Lexical Structures

Peter Wittenburg
Version 8.8.01

This paper emerged from the documents we got until now and a number of discussions within and outside of DOBES. We are sure that there are still some errors in the description of the various concrete lexica and in the description of the wishes of the DOBES teams. We ask everyone for comments.

Notation

Here it is briefly described what kind of conventions are used in the formal structure descriptions in this document (except for the first part of chapter 2.1.5).

[|]	alternatives
{ }	bundling of elements, optional
*	one or several occurrences
< >	non-literal element
" "	literal element

1. Definitions

1.1 Goal

The goal is to come to an Abstract Lexicon Model (ALM). ALM is a generic definition of lexical object classes[1], their characteristics, and their relation to each other. A typical lexicon will include a subset of those object classes and many instantiations[2] of them. The relations can exist between

- Instantiations of the same class
- Instantiations of different classes

There are cases in which elements of characteristics (e.g. words in comments) have relations to either objects or elements of other objects.

We must allow for relations themselves to have complex characteristics. Classes can possibly inherit characteristics from others.

1.2 Power of ALM

ALM has to be so powerful that it can represent all current lexical formats such as CELEX[3] and even allow to represent the class of lexica which can be implemented by a tool such as Shoebox.

1.3 Lexical Operations

It may be possible that lexical classes are associated with operations which are triggered by certain events. These operations can include such actions as to generate instances of certain objects, fill characteristics of certain instances or respond to outside requests. Concrete, it may be the case that rules are added to a lexical class and that when an entry or an attribute is filled the rules automatically determine the value of another attribute which is added as well.

1.4 Lexical Formats

Lexica have to be transferred into persistent formats for storing, retrieval, and exchange purposes. In general these formats will be file formats, but still in many cases printable formats are relevant. The latter will not be discussed here due to the inherent limitations of the paper medium and due to the fact that transformations into printable formats should be easily made possible.

There exists a large variety of legacy formats such as CELEX as a table structure in a relational database, CELEX as a set of structured files, Shoebox lexica in the typical Shoebox MDF format (grouped feature-value pairs with many predefined feature types), Spreadsheet tables and many others.

Recently XML based formats were suggested. We can distinguish at least two approaches:

- A specific DTD/Schema is defined and used to implement exactly one special type of lexicon
- A more abstract XML representation (see for example Ide/Romary) is used by including predefined generic Data Categories (DatCat).

---

[1] Lexical object classes are the (complex) building blocks of a computational lexicon. They represent the relevant linguistic concepts, their attributes, and the methods with which they can be accessed.

[2] While an lexical object class may represent for example the linguistic concept “word sense” it is clear that a concrete lexicon has many of “word sense entries”. They are called instantiations of the class.

[3] CELEX is one of the first big computer lexicon projects. It was carried out at the Max-Planck-Institute in Nijmegen and includes lexica for the languages Dutch, English and German. Information is available at the web-site www.kun.nl/celex.

These DatCats can even be taken form openly accessible repositories and integrated into concrete environments with the help of RDF. Due to the limitations of the underlying data model of XML an ALM cannot be represented equivalently in XML-based structures.

2. Existing Lexical Representations

2.1 DOBES Lexica

The following examples are taken from what we know is currently existing or wanted by the teams within the DOBES project. Some of the various structures discussed below show several details that seem to involve mainly the organization for printout, such as the literal punctuation devices. The printout should be strictly separated from the logical structure of lexical databases, although the database design has to be such that various uses of it are possible. Especially in DOBES a printout usage is relevant. In the final sections of this paper we will concentrate on the logical structure and deliberately neglect the printout aspect.

2.1.1 Aslep Project

Three table structures are used at the moment, mostly as Spreadsheet in Excel. The addition of semantic relations is required.

<lexical entry> := <orthography> <German transl> <Russian transl> <Xakas transl>
<orthography> := <string> [“[“<string>”]” | “(“<string>”)”]
<string> := <substring> [{“.”<string>} | “-“]
<German transl> := <string> [{ <substring>} {“;”} | {“;” <substring>}]
<Russian transl> := <string> [{ <substring>} {“;”} | {“;” <substring>}]
<Xakas transl> := <string> [{ <substring>} {“;”} | {“;” <substring>}]
<substring> := <sequence of characters>

Structurally the other two lexica don’t differ, however, they contain different attributes (s. 4.1). The meaning of the punctuation characters (“;”,”.”,”-“) has to be sorted out (perhaps only relevant for printout?). It seems that they are not consistently used.

2.1.2 Monguor/Salar Project

This project has three different types of lexica.

The first lexicon is a table with one nesting dimension and is implemented as Word file.

<lexical entry> := <semantic code> <English gloss> {<dialect variant> (<dialect abbrev>, <phonetic code>)}*
<semantic code> := <complex code covering various information>
others are sequences of characters

The second lexicon is also implemented as Word file. Dependent on the grammatical category there is a substructur. The type face of the headword bears some meaning.

<lexical entry> := [<italic headword> | <bold headword>] <grammatic cat> <Mongolic> <Tibetan> {<senses>}*
<Mongolic> := <phonetics> <Mongolic ortho>
<Tibetan> := <phonetics> <Tibetan ortho>
<senses> := <number> “.” <English gloss> “:” {<sub>}*
<sub> := <phonetics> <English gloss>
<grammatic cat> := [cat | {“v” <retex> <headword> <subcat> <English gloss>}]

The third lexicon is implemented as a table structure in Filemaker. The table structure is not fully clear to us yet. It has many fields from which we assume that some are located in separate tables and it has one nesting dimension.

<lexical entry> := <source nr> <locale> <token nr> <token> <lemma nr> <lemma>
<engl.label> <til> <etymology> <comments> <lin96 token>
<Chinese sense> <pos> <ntypes> <voice> <valency>,
<arguments> {<samples>}*
<samples> := <text nr> <utt nr> <sample ID> <engl. Utt sense> <Chinese utt sense>
<speaker nr>

2.1.3 Wichita Project

The IDD design offers a rich structure covering many related tables. It is implemented in the Foxpro relational DBMS. The Wichita lexicon makes use of the structural capabilities of IDD but does not use all its possibilities/features and instantly "misuses" the fields for a different purpose.

Entry Table

<lexical entry> := <entry form> <variant form> <phonetic form> <miscellaneous> <gram cat>,
<entry nr> <sp> <complete> <label> <status> <use period>
<gender> <Slang> <prerogative> <ritual>

Wichita entries are mostly verbs, but also some nouns are included. Verbs are entered in their "abstract form", i.e. this seems to be something like a citation form.

Gloss Table

<gloss entry> := <entry nr> <entry form> <gram cat> <sp> <gloss nr> <gloss> {; <gloss>}*
<index> {; <index>}* <usage> <gender> <slang> <pejorative> <ritual>
<example nr>

Example Table

<example entry> := <entry nr> <entry form> <gram cat> <gloss> <example nr> <example>
<phonetic form> <morphemic constituency> <literal translation> <free translation>,
<source>

<example> contains the example phrases or sentences, <phonetic form> can contain a more detailed phonetic transcription of that example. <morphemic constituency> contains the morpheme breakdown forthe example. The attribute <literal translation> is used for the appropriate English and linguistic glosses for the grammaticalized elements within the <morphemic constituency> attribute. The <free translation> attribute is used to enter a "free" English gloss of the example.

Paradigmatic Form Table

<para entry> := <entry nr> <entry form> <gram cat> <para form nr> <para type> <para form>
<para analysis> <para gloss> <para source> <comments> <comment SP> <SP>

Will be used to illustrate the tense/mood paradigms of verbs.

Grammatical Form Table
Dialect Cross Refs Table

Both tables will be used within the Wichita project - not yet clear in which way.

Sound Table

<sound entry> := <entry nr> <entry form> <gram cat> <gloss> <parent> <parent name>
<sound word> <sound filename> <sound nr> <speech part> <source>

The purpose of the entries of this table are not yet fully clear.

The following tables are supported by IDD, but their usage for the Wichita project is not yet clear.

