Transcribing and annotating spoken language with EXMARaLDA

Thomas Schmidt

Sonderforschungsbereich 538 'Mehrsprachigkeit'
University of Hamburg, Max Brauer-Allee 60, D-22765 Hamburg
thomas.schmidt@uni-hamburg.de

Abstract

This paper describes EXMARaLDA, an XML-based framework for the construction, dissemination and analysis of corpora of spoken language transcriptions. Departing from a prototypical example of a "partitur" (musical score) transcription, the EXMARaLDA "single timeline, multiple tiers" data model and format is presented alongside with the EXMARaLDA Partitur-Editor, a tool for inputting and visualizing such data. This is followed by a discussion of the interaction of EXMARaLDA with other frameworks and tools that work with similar data models. Finally, this paper presents an extension of the "single timeline, multiple tiers" data model and describes its application within the EXMARaLDA system.

Background

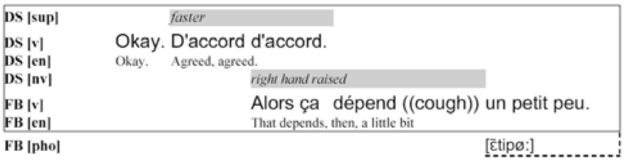
The EXtensible MARkup Language for Discourse Annotation (EXMARaLDA) is being developed at the 'SFB Mehrsprachigkeit' (Research Centre on Multilingualism) in Hamburg as the core architectural component of a database of multilingual spoken discourse. This database is intended as a platform for exchanging, archiving and analyzing the transcription data that the different SFB projects work with. The theoretical backgrounds and research goals of the projects differ greatly: they range from phonetic analyses of child language over studies of the acquisition of syntax in a generative framework to discourse analyses in a functional-pragmatic context. As a result of this diversity in research interests, the transcription systems, data formats and tools currently in use are also very dissimilar: for instance, one project works with a relational database of phonetically transcribed utterances whereas others use the syncWriter software (for a brief overview, see Bernsen et al., 2002) for creating orthographic multi-modal transcriptions in partitur notation.

This theoretical and technical diversity being an obvious obstacle in data exchange, the main challenge in the development of EXMARaLDA lies in the construction of a modeling framework that enables linguists to express their different models of spoken language on a common structural basis. Departing from such a data model, it should become possible to develop a set of interoperable data formats and tools that make the construction and exchange of richly annotated spoken language corpora easier.

A simple data model for multi-layered transcriptions

Partitur Transcriptions

Four of the fourteen projects at the SFB transcribe multiparty discourse according to the HIAT conventions (Ehlich, 1992). HIAT uses the so called partitur (musical score) notation in order to visualize temporal sequence and simultaneity between the utterances of different speakers, between different modalities (verbal and non-verbal behavior) and between segmental and non-segmental (prosodic) phenomena. As the following figure shows, further analytic information – like an utterance-based transcription and a phonetic annotation – can also be integrated into the partitur:



DS [sup]	*faster*
DS [v]	Okay. D'accord d'accord.
DS [en]	Okay. Agreed, agreed.
DS [nv]	*right hand raised*
FB [v]	Alors ça dépend ((cough)) un petit peu.
FB [en]	That depends, then, a little bit
FB [pho]	[ɛ̃tipø:]

Figure 1: A partitur transcription

Structural relations in a partitur

As figure 1 illustrates, partiturs can be used to visualize a number of structural relations between entities of spoken language: The subdivision of a partitur into several tiers reflects an assignment of entities to different *speakers* (*DS* and *FB* in the example) and to different annotation *categories*. The categories in turn can be grouped into three different *types*:

