase (age 4;04–4;11). In other words, whether the non-occurrence of the second pattern before age 4;04 is due to the fact that Moira mainly uses non-pronominal subjects. Moreover, only clauses introduced by a complementizer seem to show this second pattern. (ii) Schönenberger and Evert (2002) found that the switch to the verb-final pattern happened shortly after age 4;11 in Moira, but whether the switch happens in all types of embedded clauses at this time is less clear. It is unclear at what time Eliza

switches to the verb-final pattern, since her corpus is much smaller and recordings were made at less regular intervals. The more fine-grained age specifications that can be obtained from the XML corpus (especially for Eliza) may help to provide some answers to these questions.

The Moira XML corpus is still work in progress. In addition to the error checking and explication of feature annotations described in Section 5.2, the next steps to be tackled comprise the inclusion of utterances that have been omitted from the XML version so far. The excluded utterances are either sentence repetition tasks (which have a slightly different markup) or utterances that have been transcribed but not annotated (because they are from other speakers or contain no embedded clauses). Utterances from the second group are usually listed in the overview files to give some relevant context for annotated utterances. Our goal is to include all transcribed utterances in the corpus, so that it is a direct translation of the original overview files, but enriched with the information from the subcategorization files. When we have reached this stage, the original Word documents will be abandoned and all future work will be based on the XML corpus.

In the longer term, it is also planned to annotate embedded clauses by other speakers than Moira and Eliza (but only in utterances that have already been transcribed). For this purpose, all utterances by a given speaker can be extracted from the XML corpus (once it has been completed) and transferred to a Word document. There, they are annotated in the same way as the original subcategorization files so that they can easily be included in the automatic translation process. Finally, it is desirable to digitize the audio recordings before the tapes wear out and the original data are lost forever. Ideally, we will also be able to link the digital sound files to the corresponding utterances in the corpus, so that they can be played when utterances are presented to the user. A promising software solution for this purpose is the NITE XML Toolkit (Carletta *et al.* 2003), which uses a stand-off annotation scheme that our XML format can easily be adapted to.

References

Biron, P. V. and Malhotra, A. (2004). *XML Schema Part 2: Datatypes (Second Edition)*, W3C Recommendation, 28th October 2004. http://www.w3.org/TR/xmlschema-2/

Brandt, S. and Diessel, H. (2004) The development of relative clauses in German. Poster presented at *CLS* (University of Bristol, UK).

Carletta, J.; Kilgour, J.; O'Donnell, T.; Evert, S. and Voormann, H. (2003) The NITE object model library for handling structured linguistic annotation on multimodal data sets. In: *Proceedings of the EACL Workshop on Language Technology and the Semantic Web, NLPXML-2003* (Budapest, Hungary).

Clahsen, H. and Smolka, K.-D. (1985) Psycholinguistic evidence and the description of V2 phenomena in German. In: H. Haider and M. Prinzhorn (eds.), *Verb Second Phenomena in Germanic Languages* (Dordrecht: Foris), 137–167.

Clark, J. (1999). *XSL Transformations (XSLT) Version 1.0*, W3C Recommendation, 16 November 1999. http://www.w3.org/TR/xslt

Gawlitzek-Maiwald, I.; Tracy, R. and Fritzenschaft, A. (1992) Language acquisition and competing linguistic representations: The child as arbiter. In: J. Meisel (ed.), *The acquisition of verb placement* (Dordrecht: Kluwer), 139–179.

Harold, E. R. and Means, W. S. (2001) *XML in a Nutshell* (Sebastopol, CA: O'Reilly).

Krikhaar, E. (1992) Voeg woordloze bijzinnen in kindertaal. MA thesis, University of Utrecht.

Müller, N. and Penner, Z. (1996) Early subordination: The acquisition of free morphology in French, German and Swiss German. *Linguistics* **34**, 133–165.

Penner, Z. (1996). *From empty to doubly-filled complementizers: A case study in the acquisition of subordination in Bernese Swiss German*. Working Papers of the University of Konstanz 77.

R Development Core Team (2005*) R: A Language and Environment for Statistical Computing* (Wien: R Foundation for Statistical Computing), ISBN 3-900051-07-0. http://www.R-project.org/

Rothweiler, M. (1993) *Der Erwerb von Nebensätzen im Deutschen: Eine Pilotstudie* (Tübingen: Niemeyer).

Schönenberger, M. (2001) *Embedded V-to-C in child grammar. The acquisition of verb placement in Swiss German* (Dordrecht: Kluwer).

Schönenberger, M. and Evert, S. (2002) The benefit of doubt. Paper presented at *QITL* (University of Osnabrück, Germany).

Thompson, H. S.; Beech, D.; Maloney, M.; and Mendelsohn, N. (2004). *XML Schema Part 1: Structures (Second Edition)*, W3C Recommendation, 28th October 2004. http://www.w3.org/TR/xmlschema-1/

Wall, L.; Christiansen, T. and Schwartz, R. L. (1996). *Programming Perl, 2nd Edition* (Sebastopol, CA: O'Reilly).

Wexler, K. (1999). Very early parameter setting and the unique checking constraint: A new explanation of the optional infinitive stage. In A. Sorace, C. Haycock and R. Shillock (eds.), *Language Acquisition: Knowledge Representation and Processing*, Special issue of *Lingua*, 23–79.

Yergeau, F.; Bray, T.; Paoli, J.; Sperberg-McQueen, C. M.; Maler, E. (2004) *Extensible Markup Language (XML) 1.0 (Third Edition)*, W3C Recommendation, 4th February 2004. http://www.w3.org/TR/REC-xml/