Historical Citations Table	contains historical examples
Derivations Table	contains analyzed forms
Language Cross Refs Table	contains cross-language cognates
Internal Cross Refs Table	contains dictionary-internal cross-refs
Semantic Class Table	intended for subject-indexing, the list of semantic topics is stored in an administrative DB
Idioms Table	contains idioms
Phrasal Table	contains phrasal examples
English Index Table	headword table with English terms, linked to the gloss table to link terms to senses
Images Table	
Videos Table	
Sortorder Table	contains definitions of sort orders

A relevant point was discussed by email communication. There have to be several sort forms for lexical entries for the researchers used to western languages and the indigenous people. This has to do with the fact that each complex word is prefixed in Wichita due to some grammatical category. Sorting in IDD delivers a grouping into the few classes which is very important for the indigenous to find entries. For the researcher a sorting without the prefix would be much more intuitive.

2.1.4 Teop Project

The Teop team is looking to finally establish one lexicon the entries of which may have a complex structure. The structure of the Final Lexicon is currently described implicitly by a Word-based lexicon. Currently, also a Shoebox dictionary is used for basically two reasons: (1) Find forms occuring in the corpora which are not yet in the Final Lexicon. The comparison is done manually. (2) Use the Shoebox lexicon for interlinearization purposes within Shoebox. The third lexicon type which is in use exists from a number of files each of which is grouped along semantic categories (here I refer to Ulrike's lexicon paper).

Of course, the Teop team would like to integrate the three lexicon structures. This is already done by integrating Shoebox entries manually into the Final Lexicon and by adding more lexical information. But also the dictionaries constructed around semantic categories such as kinship terms, plant names etc should be integrated into the Final Lexicon. The Final Lexicon is formatted such that it can be printed and handed over to the indigenous community as a "document which is easy to use". Creating a lexicon such as for Teop is more complex from many reasons. One reason is that there is no standardized orthography for Teop.

In the following the structure of the three lexicon types are formally described:

Final Lexicon (currently as Word document)

<lexical entry> := <stem orthography> [<sense> | {<sense nr> <sense>}*]
{<lexical entry>}*
<sense nr> := <digit>
<sense> := <gram cat>"."{<gram subcat>"."} <Engl transl>"." {<example>}*
<example> := <orthography>"."<Engl transl> {"["["T"|"pr"] <nr>"]"}

This lexicon can have several "run-on-terms" under headword entries which are normally morphologically related such as "gaga - gaagaga - gagaagaga - vaagaga". Between the entries and these subentries there is no structural difference, however the Teop people like to find these subentries under the stem variant. The lexicon has empty lexical entries for the subentries in so far that they simply include a reference under which headword the explanations can be found. It is clear that this could be represented internally differently, but for the presentation it is optimaly the way the Teop team does it.

The Shoebox lexicon can be seen as temporary, nevertheless its structure is given:

<lexical entry> := <stem orthography><hom nr> <gram cat>“.”{<gram subcat>”.”}
[{<sense> ”Ref:” <refnr>}| {<sense nr> <sense> ”Ref:” <refnr>}*]

The entries seem to be simple, but slightly modified versions of the Final Lexicon entries.

The third type of lexicon is also seen as a temporary one: It consists of semantically-related headwords respectively arranged in lists according to different themes (i.e. kinship terms, plants, fish). The following structure could be identified:

<lexical entry> := <headword>{“-“}{<hom nr>}”,”<gram cat>”.”<gram subcat>”.”
[<sense> | {<sense nr> <sense>”;”}*]
<sense> := <engl transl>{“(“<orthography>”)”}

It was briefly discussed what kind of scheme would be preferential when combining the various thematic lexica to one unified lexicon. Semantic class membership could be represented by an additional attribute containing the class identifier or by cross-references. The latter is the more flexible instrument since it would allow to also encode further content. Since encoding thematic topics is sufficient the first solution would be feasible and is more simple.

The following principle issues were raised in discussions:

1) Homophonous words need separate lemmas distinguished by numbers and separate entries. Here the sense information requires a separation although all share the same sound sequence.
   Kahi 1 (adverb) “away from”
   Kahi 2 (tense particle) indicating future
   Kahi 3 (noun) dog
2) Polysemous words (words having several meanings) need to have separate subentries under the same lemma.
   Kakaamuru 1. (noun) extremely white sand
   2. (adjective) extremely white

2.1.5 Aweti Project

This project has currently mainly a preliminary shoebox-database, in an elaborated version of the MDF-setup. There are quite detailed ideas about what requirements a lexicon component should meet in order to be combinable with a suitable glossing format.

In the Aweti-Project the lexicon (as a component of the language, not its representation in a lexicographic work such as a database or dictionary) of is seen as a structure having two parts: (1) lexical words (with a syntactical nature) and (2) lexemes, that is, analogous lexical units in morphology (not only stems, but also affixes). The syntactical lexicon also includes compounds and derived words.

The following proposal is indirectly relevant for the structure of the lexicographical database (for this issue see below). Instead, it is meant as to provide one possible terminological system that is consistent with the traditional lexicographic practice. It is different from the structural proposal for the lexicographic account of the language component, but it can serve as a frame of reference on order to determine whereto a certain lexicographic information refers (see the diagram below).

The following conventions are used (these conventions are not used for other parts in this documents):

((	))	optional	[	]	sequence
{	}	set	(|	)	alternative
<	>	pair, tuple	x/y@		definition applies to x and y

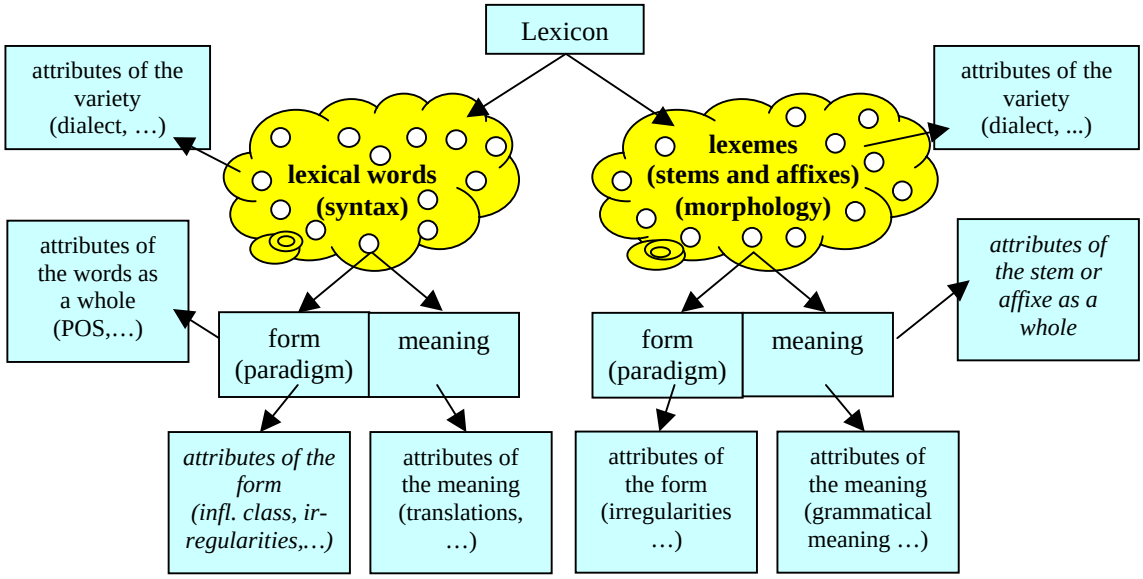
1 lexicon = <word lexicon, morphological lexicon>

2a word lexicon = {lexical words}
3a lexical word = <word paradigm, lexical meaning>
4a word paradigm = {w.form-categorisation-pairs}
5a wordform-categorisation-pairs = <wordform, syntactic categorization>
6a wordform = [phonological words]
7a syntactic categorization = {syntactic categories}
8a syntactic categories : “endpoints of the syntactic ordering - a classification system”
9 lexical meaning = (a content concept | the empty concept)

2b morphological lexicon = {lexemes}
3b1 lexeme = (stem | affix)
3b2 affix = (derivational affix | inflectional affix)
3b3a d.affix/i.affix @ <morphological paradigm, the empty concept>
3b3b stem = <morphological paradigm, lexical meaning>
4b morphological paradigm = {lexeme-form-categorization-pairs}
5b lexeme-form-categorization pair = <lexeme-form, morphological categorization>
6b lexeme-form = [morphs]
7b morphological categorization = {morphological categories}
8b morphological categories : “endpoints of the morphological unit ordering - a classification system”

10 phonological word/morphs @ <phoneme sequence, phonol.-const.-structure, phonol.-intonation structure>
11 phoneme sequence = [phonemes]
12 phonol const. structure : “provides the syllable breaking etc”
13 phonol intonation structure : “includes tones and word accents”

The numbers indicate an ontological analogy between the two branches.