- The actual *transcription* of verbal behavior (*v*-tiers above) which is used as the temporal point of reference for all other entities, i.e. every other entity is related to the verbal material by aligning the corresponding symbolic descriptions with an appropriate position in the transcription tiers. This is only possible because every symbolic description in a transcription tier is segmentable into smaller units, and because the sequence of these units corresponds to a temporal ordering of the entities (words, word fragments, phonemes, etc.) they describe.
- Like the transcriptions, *descriptions* of non-verbal behavior (the *nv*-tier above) relate to events that are independent of events in other tiers. In contrast to transcriptions, however, descriptions are atomic units that cannot further subdivided.
- *Annotations* (the *sup*-, *en*- and *pho*-tiers above) describe additional features (prosody, translations etc.) of verbal behavior that are not captured in the transcription tiers. As they are thus always related to verbal material, annotations, unlike transcriptions and descriptions, are *not* independent entities.

Lastly, the relation of entities in different tiers of the partitur can be thought of as the reference to *a common timeline*: simultaneous events or entity/feature pairs are placed at the same horizontal position, and the left-to-right direction within a tier or across tiers corresponds to temporal sequence.

The following figure, which represents the structure of the example above, sketches the "single timeline, multiple tiers" data model that results from these considerations:

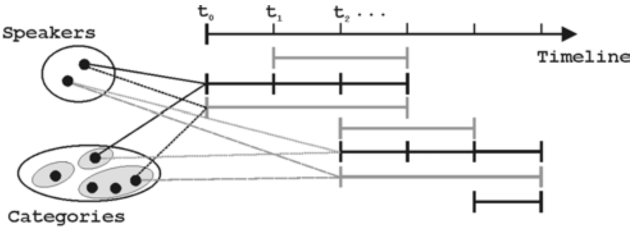


Figure 2: "Single timeline, multiple tiers" data model

In the EXMARaLDA system, data of this kind can be represented in an XML file that conforms to the *basic-transcription* document type definition:[1]

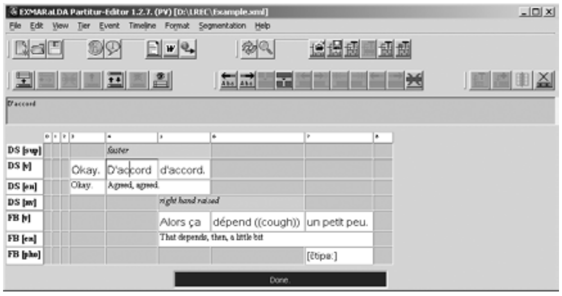
```
<!ELEMENT basic-transcription (head, basic-body)>
<!ELEMENT head (speakertable)>
<!ELEMENT speakertable (speaker*)>
<!ELEMENT speaker EMPTY>
<!ATTLIST speaker
    id ID #REQUIRED>
<!ELEMENT basic-body (common-timeline, tier*)>
<!ELEMENT common-timeline (tli*)>
<!ELEMENT tli EMPTY>
<!ATTLIST tli
    id ID #REQUIRED
    time CDATA #IMPLIED>
<!ELEMENT tier (event*)>
<!ATTLIST tier
    speaker IDREF #IMPLIED
    category CDATA #REQUIRED
    type (t | d | a ) #REQUIRED>
<!ELEMENT event (#PCDATA)>
<!ATTLIST event
    start IDREF #REQUIRED
    end IDREF #REQUIRED>
```

Figure 3: EXMARaLDA basic-transcription DTD

According to this DTD, the smallest entities of a partitur transcription are represented as *events*. Events refer to the items (*tli*) on a *common-timeline* via the *start* and *end* attributes and are grouped into *tiers* where each tier is assigned a *speaker*, a *category* and one of the three *types* described above. Additionally, each item of the timeline can be assigned an absolute time value by means of an optional *time* attribute and thus point to a position in the transcribed audio or video recording.

