ELAN – aspects of interoperability and functionality

Han Sloetjes, Peter Wittenburg, Aarthy Somasundaram

Max Planck Institute for Psycholinguistics, The Language Archive, Nijmegen, The Netherlands

han.sloetjes@mpi.nl, peter.wittenburg@mpi.nl, aarthy.somasundaram@mpi.nl

Abstract

ELAN is a multimedia annotation tool that has been developed for roughly ten years now and is still being extended and improved in, on average, two or three major updates per year. This paper describes the current state of the application, the main areas of attention of the past few years and the plans for the near future. The emphasis will be on various interoperability issues: interoperability with other tools through file conversions, process based interoperability with other tools by means of commands send to or received from other applications, interoperability on the level of the data model and semantic interoperability.

Index Terms: multimodal annotation, interoperability, multimedia, cross-platform, corpus tool

1. Introduction

ELAN [1] is a generic multimedia annotation tool that is being applied in various areas of research. Although these research fields may have a lot in common in terms of tool requirements and applied methodologies, most of the fields have a few specific requirements as well. In almost any case ELAN is one of the tools applied and is used in combination with other tools. Which these other tools are depends on the respective field of research and they can differ considerably. The section devoted to interoperability through file exchange will delve further into this and provide some examples.

Annotation in ELAN is tier based, meaning that all annotations are added to a tier; a tier is a kind of layer and a container for annotations. Tiers can be part of a hierarchy of tiers and each tier is of a specific type. The type determines the constraints on the tier and on its annotations and the position it can take in a tier hierarchy. This design of the data model has some consequences for interoperability and the section on the ELAN data model will discuss some of them.

In its current form ELAN does not provide the means for other applications to interact with the program directly via commands or shared processes. Nevertheless, some more details on this type of interaction are described in the corresponding section.

ELAN has no built in support for any specific coding scheme but allows the user to design and implement her/his own scheme as a set of controlled vocabularies. These controlled vocabularies can either be part of the annotation document and as such under full control of the user, or be stored on e.g. the Internet, in which case it is more likely to be a closed vocabulary, under the control of a single responsible person or organization. In either case, when the user creates a template, the (links to the) vocabularies are part of it. A fairly new addition is the possibility to hook a document up with a lexicon that is available via a LEXUS [2] webservice. This makes the contents of an entire lexicon available to the annotator as a resource for looking up values. On the level of semantic interoperability, ELAN allows to link annotations and tiers to concepts defined in the ISOcat [3] data category registry, which can potentially make interpretation of the meaning or function of these elements more independent of the label used (i.e. of the contents of e.g. an annotation).

2. The main features of ELAN

Before going into the details of the different interoperability related aspects mention above, the main features and characteristics of the program are briefly described here. ELAN is available on Windows, MacOS and Linux and supports multi-camera recordings by allowing up to 4 videos in a single annotation document. It is also possible to annotate audio only recordings. ELAN is mainly written in the Java programming language, which should guarantee cross platform code and binaries, but for media play back it connects to one or more of the available "native" media frameworks. One of the advantages for the user of this approach is that many media file formats are supported (the down side is that problems with installed codecs are inherited as well). Where in the early years of ELAN only mpeg-1 files were "allowed" for video and wave files for audio, now any file type that is supported natively or by additionally installed codecs is accepted. In fact this has implications for interoperability as well: while the emphasis mostly is on the possibilities of exchanging the annotation data, often the converted data need to be seen in combination with the source of the data, the media files. Using "exotic" media file types potentially reduces the interchange success rate (and along these lines one could say that an application allowing the user to pick any type of file does the same). This is the common trade-off between convenience for the user and consequences for interoperability.

The ELAN documents are stored in an XML (Extensible Markup Language) [4] file with the extension .eaf (ELAN Annotation File). XML has rapidly become the standard for structured data and although using XML in itself doesn't guarantee interoperability, it is a huge improvement on e.g. the use of binary formats, less structured text formats or a database without proper export facility. One of the predecessors of ELAN, Media Tagger, stored its data in QuickTime track files and recovering the annotation data from those files has proven to be a painful endeavor.

3. File exchange, import and export

For any tool the most common way of being interoperable with another tool is by providing mechanisms for conversion of one format into the other, either in the form of import and export functions or by making available file transformers.