The lexicographic lexicon (an account of the lexicon as a part of the language) is seen as set of lexicographic entries, each being a description of one or several lexical units (lexical words or lexemes). Each entry is a kind of cascaded structure (quite compatible with the MDF-proposal for Shoebox) where the global partitioning is done via principle differences in meaning, i.e. for every principle meaning difference there is a different entry although the citation form is identical (see the above discussion of homonymy, sec 2.1.4). At the next level the POS categorisation or other grammatical properties is used to make a difference. At the next nesting level one could have polysemy (different but related senses, se

above, giving rise to, for example, different translations). It is not completely clear how such a scheme could be adopted to highly polysynthetic languages such as Wichita.

The following concrete lexicon structure was worked out ((x==y) stands here for a logical test; from here on , the usual conventions of this document are used, cf. page 1; substructures are underlined to make reading more simple):

<lexical entry> := [{(<entry-type>==[“stem”|”idiom”|”lexical word”]) <head>
{<outer-body-L>}*} |
{(<entry-type>==[“auxiliary”|”inflect affix”]) <head> {<outer-body-I>}*} |
{(<entry-type>==”derivat affix”) <head> {<outer-body-D>}*} |
{(<entry-type>==”forms”) <head> {<outer-body-F>}*}]
<head> := <linguist-headword> {<printing-headword>} {<homograph number>}
{<phonetics/phonology>} <entry type>
<outer-body-L> := <grammar> <inner-body-L>
<outer-body-I> := <grammar> <inner-body-I>
<outer-body-D> := <grammar> <inner-body-D>
<outer-body-F> := <grammar-f> <main-ref> {<comment>}
<inner-body-L> := {<sense-number>} {<variety>} {<etymology>} {<government>}<meaning>
{<example>}* {<table>} {<comment>}* {<picture>} {<housekeeping>}*
<inner-body-I> := {<sense-number>} {<variety>} {<etymology>} <categorial effect>
{<meaning effect >} {<example>}* {<table>} {<comment>}*
{<housekeeping>}*
<inner-body-D> := {<sense-number>} {<variety>} {<etymology>} <structural effect>
<meaning effect> {<example>}* {<table>} {<comment>}*
{<housekeeping>}*
<grammar> := <POS> {<POS subcat>} {<paradigm-prop>} {<morphology>}
<meaning> := {<gloss>} {<word-level-gloss>} {<reversal>} <definition>
{<encyclopedic info>}
{<scientific name>} {<semantic domain>} {<index of semantics>}
{<thesaurus>} {<semantic relation>}* {<cross-ref>}*
<categorial effect> := {<condition>} <to-form-category>
<meaning effect> := [<encyclopedic info> | <definition>] {<literally>} {<semantic domain>}
{<index-of-semantics>}
<structural effect> := {<from-lexeme-category>} {<to-lexeme-category>}
{<from-form-category>} {<to-form-categroy>}
<grammar-f> := <form-categories> ??
<main-ref> := <main-entry-cross-ref> {<sub-entry number>}
<comment> := {<anthropo notes>}* {<discourse-notes>}* {<grammar-notes>}*
{<phonology-notes>}* {<question-notes>}* {<sociolinguistic notes>}*
{<general notes>}*
<variety> := [<usage> | <only-restrictions>]
<etymology> := {<borrowed-word-loan>} {<source-form>} {<ety-proto-form>} {<ety-gloss-E>}
{<ety-source>} {<ety-comment>}
<table> := ???
<example> := {<reference>} {<example-v>} {<example-morphology>}
{<explanation-of-form>} {<explanation-of-meaning>} {<example-free-trans>}
<example-morphology> := {<example-morph-by-morph>} {<example-literally>}
<housekeeping> := {<bibliography>} {<date>} {<status>} {<source>}* {<ety-comment>}
<phonetics/phonology> := {<phonetic form>} {<syllables>} {<lexical-tones>}
{<word-accent-or-pitch>}
<government> := {<valency>} {<government category>} {<compl-spec>}*
<paradigm-prop> := {<paradigm>} {<stem-type>} {<deficiencies>} {<irregularities>}
<morphology> := {<morph-by-morph>} {<literally>}
<semantic relation> := <lexical function> <lexical function lexeme> {<lexical function gloss>}
<cross-ref> := <cross-reference> {<cross-ref gloss>}

<compl-spec> := {<government form>} {<government-ont>}
<irregularities> := {<general-irregularities>} {<morphophonemics>} {<special form>}*
{<variant>}*
<special form> := <pardigm-form> <paradigm label> {<paradigm form gloss>}
<variant> := <variant form> {<variant comment>}

2.1.5 Trumai Project

The Trumai project's lexicon is realized with Shoebox and a structure was specified that uses grouping and nesting. The following information is contained: orthographic transcript, phonetic transcript (could be several dependent on observed pronunciations), grammatical cat (and subcategories for verbs and nouns), glosses in English and Portuguese, examples with translations in English and Portuguese, morphemic decomposition, citation form (for body parts and kinship terms), date of creation/modification. In the future media information and native definitions should be included. Also links to texts such as a sketch grammar would be useful.

<lexical entry> := <headword> <phonetic transcript> {<sense>}* {<citation form> <E-trans>
<Port-trans>} <definition> <kinship term> <morph decomp> <date>
<sense> := <POS> <E-gloss> <P-gloss> > {<example>}*
<example> := <orthographic trans> <E-transl> <P-transl>

A special solution from Shoebox is used for compounds. In case of a compound as headword the components are connected by "underscore". This tells Shoebox that the whole headword has to be treated as one unit. For interlinearization purposes in Shoebox this is relevant.

2.1.6 Kuikuro Project

The Kuikuro project maintains two types of lexical databases: (1) contains inflected nominal and verbal words, adverbs, postpositions and particles and (2) contains lexical forms encountered somewhere in the corpus. Finally, the fusion of both lexica to one is intended. This will include the following information:

<lexical entry> := <headword> <citation form> <phonetic trans> <POS> <E-gloss> <P-gloss>
<Kuikuro-def> <E-def> <P-def> <scientific name> {<example>}* <paradigm>
<semantic domain> {<comment>}* {<cross-ref>}*
<example> := <Kuikuro trans> <E-transl> <P-transl>

It seems that different type of comments and cross-refs are intended to be included. In that case the cross-refs have to be associated with a label denoting its type.

2.2 GenelexModel

The Genelex concept was derived from the lexical work in the PAROLE and SIMPLE projects. Genelex stands for "Generic Lexicon" and is mainly designed for automatic processing support. Genelex separates in levels of description. At the highest level the following three layers are defined:

(1) Morphology
(2) Syntax
(3) Semantics

Parole	CombUF *	usage features
	ParoleMorpho ?	morphology
	ParoleSyntax ?	syntax
	ParoleSemantic ?	semantic

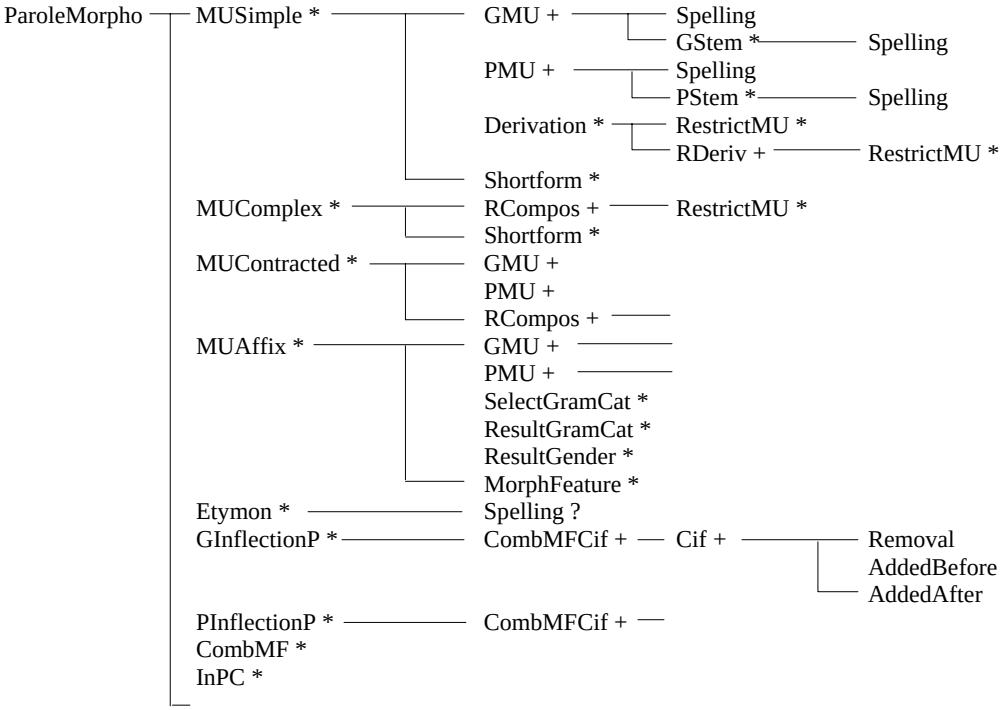
The Genelex view is described as one where any lexical item can be seen either as a progression through the three layers or as a set of information regarding one layer. This modular view has as consequence that for example no distinction related to polysemy is formally needed until the semantic layer. This is in contrast to most of the existing lexica in print form and most of the lexica described in this document.