An editor for partitur transcriptions

For creating and editing *basic-transcriptions*, EXMARaLDA provides the Partitur-Editor[2], a tool written in Java that visualizes the data as a partitur and allows interactive editing of tiers (adding, deleting, reordering), events (adding, deleting, splitting, merging and a number of other specialized functions), the timeline and the speaker table. In contrast to most other transcription tools currently under development, the Partitur-Editor offers extensive support for the use of different font types, styles and sizes and thus enables the user to typographically distinguish different types of information:

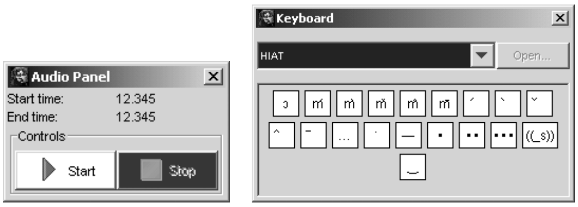


D'accord

	0	1	2	3	4	5	6	7	8
DS [sup]					faster				
DS [v]				Okay.	D'accord	d'accord.			
DS [en]				Okay.	Agreed, agreed.				
DS [nv]						right hand raised			
FB [v]						Alors ça	dépend ((cough))	un petit peu.	
FB [en]						That depends, then, a little bit			
FB [pho]								[ɛ̃tipa:]	

Figure 4: EXMARaLDA Partitur-Editor

Beside these essential editing functionalities, the Partitur-Editor also provides some basic support for audio playback given that the timeline items of the basic-transcriptions have been assigned absolute time values (see above). Furthermore, as a truly Unicode-enabled tool, the editor comprises a customizable virtual keyboard for input of symbols that are not available via the system keyboard:

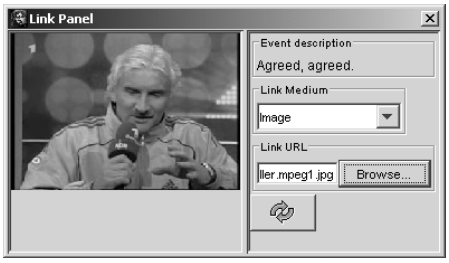


Start time: 12.345
End time: 12.345

HIAT

Figure 5: Audio playback panel and virtual keyboard

Especially for the analysis of multi-modal behavior, it is often desirable to link parts of the transcribed material to portions of the underlying recording or to image data. To this end, the Partitur-Editor contains a link panel in which single events can be associated with external audio, video or image files.



Event description: Agreed, agreed.
Link Medium: Image
Link URL: ller.mpeg1.jpg

Rendering partiturs on screen and paper

The kind of discourse analysis that uses HIAT as its transcription system (and, in fact, a great number of other linguistic methodologies) relies heavily on a qualitative interpretation of written transcripts. While being able to display a partitur on the screen may be sufficient for some purposes, the possibility of having a readable *printout* of the transcription remains a vital requirement from these users' point of view. Paradoxically, many transcription tools currently under development attach very little or no importance to that aspect, either because their focus is entirely on computer-based (and hence "screen-centered") analysis methods, or because of the alleged ease with

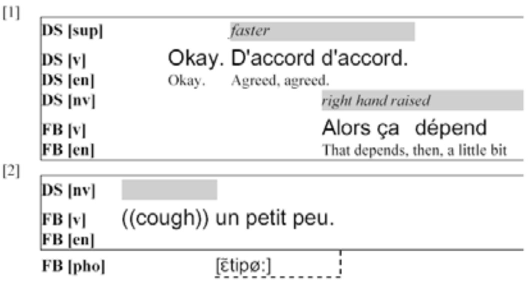
[1] For the sake of simplicity, some details of the DTD have been left out.

[2] The Partitur-Editor is freeware and can be downloaded from http://www.exmaralda.org.

which XML-encoded data can be transformed into presentation formats via XSL-Stylesheets.

However, the non-hierarchical nature of the "single timeline, multiple tiers" data model makes the use of stylesheet transformations a non-trivial matter, and the interlinear structures in a partitur are a notoriously difficult area for common rendering software like browsers and word processors, see (Bow et al., 2003)[3].

