Annotation Formats

This guide describes a number of annotation formats that are in use, the problems that are associated with them and where users have to take care of.

General Framework:

- The creation of annotations is one of the most important tasks of linguists when documenting a language or processing a text document. Annotations come in many different formats and encodings which makes it often very difficult to access them and to integrate them in interoperability scenarios.
- Therefore it is seen as important to discuss the properties of a number of well-known formats to make users aware of the possible choices.

1. General Remarks

Annotations can come along in many different flavors. They can be simple comments or transcriptions to textual or media sequences, but they can also amount to complex multi-tiered annotation structures including hierarchies, cross-references and different types of linking to time and to symbols. When creating or manipulating an annotation structure exclusively with one specific tool, the only question that used to be relevant for linguists was whether the tool suited the needs. This traditional view changes since people now want to re-use annotations, compare them for different languages, let them interact with lexica, combine them with other type of documents and maintain accessibility for many years. These requirements make it necessary to store the annotation contents in open formats and to take care that the encodings are standardized or documented. Traditionally the formats of annotation structures were not relevant to the users – only the tools were evaluated. This view changed completely, since continuously new tools with more functionality will become available. The format is now the essential anchor for capturing all the work of the linguists. Therefore, linguists should be aware of the format issue.

The table below may give an impression about some of the major formats that are used.
It indicates some relevant criteria that users should be aware of when deciding about a format: (1) Is it a widely accepted standard or best practice; (2) Does it supported structuring, hierarchies and cross-references; (3) Is it flexible to adapt to the user needs and does it support changes; (4) Does it allow to check for validity; (5) Does it support an open and human readable format; (6) Does it support UNICODE; (7) Is it supported by good tools and will it be supported in future by good tools;

Type	Major Properties
Shoebox	plain text; human readable; nicely structured by tags; flexible structure; hierarchies and therefore interlinearization is possible; no validation possible; arbitrary character encoding; utterances are the bases; no media linking;
Toolbox	same as Shoebox; but support for UNICODE and XML;
CHAT	plain text; human readable; flat structure; utterances are the basis, no hierarchies and no interlinearization; limited expressional power; special symbols used to encode properties; no explicit validation; media linking possible; UNCIODE support;
WORD	document format is proprietary and not human readable, no tag structure, no consistency control; mixture of visualization and representation aspects; RTF export possible but not easy to process; XML output is of limited use; no media linking;
EXCEL	flat table structure, field labels can be seen as tags; only simple annotation structures; xls format is proprietary; mixture of visualization and representation aspects;
Relational Databases	requires a logical design for related tables; flexible structure and hierarchies possible; validation of content by database mechanism; content can be transferred to XML in different ways; mostly no schema after XML output – no validation; character encoding often unclear after export; no reference concept; often fixed setups are defined;
XML	plain text; human readable; standard for structuring documents; requires UNICODE as character encoding; hierarchies are possible; interlinearization can be expressed; to define a specific format an XML schema is necessary, with schema a validation is possible; a mechanism for expressing typed references is available;
EAF	a XML schema based flexible format that can represent even complex annotation structures; supports time linking and linking to symbols; coherent with the concept of annotation graphs; UNICODE support;
LAF	Linguistic Annotation Framework; proposal for a generic annotation framework based on XML; not yet implemented

3. Résumé and Recommendations

Based on what was said we can make the following recommendations:

- Only tools that support UNICODE should be chosen, since character encoding problems are in general difficult to solve, i.e. the conversion is time and money consuming.
- Only tools that generate clearly structured annotation documents with explicit tags and suitable time linking should be used, in particular if they support constraints and controlled vocabularies.
- XML schema based formats are optimal in many dimensions, since they allow others to parse the structure, to build tools, to check validity.
- Even more optimal are such XML formats that are based on abstract annotation models such as EAF.

In this sense and due to its flexibility and useful linguistic functionality, Shoebox is still one of the best programs around. In particularly, the new version Toolbox that supports UNICODE and XML is an excellent tool.

A number of tools for annotating media streams were produced in the last years based on the ideas of abstract annotation models from MPI and the Annotation Graph model from Bird and Liberman such as ELAN, TASX. They all support UNICODE and schema-based XML and can therefore be recommended. Transcriber is an excellent tool for efficient speech transcription and it generates XML output. However, it does not support other annotation types and its conventions are not inline with the widely used models which may create conversion problems.

Tools based on relational databases have their great advantage in the easiness with which users can create complex table structures and simple user interfaces. However, they encapsulate all data, i.e. if the tool changes or if it is not available anymore it will be difficult to extract the content. Also it is the experience in many cases that it is not possible to generate correct and suitable XML formats from the database contents, i.e. again expensive conversion is necessary. There are many annotations created with the help of relational database systems and it is likely that many of them will not be accessible anymore after some time. So we cannot recommend to use such tools for all tasks.

This document is not intended to give a comprehensive overview about tools.