Morphology

This layer includes orthographical and phonological information centered around Morphological Units (MU). The headword of simple MUs is either a Graphical Unit or a Phonemic Unit. Compound MUs don't have such headword unit, the graphic and phonemic forms are deduced from the units making up the compound unit which are marked by references. Both units (graphical (GMU) and phonemic (PMU)) have a number of attributes and are linked by unique numbers. Both can be associated with stem information, which is an attribute set in its own.

The information to be encoded in this layer is the following:

- info about written and phonetic forms including abbreviated forms
- grammatical categories
- morphological features or combinations of morph features (characterization of a paradigm of inflected forms)
- inflected forms as (1) inflectional behavior of simple words or (2) a system of inflection of compounds
- derivational forms
- abbridged forms
- usage values
- etymological characteristics



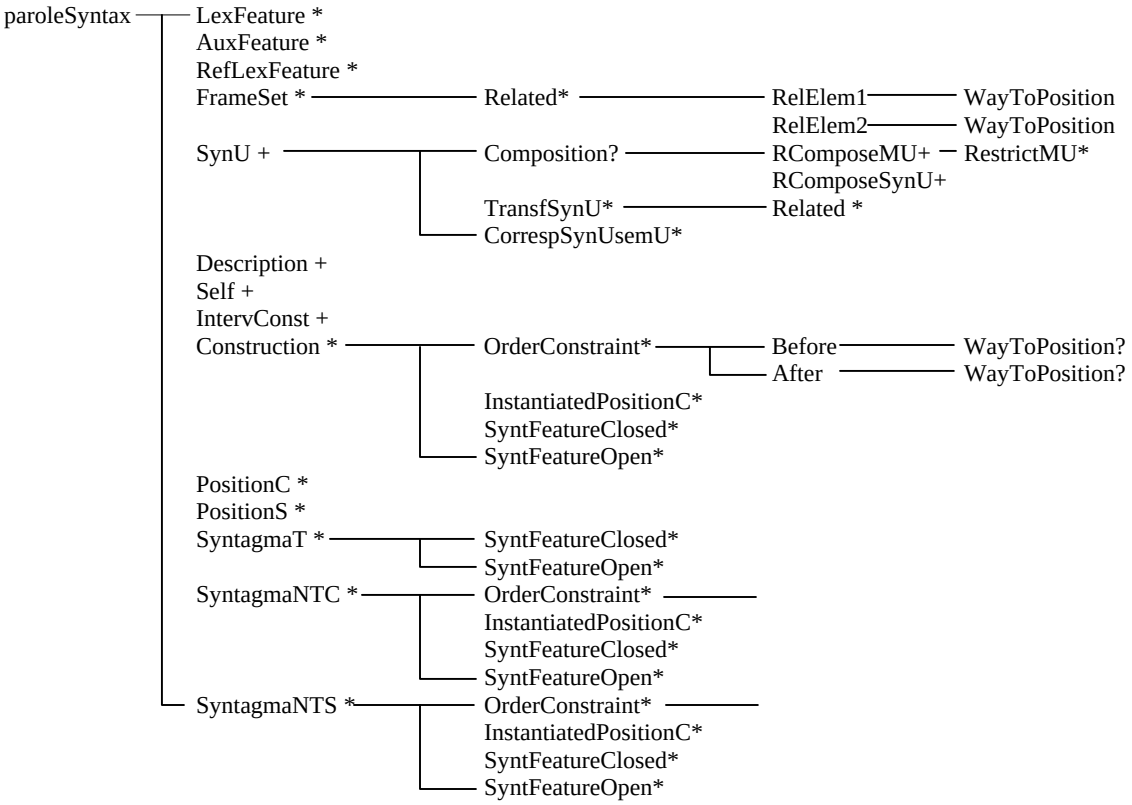
CombComb *

In this scheme each node is associated with a list of attributes. CombMF stands for combination of morphological features, Cif for calculation of inflected form, InPc for systems of inflection for composed morphological units, and CombComb is a relation.

Syntax

The syntactic layer is centered around Syntactic Units (USYN). The information encoded contains(1) information about the construction of an item and (2) information about compound syntactic units including their variations in their surface realization. With respect to the construction of an item the following characteristics are mentioned: basic saturated construction in terms of governed positions, transformational possibilities of such constructions, linearity, pronominalization, optionality of positions, distribution of position, grammatical function of a syntagm in a position, thematic role of the syntagm in a position.

With respect to the compound syntactic units the following characteristics are mentioned: interaction between the “internal” and “external positions”, paradigms of lexical realization of a position in a compound unit, optionality of such a position, accepted transformations



WayToPosition incorporates a recursive mechanism to go down in a possible rewriting. SyntagmaT is a terminal position occupant, SyntagmaNTC a non-terminal position occupant, and SyntagmaNTS is used to describe the internal structure of a compound unit. SyntFeatureOpen/Close refers to restricting features (set of constraints on a phrase). InstantiatedPositionC stands for an interface between SyntagmaNTC/Construction and PositionC. LexFeature is to

specify the lexicalization of a syntactic leaf or of the head of the phrase. AuxFeature encodes auxilliaryused to conjugate the verb. RefLexFeature is similar to LeFeature but used for compounds.

Semantics

Two levels of representation are distinguished: (1) lexical semantics and (2) more cognitive type of contents. Semantic Units (USEM) as main entities of lexical semantics are related to at least one USYN where constraints or filters could restrict the relation. A USEM may also be connected to a linguistic predicate, which can synthesize semantic information about predicative USEMSs. A few characteristics are mentioned which are associated with USEMs:

- a set of semantic features (domain membership, connotative value, scenario pragmatics, ...)
- cross-refs expressing paradigmatic relations (hypo/hyperonymy, synonymy, ...)
- cross-refs expressing semantic derivation
- cross-refs expressing relations of collocation preference

Abstractions of cognitive units (concepts) from USEMs or predicates should be covered as well. Also multilingual semantic relations have to be expressed in a lexicon.

The list of semantic elements is not commented. The structure is just to indicate the complexity of the lexicon.

Multilingual components of Parole are not described here.

2.3 CELEX Lexica

In the CELEX project three lexica were produced (English, Dutch, German) all relying on the same principles. To implement the three DB a relational database design was made resulting in a number of related tables per language. In the following the Dutch version will be depicted. The basic split is made between lemma and wordform tables. Not all attributes which are part of the CELEX tables are listed here.