The EXMARaLDA system therefore provides an extensive functionality for transforming a *basic-transcription* into a printable form. It allows the user to parameterize the formatting properties (font types and styles, borders, numbering etc.) of a partitur and to specify page formats (size and margins) and, based on these parameters, calculates a line-wrapped version of a partitur that can then be output directly to a printer or imported into a word-processor as an RTF file or into a browser as an HTML file:



[1]

DS [sup]		*faster*	
DS [v]	Okay.	D'accord d'accord.	
DS [en]	Okay.	Agreed, agreed.	
DS [nv]			*right hand raised*
FB [v]			Alors ça dépend
FB [en]			That depends, then, a little bit

[2]

DS [nv]	
FB [v]	((cough)) un petit peu.
FB [en]	
FB [pho]	[ɛ̃tipø:]

Figure 6: A wrapped partitur

Exchange with other tools and formats

The "single timeline, multiple tiers" data model is not unique to the EXMARaLDA system. Among other tools or systems that work with a comparable data model are:

- the TASX-Annotator developed at the University of Bielefeld (see the contribution from Milde to this workshop),
- the Praat software developed by Paul Boersma (http://www.fon.hum.uva.nl/praat/),
- the EUDICO Linguistic Annotator (ELAN), developed at the Max-Planck-Institute for Psycholinguistics in Nijmegen (Brugmann, 2003).

Although the structures of these data models are not a hundred percent identical to that of an EXMARaLDA *basic-transcription*, they are sufficiently similar to make a fully automatic conversion in both directions possible. Such import and export filters are an integral part of the EXMARaLDA system, and they have proven especially valuable because the EXMARaLDA Partitur-Editor on the one hand and the TASX-Annotator, Praat and ELAN on the other hand address partly complementary needs: whereas the Partitur-Editor is superior to the other tools with respect to parameterizability of the visualization and output functionalities, it offers only minimal support for the interaction of digitized recordings with the transcription process. The TASX-Annotator, Praat and ELAN, on the other hand, provide precisely that kind of support, and an interoperability between the tools therefore has a great synergetic value from the users' point of view.

Thus, one project at the SFB uses Praat for a rough segmentation and transcription of the digitized audio recordings and then imports these data into the EXMARaLDA Partitur-Editor for a refinement of the transcription, an addition of analytical annotations and the print-out of transcripts (see Schmidt, 2003b). Similarly, other users make their primary transcriptions of video recordings in TASX or ELAN and then transfer these data to EXMARaLDA for further processing and output.

Legacy data and other data

The different SFB projects have large amounts[4] of legacy data which, in their original form, have very limited potential for exchange and reuse. One major part of the database project therefore consists in the conversion of these legacy data into the EXMARaLDA format.

On the one hand, this pertains to partitur transcriptions created with the software tools HIAT-DOS and syncWriter. As the data models of these tools are geared towards visual display rather than logical structure, a fully automated conversion is not possible. The corresponding conversion methods therefore map parts of the data structure to an EXMARaLDA *basic-transcription* and thus reduce the cost of manual post-editing as far as possible.

Many legacy data, on the other hand, have a much simpler structure than a partitur transcription: they have been created with simple text editors or as RDB-tables and follow the concept of a simple line-for-line transcription where each line contains exactly one utterance and temporal overlaps are marked with an appropriate bracketing:

DS: Okay.
DS: D'accord <d'accord.>1>
FB: <Alors >1> ça depend ((cough)) un petit peu.

Figure 7: A line-for-line transcription

These kind of data can be imported into EXMARaLDA via the "Simple EXMARaLDA" interface, an import filter that operates on plain text files and maps the structure of a line-for-line transcription onto the "single timeline, multiple tiers" data model. Conversion in this case is fully automatic, i.e. it requires no manual post-editing.

Beyond the single timeline

The "single timeline, multiple tiers" data model has proven to be useful because it is powerful enough to express a lot of structural relations in spoken language while at the same time being sufficiently simple and intuitive to form the basis of user-friendly and efficient implementations.