ELAN currently offers mainly import and export function to achieve interoperability with a number of tools that are important for (subgroups) of users. The most prominent ones are:

3.1. Toolbox/Shoebox

Toolbox [5] is an application that is being used, and has been used already for a very long time, by field linguists, amongst

Copyright © 2011 ISCA

others. The Toolbox transcriptions contain records of interlinear text, the layers of which are referred to as "markers", corresponding nicely to ELAN's tier concept. The relation types between markers, one-to-one and one-to-many, translate well to ELAN's implementations thereof. Toolbox doesn't have dedicated markers for storing time information to link records to segments of the media. To retain time alignment information ELAN writes and reads a few specialized markers (e.g. "\ELANBegin") on successive export and import actions. Toolbox transcriptions are stored in text files with extensions .txt, .sht or .tbt, in which the "vertical" alignment between tokens on related markers is realized by means of padding with whitespaces. This has proven to be difficult to validate and easy to disrupt (e.g. by a find-and-replace in a text editor). Recent versions of Toolbox have an export to XML format; the import and export functions of ELAN still have to be extended to support the .xml format as well.

3.2. FLEx, interlinear text format

FLEx (FieldWorks Language Explorer) [6] is an application for language documentation and analysis and is intended to be the successor of Toolbox. Its interlinear text output format (XML) can be imported into ELAN, but there are some issues to solve. As with Toolbox the FieldWorks data structure currently has no built-in elements for time alignment. One of the reasons for users to import the data into ELAN is to view and inspect the annotations in context with the source, i.e. aligned with the audio and or video fragments. For round-tripping purposes it is necessary that the alignment created in ELAN can be stored and retrieved in successive export and import actions. Once it has been sorted out how to solve this problem (either the FieldWorks data model is extended with elements for time alignment or the time alignment information has to be encoded in a "comment" or "note" element) ELAN will adapt its import function and add an export to FLEx function.

3.3. Praat TextGrid, PointTier and IntervalTier files

Many field linguists, phonologist and phoneticians use both Praat [7] and ELAN. Annotations created in Praat and saved as TextGrid files can be imported into ELAN, with the remark that Praat's PointTiers (single time events) are converted to annotations with a customizable duration (because ELAN does not support annotations without duration). Similarly ELAN tiers can be exported to TextGrid files, where possible hierarchical relations between tiers are flattened out.

Apart from exchanging annotation data it is also possible to link timeseries data created by Praat and stored as PitchTier or IntensityTier files. This type of data is just visualized in a dedicated viewer alongside the waveform and the annotations.

3.4. CLAN tool's CHAT format

In the CHILDES [8] system, which is being used for studying conversational interactions, the CLAN tools are developed. The annotation files produced by this system, CHAT .cha files, can be imported in and exported from ELAN. But these functions in ELAN are deprecated; users are advised to use the CLAN commands "chat2elan" and "elan2chat" instead. One of the incompatibilities between the two formats is that in ELAN annotations on depending tiers are always forced inside the time interval of the parent annotation, which is not always the case in CHAT.

3.5. Transcriber format

Transcriber [9] is a speaker turn oriented transcription tool which XML file format, .trs, can be converted to ELAN in a fairly straightforward fashion. ELAN offers the option to either create a single tier for all speakers (useful for recordings with many speakers) or create a tier for every single speaker. There is currently no export function, partly because many ELAN transcriptions are not easily transformed into speaker turns.

3.6. CSV, tab-delimited text files

Although not an annotation format, tab-delimited text is arguably the most important export format. It is the format for transferring data to spreadsheet applications (e.g. Excel), R or a database system in order to perform quantitative analysis. Complete transcription can be exported this way but this is also the dominant format for saving search results and overviews like annotation statistics.

For importing data tab-delimited files can be used as well, but since the table can consist any type of data the user has to configure the import process: which columns contain time information, which columns contain annotations or tier names?

Furthermore CSV is a common format for exchanging timeseries data. As such it is supported by ELAN as well.

3.7. Subtitle formats, presentation formats

A few formats are only available for export such as the subtitle formats, the interlinear text and "traditional transcript" formats. The list of subtitle formats contains SMIL, QT Text, SRT and STL.

3.8. Word list, lists of annotations

These options export a list of unique words or unique annotation values, possibly with an occurrence count.

3.9. LAF/GrAF

GrAF [10] is the XML serialization format of ISO's LAF (Linguistic Annotation Framework), which describes a graph model for stand-off annotations. The main elements are "nodes" and "edges", where nodes are placeholders for annotations that can reference segments of the primary data via "span" elements. The annotation content is modeled as a label and a feature structure. There are "set" elements for grouping nodes or edges. The GrAF format is explicitly positioned as a pivot format for exchanging annotations between tools/formats.