Lemma Orthography

- Nr.of spellings, spelling nr, spelling status, spelling frequency
- Spelling
  - Head word
  - Headword syllabified
  - Stem
  - Stem syllabified
  - Abstract stem

Lemma Phonology

- Phonetic transcription
  - Headword
  - Headword syllabified
  - Headword syll&stress
  - Stem
  - Stem syllabified
  - Stem syll & stress
- Phonetic patterns
  - Headword syllabified (cv pattern)
  - Stress syllabified
- Phonological stem representation

Lemma Morphology

- Nr. of morpho analysis, status of morpho analysis
- Segmentations
  - Immediate segmentations (stem+affixes, class labels, stem+affix labels, stem allomorphy, affix subst.)
  - Complete segmentations flat
  - Complete segmentations hierarchical
  - Other (nr of components, nr of morphemes, nr of levels)

Lemma Syntax

- Word class
- Subclassification nouns (full gender, de-het distinction, proper noun)
- Subclassification verbs (perfect tense, subclasses, subcategories, lexical verbs)
- Subclassification adjectives (adverbial usage)
- Subclassification numerals (cardinal/ordinal)
- Subclassification pronouns (subclasses)

Wordform Orthography

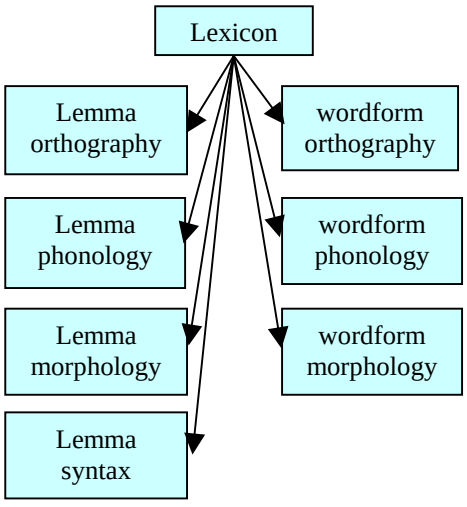
- <<
- spelling
  - plain
  - syllabified

Wordform Phonology

- Phonetic transcription
  - plain
  - syllabified
  - syll&stress
- Phonetic patterns

Wordform Morphology

- Lemma info
  - Id
  - Orthography
  - Phonology
  - Morphology
  - Syntax
  - Frequency
- Inflectional features (separated, singular, plural, diminutive, genitive, infinitive, …)



2.4 Structures found in Bell/Bird Paper

The following structures were found in a paper from Bell/Bird presented at the LDC workshop. All examples were taken from printed lexicons. The characters x, y, z, etc stand for attributes where I was not clear what they stand for.

2.4.1 Examples 1

The first three examples show how sense definitions are treated in or amongst lexical entries.

Javanese example

<lexical entry> := <wordform> <x>"/"<y>"*("<z>")" {<sense nr> <sense description>}*

Orokolo example

<lexical entry> := <wordform>{$^{<nr>}$} {"(var." <variant>")"} {"["<phonetic descr>"]"}
"["<z> "." <u>"]" <gram cat>"." <sense description> {"("explanation")"}
{<lexical entry>}*
<explanation> := <can be various such as a translation>

The terms u, x, y, z are used here to indicate that it was not clear what kind of linguistic content was coded.

A wordform entry can include another related wordform with all descriptors which obviously has some relation to the main entry. Variant information and phonetic form are only given for the first entry. Each wordform can be associated with a number which identifies that subsequent lexical entries have the same wordform but have different senses. Here the lexicon has separate entries for all words sharing the same form but having separate meanings.

Urdu Example

<lexical entry> := "s." <Arabic writing> <orthographic form>","
@(<Arabic writing> <translation>) "s.f." <gloss>

The @ sign is used here to indicate that in the printed form several lexical entries could be associated with one sense definition which they then have in common.

2.4.2 Examples 2

The next two examples are taken to show how order is used.

French Example

<lexical entry> := <wordform>"," <x>, {"m"|"f"}"pl" "-"<plural form> "["<phonetic descr>"]"
<gram cat> <sense>

Tsimshian Example

<lexical entry> := <nr>. <orthographic form>"." <gram cat>"." <sense>"."
"["<phonetic descr>"]" <x>"."

2.4.3 Example 3

Waskia example

<lexical entry> := <wordform> {<x>} <sense> {";" <wordform> <derived form> <sense>}*
{<derived form>"-" <x> {<sense>"("<y>")";}*}*

In this lexicon a number of subsequent entries can occur sharing the same form where the sequence bears information (frequency). Also this lexicon has a nested structure in so far that derived forms can occur within a lexical entry. Multiple sense definitions can bear information as well.

2.4.4 General Models

Typically a lexical entry has an "orthographic headword", pronunciation info, morphosyntactic info, and sense definitions. Often additional info is associated. Sometimes sense definitions are made by referring to another entry.

<lexical entry> := <ortho form>, <phonetic form>, <morphosyntax>, <sense>, ….

or

<lexical entry> := <ortho form>, <phonetic form>, <morphosyntax>, <reference>, ….

Often the headword gives the phonetic form.

The reason for having several lexical entries with the same form as headword can be either in different senses or different phonetic forms. B&B also report about cases where words mentioned in comments refer to other lexical entries.

An entry can be formally subdivided into a head and a body. While the head is the information normally used to search upon, the body contains all other information associated with the entry. So the head is the representative of all relevant information. Sometimes the written lexica don't make that sharp distinction and include body information in the head. The body information can include various types of structures. Sub-nesting is usual. Sub-sub-nesting does not occur in general. Sometimes tables were found to denote vowel length and quality for example of a phoneme within words that can have different morphosyntactic variants.

2.5 Structure used by Schultze-Berndt

ESB uses a lexicon implemented with Hypercard where she has various sub forms for different word categories. Each form allows to hook up a number of attributes of the same type to a headword.

<lexical entry> := <wordform>, <gram class>, {<gloss>}*, {<allomorphs>}*, {<alternatives>}*, {<citation forms>}*, {<categories>}*, {<semantic class>}*, {<argument structure>}*, {<references>}*, {<paradigms>}*, {<speakers>}*, {<translations>}*, {<definition>}*, {<notes>}*, {<other languages>}*, {<semantic relations>}*, {<base forms>}*, {<derivations>}*, {<cognates>}*, {<collocations>}*, {<complex verbs>}*

The structure of the lexicon is fairly simple. Semantic relations are not made explicit by referring to other IDs but by using semantic classes. However, she would like to have references which can link words in comments to other entries or to other comments to create a semantic network of entries or terms.

2.6 Peters' proposal

Wim Peters proposed a lexicon structure which is oriented towards a few major dimensions and groups attributes along these dimensions.

The major dimensions are: Orthography, Morphology, Phonology, Syntax, Morphosyntax, Semantics, Usage. There is just one lexicon which contains all entry types: words, stems, affixes, and phrases.

TOP dimension

<lemma ID>, <citation form>, <entry type>, <language ID>, <frequency>, <etymology>, <inflection type>

Ortho dimension

Wordform Group

<wordform ID>, <lemma ID>, <spelling plain>, <spelling syllabified>, {<freq per resource.}*,
{<sample ID>}*, <supertype>, <status>, <dialect ID>
Dialect Group
<dialect ID>, <dialect name>, <description>, <type>, <status>, <cross ref to dialect ID>
Resource Group
<resource ID>, <type>
Language Group
<language ID>, <language name>, <related language>
Sample Group
<sample ID>, <content>, <translation>, <speaker>, <type>

Phonetic dimension

Phonetic Wordform Group
<lemma ID>, <wordform ID>, <transcription type>, <wordform>, <wordform syllabified>,
<wordform syllabified + stress>, <cv pattern>, <sample ID>, <sample position>,
<phonemic transcription>, <phonetic transcription>, <supertype>

Morphology dimension

Morphology Group
<wordform ID>, <lemma ID>, <analysis nr>, <status>, <immediate segmentation>,
<flat complete segmentation>, <hierarchical complete segmentation>, <stem>,
<nr of morphemes>, <nr of levels>, {<cognates>}*, {<lemma ID>}*

Syntax dimension

Syntax Group
<lemma ID>, <POS>, <subcategorization ID>
Verb Frame
<verb subcat ID>, <slot nr>, <optionality>, <type>, <alternation type>, <semantic type>,
<semantic argument>
Noun Frame
<noun subcat ID>, ...
Adjective Frame
<adjective subcat ID>, ...
Verb Type
<lemma ID>, <POS = v>, <type>, <separable>

Semantic dimension

Semantic Group
<lemma ID>, <POS>, {<sense nr>}*, <gloss def>, <semantic class>, <domain>,
<semantic relation ID>,
Semantic Relation
<semantic relation ID>, <relation label>, <target: lemma ID/POS/sense no>, <relation type>
Regular Polysemy
<lemma ID>, <POS>, <sense nr>, <relation label>, <target: sense nr>

3. Overviews and Proposals

3.1 Grid for Lexicon Evaluation

Within the MILE project a grid was developed to describe the content of lexica. It does not include remarks about structure, but is an enumeration of attributes of a lexical entry. The main categories and their elements are:

Headword
Phonetic transcription
Variant form
Inflected form
Cross-reference
Morphosyntactic information
    POS, inflected class, derivation, gender, number, mass vs count, gradation,
Subdivision counter
Entry subdivision
Sense indicator
Linguistic label
Syntactic Information
    Subcategorization frame, obligatority of complements, auxiliary, light/support construction, periphrastic constructions, phrasal verbs, collocator, alternations
Semantic Information
    Semantic type, argument structure, semantic relations, regular polysemy, domain, decomposition
translation
gloss
near equivalent
example phrase
multiword unit
subheadword
usage note
frequency

3.2 Manning's Lexicon Ideas

Manning and his coworkers have provided very interesting material about online lexica especially for endangered languages. The basic ideas will briefly be reported.