However, it is beyond doubt that the transcription and annotation of spoken language can lead to data structures that are not covered by this simple data model. Again, EXMARaLDA is not unique in acknowledging this limitation and recognizing the need for more powerful mechanisms: The approach taken by TASX is the so called "TASX level 2" data model where events can either refer to the common timeline or to events in other tiers thus

[3] What (Bow et al., 2003) discuss under the notion of "interlinear text" is conceptionally slightly different from my notion of a "partitur" (cf. Schmidt, 2003a). The difficulties in rendering, however, are very similar for both concepts.

[4] More than 1000 hours of transcribed spoken language, or over 2500 single transcriptions, as a rough estimate.

allowing the construction of hierarchical annotation structures (Milde/Gut, 2003). The EUDICO Abstract Corpus Model (Brugman, 2003) also goes beyond strictly time-based structures by allowing symbolic subdivisions and symbolic associations of entities in different tiers.

The EXMARaLDA approach is different from these approaches because it does not abandon the timeline metaphor altogether, but instead extends it to a more complex construction: a *segmented-transcription*. In contrast to an *basic-transcription*, the timeline of an EXMARaLDA *segmented-transcription* can have bifurcations. This is a need that arises as soon as a *temporally* structured transcription is segmented into *linguistic* units. For instance, in the above example, a segmentation of the verbal tiers into words will lead to a data structure in which the temporal relation between some words of different speakers cannot be determined:

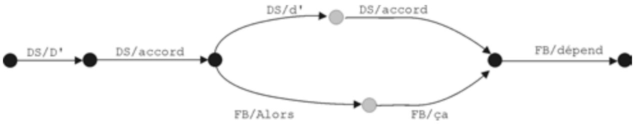


Figure 8: A bifurcated timeline

The segmentation of transcribed material into linguistic units being the most important prerequisite for most analytical processes (like additional annotation or search), the EXMARaLDA segmented-transcription thus provides an extension of the “single timeline, multiple tiers” data model that is crucial in obtaining truly computer-suitable representations of spoken language.

Segmenting with finite state machines

EXMARaLDA does not provide a tool for inputting and editing *segmented-transcriptions* directly. Instead, *segmented-transcriptions* are automatically generated from *basic-transcriptions* on the basis of the punctuation in the transcription tiers. This punctuation is interpreted as an implicit markup, i.e. as symbols marking the beginning and the end of linguistic units, and transformed into explicit XML-markup by means of a finite state machine (FSM):

Basic-Transcription		
Event	**Event**	**Event**
Okay.	D‘accord	d‘accord.

|

FSM for HIAT

↓

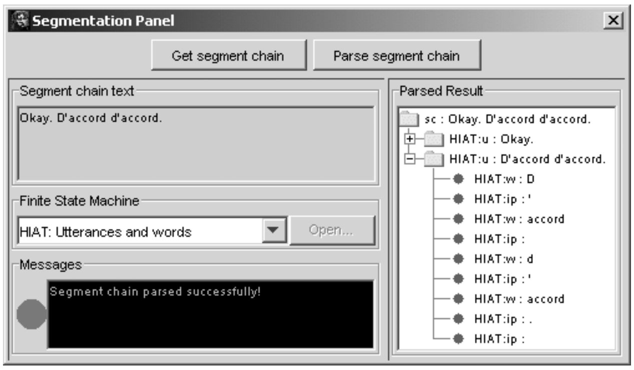
Segmented-Transcription									
Segment Chain									
Utterance		**Utterance**							
W	**P**	**W**	**P**	**W**	**P**	**W**	**P**	**W**	**P**
Okay	.	D	‘	accord		d	‘	accord	.