A first attempt to export an ELAN transcription to a pre-release or alpha version of the GrAF XML format has shown that it is possible to store all annotation information in the GrAF format. Moreover in many cases the annotation structures can be represented in several ways in GrAF. Although it is possible to group annotations (nodes) in sets, which is similar to the function of tiers in ELAN (and other similar tools), part of the information of tiers is easily lost. Properties like "participant" (speaker assignment) and the structural relations between tiers are difficult to retain and this makes round tripping lossier than wished for.

A check against the release candidate version of the GrAF format still has to be performed including import of (possibly a variety of) annotation structures in GrAF format.

3.10. Multiple file export

Some of the export functions are available for a collection of files (i.e. a local corpus) as well. The number of operations that can be performed on an entire (sub-) corpus will be increased step by step. Not only the import and export types of operations but others as well. The advantages for the users are manifold: improved speed, reduction of repetitive task, more convenience in getting the required results etc.

4. Process based interoperability, plugin mechanism

Another type of interoperability can be achieved when an application directly communicates with another application in order to let it perform certain tasks or process certain data. As the "calling" application in this scenario ELAN currently supports some interaction with Praat: open a selection of a file and show it in edit mode with a Spectrogram etc. or clip a selection of the file. Other applications that support execution from the command line can be called with parameters stored in a customizable configuration text file. The main purpose is for clipping the media file(s), a function that ELAN doesn't provide for itself.

ELAN is in its current state not available as a "receiver" application in this type of command driven interaction. It is not scriptable and does not listen for commands coming in. Internally the structure is there: all relevant actions are implemented according to the Command design pattern. What is needed is a listener service and publication of the syntax for the commands and their parameters. A command line application to send commands to ELAN and that would allow execution in a pipe/chain would be a valuable extra. It should be noted however that for the majority of users this would likely not be a feature that they would use often; it is our impression that most users will prefer operation via the user interface.

Although ELAN does not expose a general plugin interface it can be extended in various ways. A Service Provider Interface (SPI) had initially been specified for timeseries data handlers and metadata viewer components, both can now be installed as regular plugins in ELAN's extension folder. A plugin interface for audio and or video (or even text) recognizer components followed, allowing to add pattern recognition based semi-automatic annotation capabilities (applied in e.g. ELAN-Extra [11]). In the AVATecH [12] project a different extension mechanism has been worked out. This integration is done in a way that supports CLARIN [13] principles, whereby each recognizer is being described by CMDI (Component Metadata Infrastructure) [14] metadata, can be selected based on that information extracted from a registry and can be invoked with the help of the metadata file describing the application to call and which parameters it accepts.

A similar approach has been followed in the preliminary implementation of interaction with the lexicon tool LEXUS. Because the requirements for this interfacing have not been fully defined yet, the implementation will necessarily change over time. As long as LEXUS is only available as a web application the only way to interact with it is via its webservice, which is at the time of writing limited to perform queries and retrieve lexical entries that in ELAN can be applied to annotations. Adding entries and changing entries are planned future enhancements of the webservice.

5. Semantic interoperability

ELAN is agnostic as to the meaning of annotation labels and does not enforce or advocate any specific coding scheme. Annotation content is merely Unicode text. The user can set up a coding scheme by creating a set of controlled vocabularies and associate tiers with them, save the scheme as a template file, use it for all new transcriptions and share it with other users and other research teams. This improves consistency and comparability between files based on the same template. But since the choice of labels is completely free it remains difficult to compare with any other file. With that in mind the possibility to link annotations and tiers with a data category in the ISOcat data category registry has been added. Independent of the label of annotations, two annotations that are both linked to the same data category are semantically comparable by that virtue.

ISOcat is accessible for applications through a webservice and ELAN has a client for that service built-in. The functions provided by the service are adequate for the purpose of linking elements in ELAN with a public data category. It is not possible to login to ones own workspace and to use private domain categories that have not been authorized yet.

6. Data model, EAF file format

The ELAN data model is based on the Abstract Corpus Model [15] that has been extended when needed over the years. The basic setup remained the same: the main objects are TIER, ANNOTATION, LINGUISTIC_TYPE and TIME_SLOT. Many tier based annotation systems have in common that tiers act as containers for annotations that in turn may contain links to time intervals of the media. In most cases relations between tiers can be specified and/or grouping of tiers is supported [16]. The EAF is a fairly straightforward serialization of the objects in the data model. What stands out is the explicitness of the naming of the elements for storing time information, TIME_SLOT, where other tools or formats chose a more generic concept, e.g. "anchor". This explicit naming doesn't affect interoperability but it makes the use of EAF for other types of media than time-based media less likely. Although possible, it would be a bit awkward to use TIME_SLOT for storing e.g. indexes in a text file.