In the Warlpiri and Kirrkirr papers they mainly report on visualization via modern web tools and browsing in for example semantic spaces to find words and study their (semantic) relations. They also speak about orthographic/phonetic similarity as an easy mean to access the lexicon, since often there is no established orthography. The reason for focusing on such access methods is that mainly indigenous people are often not able to work with traditional lexica. The easy access to lexical concepts via semantic relationships and proximity seems to motivate people to use and study it. This is important for all ideas about revitalization of languages.

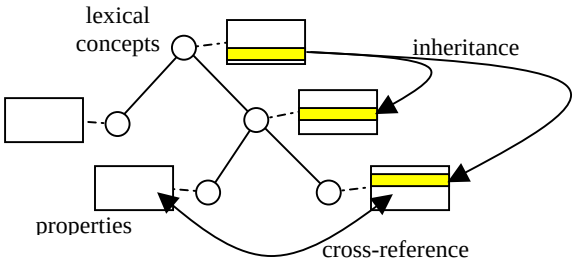
The lexical structure they use offers the kind of power which we know from Shoebox. In fact they seem to handle two formats: one is Shoebox MDF and the other is an XML format which can be generated from the Shoebox file. To my knowledge it is one of the few lexica where cross-references from comments within entries refer to other entries and where tools offer to visualize and to use them.

3.3 Ide/Romary Papers (LORIA)

3.3.1 A Formal Model of Dictionary Structure and Content

In this paper they describe a formal model of a lexicon and methods of structure transformation. Only the first aspect is of interest in this note.

Basically the lexicon is modeled as a tree structure where each node in this tree is associated with a number of properties. Properties can be assigned explicitly to a node or they may be inherited from the parent node. Properties are feature-value pairs whereby values be either atomic or feature-value pairs themselves.



In their model they differentiate between several modes of information propagation dependent on the type of feature: (1) Cumulative features gather all higher level values when traversing the tree. (2) Overwriting features take only one value at a time and a new value is replacing the inherited value. (3) Local Features are not subject of inheritance mechanism.

Another element of their abstract model are "dependency rules" that block the propagation of subordinate features when a privileged feature is overwritten. The values of features can be cross-references to other nodes allowing to implement complex structures.

They did not show in how far this simple abstract model can efficiently cope with the variety of lexical structures we are faced with.

3.3.2 A Flexible Framework for Representing Computational Lexicons
In this paper Ide and Romary discuss the difference between structure and "data categories" (DC). DC are in fact abstract lexical building blocks which can be assembled by structure definitions to form a concrete lexicon. A "Generic Markup Tool" allows to specify a basic structural skeleton making up hierarchies of levels and to associate features with each level. This simple scheme supports modularity and enables easy integration of modules.

The DC as lexicon building blocks should be made available by registries. They can be re-used when they can be retrieved and have a formal structure. Retrieval is supported by metadata descriptions formulated in RDF. Each DC is defined by a style and a vocabulary enabling its automatic integration into larger lexical components.

With a few examples the authors demonstrate how DC can be built and described by metadata and how local lexica could be generated by re-using components. The paper was intended to describe the basic concept, i.e. the real work has to be done.

3.4 Zajac/Viegas/Sheremetyeva Paper (NMSU)

(Title: The Generic Structure of a Lexical Knowledge Base Entry)

The paper which I had access to was a draft version, so my interpretations can only be preliminary. Nevertheless, the ideas described were fairly clear and based on practical experience with an implementation of a computational lexicon and the Tango definition language.

In this paper a multilingual lexical Knowledge Base is based on a set of monolingual dictionaries, translation relations, and a schema describing the structure of the entries in the dictionaries. The structural basis for all lexicon work is the Tango language which is based on typed feature structures for defining and using inheritance hierarchy of type definitions. Each monolingual dictionary is composed of three components: (1) dictionary schema, (2) dictionary data, (3) set of lexical rules to produce new data from existing ones.

Each dictionary is seen as a tree-like structure of senses where the basic entries correspond to word senses. Such entries can be grouped together to larger entries where senses share the same lemma, different categories the same form, different lemmas the same derivational family etc. Inheritance is yielded by stating that all elements defined for higher order entries are defined also for all sub-entries. Relationships between lexical entries are modeled using binary links which are instances of link classes (synonymy, derivation, translation, thesaurus, ...). Links can have arbitrarily complex internal structure and can bear information. Lexical Rules can be seen as productive relations between entries.

Entries have a set of features which the authors call “zones”. The following zones were indicated:

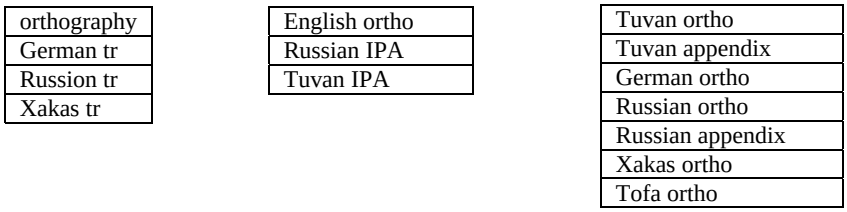
Type	**Feature**
SuperEntry	key, sense
Entry	sense
EntryElements	form, gramcat, semantics, synSem, relation, lexicalRule, translation, usage, etymology, definition, cross-reference, example, note
Relation	dom, range
Form	type, ortho, morph
FormType	0
Orthography	exp,usage
Morphology	lex, infl, drv
MorphLexical	0
Inflectional	0
Grammar	pos, subcategory, frame
POS	0
Subcategory	0
Translations	(list of language codes)
TMR	name, aspect, attitude, modality, set rel (and role names)
LexicalRelations	paradigmatic, syntagmatic
ParaLex	synonym, antonym, hyponym

Hyponymy	basic level
Collocation	base, collocate, freq
LexicalRule	root
Usage	geo, time, dom, style
Geographic	0
Temporal	0
Domain	0
STyle	formality, simplicity, color, force, directness, respect, acceptability, figurative, frequency
Definition	def, source
Translation	dom, range
Example	eg, source
Quotation	eg, source
Citation	eg, source
Cross-Reference	dom, range, note

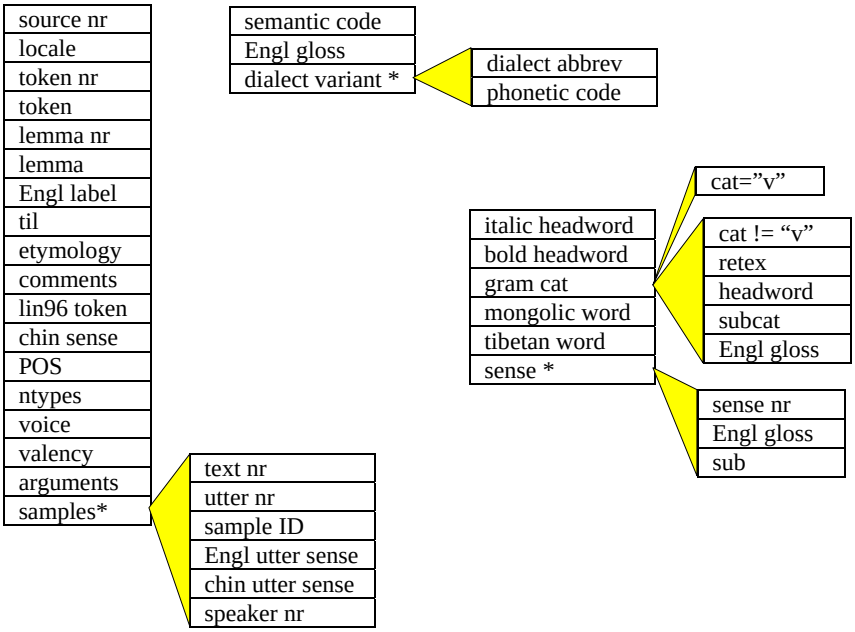
4. Graphical Structure Descriptions

4.1 DOBES Lexica

ASLEP (3 lexica)