Figure 9: Segmentation

It is important to note that this process of segmenting transcriptions of spoken language is different from what a sentencizer or tokenizer does for texts of written language: as punctuation use in transcriptions is always done according to a specific transcription convention, the types and positions of punctuation symbols are totally predictable – for instance, there will be no ambiguity about whether a particular punctuation mark must be interpreted as an utterance terminator or as a word terminator.

As segmentation is thus dependent on the transcription system used, the algorithm must be parameterizable. This is achieved by using different finite state machines for different transcription systems. At the time of writing, three different FSMs – one for HIAT, one for DIDA (Klein/Schütte, 2001) and one for CHAT (MacWhinney, 2000) – are integrated into the EXMARaLDA system. As the FSMs are also formulated as XML files, this mechanism can be easily adapted or modified to meet the conventions of other transcription systems. Furthermore, encoding the segmentation algorithm as an XML file also ensures that it is largely independent of the rest of the software and could thus be readily integrated into other environments.

Besides being the basis for the transformation of a *basic-transcription* into a *segmented-transcription*, the finite state machines can also serve as a means for controlling the validity of transcriptions. A failure of the segmentation algorithm will tell the transcriber that somewhere in the transcription a certain symbol does not conform to the underlying conventions. In order to be able to easily identify such errors, the EXMARaLDA Partitur-Editor provides a segmentation panel that allows the user to go through the transcription step-by-step and find places where the segmentation algorithm runs into an problem:



Segment chain text: Okay. D'accord d'accord.

Finite State Machine: HIAT: Utterances and words

Messages: Segment chain parsed successfully!

Parsed Result

```
sc : Okay. D'accord d'accord.
  HIAT:u : Okay.
  HIAT:u : D'accord d'accord.
    HIAT:w : D
    HIAT:ip : '
    HIAT:w : accord
    HIAT:ip :
    HIAT:w : d
    HIAT:ip : '
    HIAT:w : accord
    HIAT:ip : .
    HIAT:ip :
```

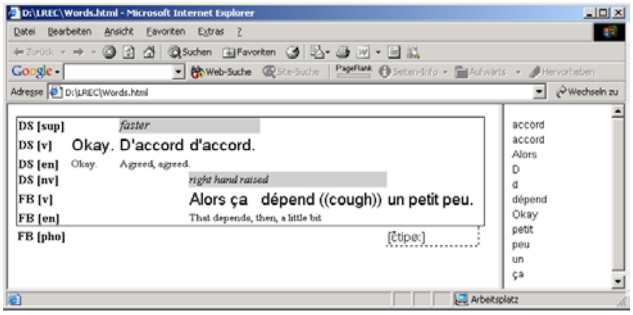
Figure 10: Segmentation Panel

Making use of segmentation

The *basic-transcription* data model is used for input and visualization of transcriptions in partitur notation. After a *basic-transcription* has been successfully transformed into a *segmented-transcription*, further processing methods become possible:

Building on the segmentation into utterances (or equivalent units), a visualization in a line-for-line notation as in figure 7 can be calculated. The same transformation can also be used as the basis for a conversion of time-based EXMARaLDA data into formats that follow a more hierarchically structured data model (e.g. the TEI format for the transcription of speech).

Similarly, the segmentation can be used for a calculation of alphabetic word lists. A much used feature of the Partitur-Editor is the option to output such word lists in HTML and link them to a HTML output of a partitur transcript thus enabling a quick word search in context:


D:\LREC\Words.html - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Favoriten Extras ?
Zurück Suchen Favoriten
Google Web-Suche Site-Suche PageRank Seiten-Info Aufwärts Hervorheben
Adresse D:\LREC\Words.html Wechseln zu
Arbeitsplatz

DS [sup] faster
DS [v] Okay. D'accord d'accord.
DS [en] Okay. Agreed, agreed.
DS [nv] right hand raised
FB [v] Alors ça dépend ((cough)) un petit peu.
FB [en] That depends, then, a little bit
FB [pho] [ɛ̃tipø:]

accord
accord
Alors
D
d
dépend
Okay
petit
peu
un
ça

Figure 11: Word list and linked transcript

Last but not least, a segmentation of transcriptions will be the prerequisite for any elaborate analysis method like detailed annotation, querying etc. EXMARaLDA does not yet provide a generic tool for performing such analyses, but first tests with a small corpus of *segmented-transcriptions* and a standard RDB-system (Schmidt, 2003b) have shown that the potential of EXMARaLDA data clearly exceeds the possibilities of older tools and formats in that respect.

