**Participants:**

Bettina

Max

Defining prerequisites: **where** (lexicon or word-form level) do we want to describe **what** (lexical entries or morphs) in **which way** (linear or binary segmentation)?

Working examples

German noun:

*Verkehrszeichen* ‘traffic sign’

Components:

Lexical entry: Verker ‘traffic’

Lexical entry: Zeichen ‘sign’

Empty morph (or genetive): infix -s-

English adjective:

*unprettified*

Components:

Lexical Entry: prettify

Lexical Entry: prefix un-

Grammatical morph: suffix -ed

1. Sometimes grammatical morphs are also part of word-formation processes and hence entailed within lexical entries and not only their word-forms. At the moment the decomp and ontolex module do not allow to describe them as ontolex:Affix resources because they are not considered as separate lexical entries.

Do we want to allow that grammatical morphs are included in the decomposition of lexical entries that are compound or derived words? In what way?

1. The ontolex:Affix class is reserved for all morphs that are also lexical entries. This might cause confusion among morph resources because all other (grammatical) morph need to have another type assertion and also cannot occur in object slots that have ontolex:LexicalEntry as range.

What should be the difference in describing lexical entry morphs and non-lexical entry morphs (e.g. negative prefix *un-* vs. participle suffix *-ed*)?

1. In morphological analysis segmentation is achieved in binary layers. The Component resources in the decomp module, however, “represent a fixed list of each of the elements that compose a lexical entry.” The given example in the ontolex specification for “AfricanSwineFever” uses the decomp:subterm property which again links only lexical entries and the decomp:constituent property that links the Component instances is rather fuzzy defined (can an inflected form or a grammatical/empty morph also be a component). As I understand it, all linear segmented elements can be component but also all complex elements (“African” is not fully segmented), but what is then the “fixed” list?

Do we want to adopt such a listing also for complex derived words or shall a binary segmentation be applied?