Monguor/Salar (3 separate Lexica)



Wichita (related tables)

entry form
entry nr
gram cat
variant form
phonetic form
miscellaneous
complete
label
status
use period
gender
source lang
prerogative
ritual

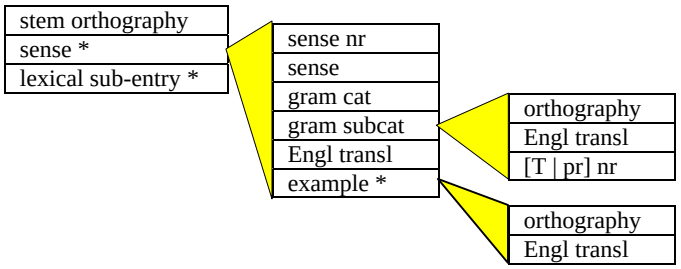
entry form
entry nr
gram cat
gloss
gloss nr
index
usage
gender
slang
pejorative
ritual
example nr

entry form
entry nr
gram cat
gloss
example nr
example
phonetic form
morphemic constit
literal transl
free transl
source

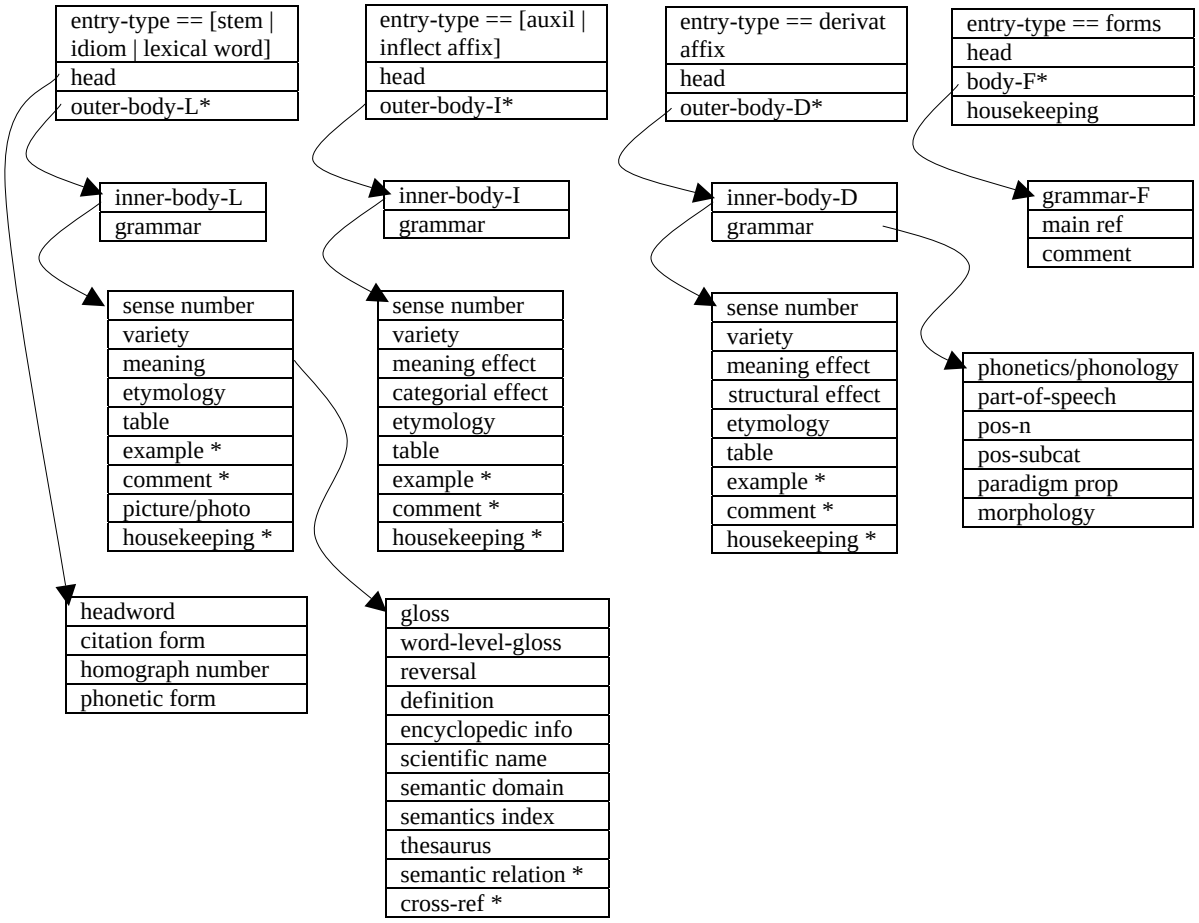
entry form
entry nr
gram cat
para type
para form
para analysis
para gloss
para source
comments
comment SP
SP

entry form
entry nr
gram cat
gloss
parent
parent name
sound word
sound filename
sound nr
speech part
source

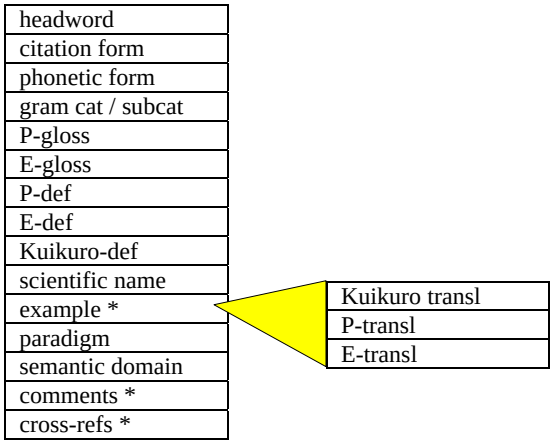
Teop



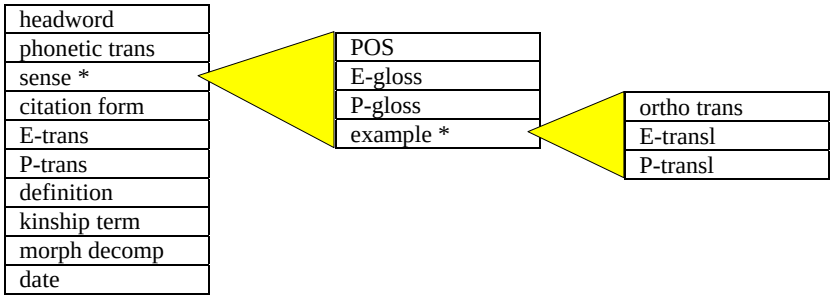
Aweti (only main blocks shown)