Time-based data models and XML technology

The "single timeline, multiple tiers" data model is arguably the most simple and intuitive one for describing the kind of transcription data that discourse analysts and many other linguists work with. As shown above, its conceptual structure can straightforwardly be represented physically in an XML file, and the resulting corpora can thus profit from many of the benefits that XML as a wide-spread standard offers – the data become exchangeable between different tools and platforms, the full use of Unicode becomes a matter of course (an aspect that is of obvious relevance especially to multilingual data), and XML also lends itself to making some processing methods for transcription data (e.g. the segmentation by FSMs, see above) parameterizable in a software-independent way.

By and large, however, the role of XML in the EXMARaLDA system remains limited to that of a standardized storage format, and the full potential of XML technology can thus not be exploited. The reason for that is that most of XML technology is closely tied to a primarily hierarchical data model, whereas we do not see hierarchies as the primary structural relation in our kind of data. For the time-based data model(s) that result from this consideration, XML technology therefore does not always constitute the optimal framework, for instance:

- As DTDs and Schemata primarily serve the purpose of checking the well-formedness and validity of tree-structures, they will not be sufficient to describe and verify XML-encoded instances of the "single timeline, multiple tiers" data model. For instance, the DTD in figure 3 will not check whether the start and end points of a given event follow one another in the timeline or whether two events in one tier do not overlap.
- As XSL is mainly a language for transforming source trees into result trees, it is not well suited to calculate visualizations whose primary structure is not hierarchical. Partitur transcriptions are a case in point for such visualizations.
- As query languages like XQuery are also designed around a hierarchical data model – they are efficient in navigating and querying tree structures – their use for querying multi-layered data like those presented here is also questionable.

Two extreme conclusions could be drawn from this dilemma: One would be that time-based data models, since they cannot tap the full potential of XML technology, are not the most useful approach to the goal of constructing richly annotated language corpora. It is this view that underlies (Carletta et al., 2000)'s criticism of the (time-based) annotation graph formalism.[5] The other would be that XML, since its associated technologies do not adequately support the intuitive time-based data model, should not be considered a relevant factor in the construction of such corpora.

EXMARaLDA follows an approach that lies in-between these two extremes. On the one hand, it relies strongly on XML as a standardized storage format and, insofar as it structures time-aligned entities into a system of tiers, also partly accommodates the prototypical hierarchical XML data model.[6] On the other hand, it does not view XML technology as the paramount criterion that decides on the choice of data structures and processing methods for a spoken language corpus – because spoken language is very rich in non-hierarchical structures (at least according to the models that many transcription systems work with), prioritizing hierarchical relations over other relations would mean an artificial restriction hindering an efficient processing of transcription data rather than facilitating it.

A drawback resulting from the latter point is the lack of an industry-supported framework or API that would help developers in the construction of tools for input and analysis of time-based data in the same way that XML technology aids the processing of hierarchically structured data. In that respect, interoperability between existing tools and formats for time-based data becomes a very important requirement. The possibilities of data exchange between TASX, EXMARaLDA, Praat and ELAN, as described above, are already a major step in this direction. Further harmonizing the respective formats and, in particular, a common approach to an extension of the "single timeline, multiple tiers" data model would seem like a good next step.