Another example of explicitness as compared to other systems concerns the tier typology. The relations that can exist between tiers, the constraints that apply to subordinate tiers are described quite precisely. This does have consequences for interoperability: while each tier in EAF can be easily converted to a generic type of tier (a collection of annotations) in most existing formats, this type information is liable to get lost on attempts to do round tripping between tools or formats.

7. Outlook

7.1. Functional perspectives

Annotations of all types on media streams are the basis for any linguistic theorization, however, their creation for content representing normal scenes is very time consuming as statistics within the DOBES project have shown: transcription 1:35, translation to a major language 1:25 and for deeper analysis factors higher than 1:100. This is the reason why we can speak of an annotation backlog, i.e. an increasing number of recordings are not available for research purposes, since they are not described in any way.

This is the reason why in the before mentioned AVATecH project a set of audio and video recognizers is being worked

out that offer the possibility to speed up the annotation process and foster linguistic theorization. We differentiate between first phase detectors that are currently being integrated into the ELAN framework and second phase, more complicated recognizers that will be added next. For audio processing the following recognizers are being integrated: silence/non-silence, speech/non-speech, speaker clustering, speaker diarization, intonation contour detection. For video the following recognizers are being integrated: shot and sub-shot detection and key frame extraction, skin color detection and hand/head movement detection.

Intensive discussions with researchers have shown that the integration of these recognizers is eagerly waited for, however, usability is important for acceptance. E.g. proper support for typical workflows, including manual corrections, should be provided for.

Thus ELAN is being extended into a tool that allows invoking typical audio/video recognizers, be it locally or by accessing a webservice. The integration requires more complex interaction in e.g. parameter selection and appropriate visualizations should be supplied.

Version 4.0 already supports a few recognizers; support for the others mentioned above will follow in future versions. By integrating more recognizers over time, we hope to overcome the annotation backlog problem, to speed up the tedious annotation process and to also help improve the quality of the annotations.

7.2. Interoperability perspectives

In the area of tool-to-tool file conversions several formats will be added or be improved. Until there is a common interchange format that renders comparable results as direct conversion from format A to format B, these kinds of specialized import/export functions will remain necessary.

In the area of process based tool interoperability at least the extension framework will be extended as well as the role of ELAN as a client of webservices. E.g. running a recognizer that is not available on the local machine but only can be accessed via a service. Eventually ELAN will also expose a service for other applications to send commands to.

Updates of the data model and the files format will occur every now and then, but modifications with the (sole) intention to improve interoperability are difficult to achieve. These kinds of modifications will almost certainly break backward compatibility. And since it is already difficult enough for most tools to find an audience and build a user base, the risk of losing users by breaking backward compatibility is not easily accepted.

Proper documentation of the data model and the file format is important especially for other tool developers but also for archive managers and at least some of the users of a tool. Even though the importance of documentation is fully recognized, it often is lacking or lagging behind. In practice the pressure on the development tasks easily take precedence over tasks that are of less significance to the average user. Reference documentation for the EAF file format and the way it is interpreted by ELAN will be made available in the first half of this year.

8. Conclusions

Overall a fair level of interoperability with other tools and formats has been achieved. ELAN has found a place in the workflow of a variety of research groups in various disciplines. All import and export file formats have been added because of a demand for it.

Nevertheless, ample room for improvements remains, in all areas discussed as well as in providing up-to-date reference documentation.

9. References

[1] http://www.lat-mpi.eu/tools/elan/
[2] http://www.lat-mpi.eu/tools/lexus/
[3] http://www.isocat.org/
[4] http://www.w3.org/standards/xml/
[5] http://www.sil.org/computing/toolbox/
[6] http://fieldworks.sil.org/flex/
[7] http://www.fon.hum.uva.nl/praat/
[8] http://childes.psy.cmu.edu/
[9] http://trans.sourceforge.net/
[10] Ide, N., and Suderman, K. (2007). GrAF: A Graph-based Format for Linguistic Annotations. Proc. Linguistic Annotation Workshop, pp. 1–8.
[11] Leitner, C., Schickbichler, M., and Petrik, S., "Example-Based Automatic Phonetic Transcription", Proceedings of the Seventh conference on International Language Resources and Evaluation, LREC 2010
[12] http://www.mpi.nl/avatech/
[13] http://www.clarin.eu/
[14] http://www.clarin.eu/toolkit
[15] H. Brugman and P. Wittenburg. 2001. The application of annotation models for the construction of databases and tools. IRCS Workshop on Linguistic Databases, University of Pennsylvania. Philadelphia.
[16] Schmidt, T., Duncan S., Ehmer, O., Hoyt, J., Kipp, M., Loehr, D., et al. (2008). An exchange format for multimodal annotations. Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC-08).