Kuikuro



Trumai

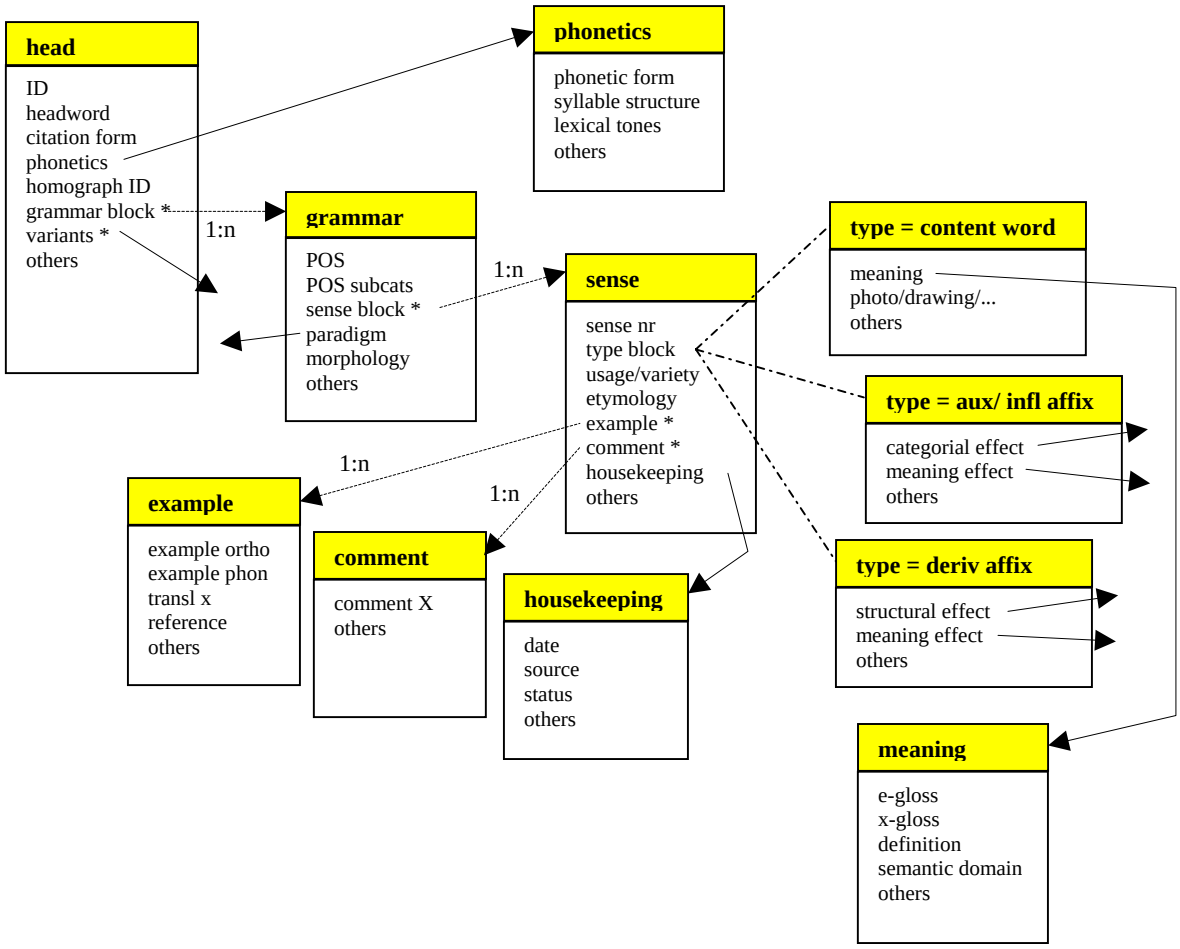


5. Conclusions

In this chapter it is tried to derive suggestions from the different lexica studied. Since the specifications for lexica differ very much, we first develop a generic lexicon scheme which might be sufficient for the DOBES project. Afterwards we will try to extend that model such that it may fit with more general lexica such as GENELEX.

5.1 Generic Lexicon Scheme for Documentation Purposes

Below we define a number of "building blocks", link them together, and introduce some structural properties. Every building block can have as many attributes as people may think of. The specific design and the shown attributes have to be seen as exemplaric and serve to explain structural properties[4]. It should be clear that no one has to use all the attributes which are shown in the following scheme and that everyone can create his/her own block design. Also, this scheme does not include implementation issues, i.e. the scheme can be implemented as a set of tables, as typed-feature structures, or in other ways.



[4] In this exemplaric design it may seem strange that we have put the entry type distinction under grammar-sense. This would lead to the effect that one has to encode structural effects of affixes under sense. This may seem not that convincing. Further, we did not include here the possibility that for example prosodic differences could lead to sense differences and we don't have semantics. It may also be a very useful extension to have metadata and housekeeping info on various levels (perhaps even on attribute level). But again: this scheme is only used to express the structural ideas and not all encoding details.

In the scheme we can identify 3 different types of structural properties: (1) We can define a simple grouping of attributes which can be re-used at different positions in the lexicon (solid arrows). This includes the option that a block can be referenced by several higher blocks. (2) We can define blocks of attributes which can occur several times such as various examples (stippled arrows). (3) Different attribute sets dependent on some condition such as the type of the headword (stippled lines).

In addition to defining these structural elements the lexicon has to support **cross-references** from any attribute in some block to any other lexical entry or attribute of an entry (the possibility to define cross-refs even between elements of attributes was not required by the DOBES group members). In DOBES it seems to be sufficient when the cross-refs have just a label to be able to distinguish them. Cross-refs can lead to cyclic structures, of course. If it makes sense to point for example from one comment field to another within the same entry we even could get cyclic structures within an entry (can't see for what this could be relevant).

We did not make any statements about inheritance such as suggested in 3.3.1. To better understand the implications of such mechanisms we need to discuss concrete examples in the DOBES group.

The structure above does not make statements whether the headword is a wordform or a lemma/stem. This distinction is relevant for much of the work within DOBES and the focus will change dependent on the language. From Wichita for example we have understood that utterances in general are highly inflected wordforms, so we assume that the Wichita lexicon will focus on wordforms as entries and associate many attributes with these forms. In languages such as Aweti and Teop for example it seems to be necessary and useful to have both a wordform lexicon as well as a lemma lexicon. Most of the attributes will be associated with the lemma to avoid information doubling. Also cross-references will normally be made between the lemma entries. The wordform lexicon (wordlist table) is mainly to be used to make the link between the corpus and the lemma lexicon. This structural distinction was made in CELEX also. The structural elements to be supported are the same in both cases, i.e. they are independent from the type of the entry.

From chapter 2.1 we know that currently most of the DOBES teams would only use a fraction of the structural possibilities and only use a minimal set of attributes. Of course, the scheme shown above allows one to create a simple table if this is required.

A lexicon tool for DOBES as it stands now would have to support the following structure related functionality[5]:

1. Create a block which has a label and a number of attributes.
2. Define the attribute types (value range and constraints) of that block.
3. Define per attribute whether it is a leaf, a label referring to a sub-block, or a label combined with a condition on an attribute value.
4. Define whether the label can refer to one or several sub-block instantiations.
5. In case of repetition, semi-automatically generate numbering for cross-referencing purposes.
6. Define the labels of the cross-refs to be used and allow the user to actually create links.
7. The tool must allow one to modify the structure especially with adding attributes, cross-refs, and sub-blocks.
8. It must be possible to define attributes which contain pointers to various other type of information such as sketch grammars, text sources, etc.
9. Perform consistency checks

These structural features make only sense of course, if they are supported by operations. It should be possible to define sort-orders and to specify which attribute should be used for sorting. Searching should involve combinations of attributes from various building blocks. Visualization should include the usage of

[5] This is not the place to speak about other functionality such as searching, sorting, printout generation, lexicon-corpus interaction etc. This will be dealt with in another document.

references and relations such as embedded in the “semantic domain” attribute. Much more can be said including the relation between lexicon and corpus. This has to be worked out more carefully.

An operation of special interest is that one to generate printed versions of a lexicon. Here we rely on techniques (XSLT) which allow the user to specify which structural elements should be arranged on paper in a certain layout, in a certain type face etc.

For two concrete lexica used within DOBES we briefly indicate which steps would have to be taken to implement such lexica with the help of the tool sketched above.

1. ASLEP

- Create a group with a few attributes such as “orthography”, “German translation”, “Russian translation” etc. All attributes are leafs and are not constrained.
- Specify that the character set to be used in the Russian translation is Cyrillic.
- etc

2. Teop

- Create the following building blocks:
    - “main entry” with the attributes “stem orthography”, “sense”, “example”, the two last referring to sub-blocks with many instantiations.
    - “sense” with the attributes “sense nr”, “sense”, etc. all being leaves except the example field which refers to many instantiations of the example sub-block.
    - “example” with the leaf attributes “orthography”, “Engl. translation” etc.
- Specify that the “sense nr” field can only be a positive integer.
- Specify the possible values for the field “gram cat”.
- etc

The Wichita table structure can be implemented in the same way. Also the complex looking Aweti structure can be implemented that way.

5.2 Unification in DOBES

There are some topics to be addressed in DOBES which are comparable to those discussed for annotations:

- Can agreements be made on linguistic level about building blocks, their attribute set, and their re-usage by the different teams?
- Is there a core lexicon structure which is mandatory for all teams within DOBES?
- Can the value range of certain fields be generally defined?
- In how far does the variety of languages limit the unification?

The issue is here the same: Setting up a coherent archive requires a high degree of unification on various levels. If there is a structure which is general enough a unification could be achieved at least on structural level.

Glossar

ALM	Abstract Lexicon Model, a generic model of a computerized lexicon)
CELEX	a computer lexicon project in The Netherlands with large lexica in three languages (Dutch, English, German)
DTD	Document Type Definition, a description of the structure of a class of XML documents
RDF	Resource Description Framework, an XML-based language with help of which one can define structures build up of other building blocks and which allows to describe semantics
XML	eXtended Markup Language, a language which allows to represent the hierarchical structure of documents
XML-Schema	a more powerful mechanism to define the structure of an XML document