Outlook

By the time of writing, EXMARaLDA can be said to have left the stage of a prototype system. The tools and formats are used in the every-day-work of linguists both inside and outside the SFB for research and teaching.[7] Beside maintenance and improvement of the existing tools, further work will focus on corpus management and corpus analysis. Two tools addressing these aspects are currently under development: One is the EXMARaLDA Corpus manager (Wörner, forthcoming), a tool which supports the creation

[5] "We propose that since most XML use privileges element hierarchies by making hierarchical structures easy and fast to navigate, element hierarchies should be used to represent the most important relations in an XML data set." (Carletta et al., 2000)

[6] In that respect, it is a less powerful but also an easier-to-handle data model than the more general annotation graph data model.

[7] Judging by download figures for the Partitur-Editor, the total number of EXMARaLDA users should be somewhere between 500 and 1000.

and management of corpus meta-data and the linking of this information to the actual transcriptions. The other is a concordance tool designed to help with the search and analysis of transcribed and annotated phenomena in an EXMARaLDA corpus.

The ongoing conversion of legacy data into the EXMARaLDA format and the use of EXMARaLDA tools for the creation of new data should meanwhile lead to a number of "real-life" sized multilingual corpora of spoken language that will allow an insight into possibilities for further development and optimization of the framework.

References

Bernsen, N. / Dybkjaer, L. / Kolodnytsky, M. (2002). An Interface for Annotating Natural Interactivity. In Kuppevelt, J. v. / R. W. Smith (eds.). Current and New Directions in Discourse and Dialogue. Dordrecht: Kluwer.

Bow, C./Hughes, B./Bird, S. (2003). Towards a General Model of Interlinear Text. In Proceedings of the E-Meld Workshop on Digitizing and Annotating Texts and Field Recordings. Lensing: LSA Institute, Michigan State University.

Brugman, Hennie (2003). Annotated Recordings and Texts in the DoBeS Project. In Proceedings of the E-Meld Workshop on Digitizing and Annotating Texts and Field Recordings. Lensing: LSA Institute, Michigan State University.

Carletta, J./Isard, A./McKelvie, D. (2000): Linguistic Data Processing For Everyman. In Proceedings of the Workshop on Web-Based Language Documentation and Description. Philadelphia: Institute for Research in Cognitive Science, University of Pennsylvania.

Ehlich, K. (1992). HIAT - a Transcription System for Discourse Data. In Edwards, J. / Lampert, M. (eds.). Talking Data – Transcription and Coding in Discourse Research. Hillsdale: Erlbaum, 123-148.

Klein, W. / Schütte, W. (2001): Transkriptionsrichtlinien für die Eingabe in DIDA. Mannheim: Institut für Deutsche Sprache (IDS).

MacWhinney, Brian (2000): The CHILDES project: tools for analyzing talk. Mahwah, NJ u.a. : Lawrence Erlbaum.

Milde, J.T./Gut, U. (2003): Multimodale bilinguale Korpora gesprochener Sprache: Korpuserstellung, -analyse und -dissemination in der TASX-Umgebung. In In Seewald-Heeg (ed.). Sprachtechnologie für die multilinguale Kommunikation - Textproduktion, Recherche, Übersetzung, Lokalisierung (Beiträge der GLDV-Frühjahrstagung 2003). Sankt Augustin: gardez!, 406-420.

Schmidt, T. (2003a). Visualising Linguistic Annotation as Interlinear Text. In Working Papers in Multilingualism, Series B (46). Hamburg.

Schmidt, T. (2003b). Korpus „Skandinavische Semikommunikation" - ein mehrsprachiges Diskurskorpus auf XML-Basis. In Seewald-Heeg (ed.). Sprachtechnologie für die multilinguale Kommunikation - Textproduktion, Recherche, Übersetzung, Lokalisierung (Beiträge der GLDV-Frühjahrstagung 2003). Sankt Augustin: gardez!, 421-427.

Wörner, K. (forthcoming): CoMa – A corpus manager for EXMARaLDA data. To appear in Working Papers in Multilingualism, Series B. Hamburg.