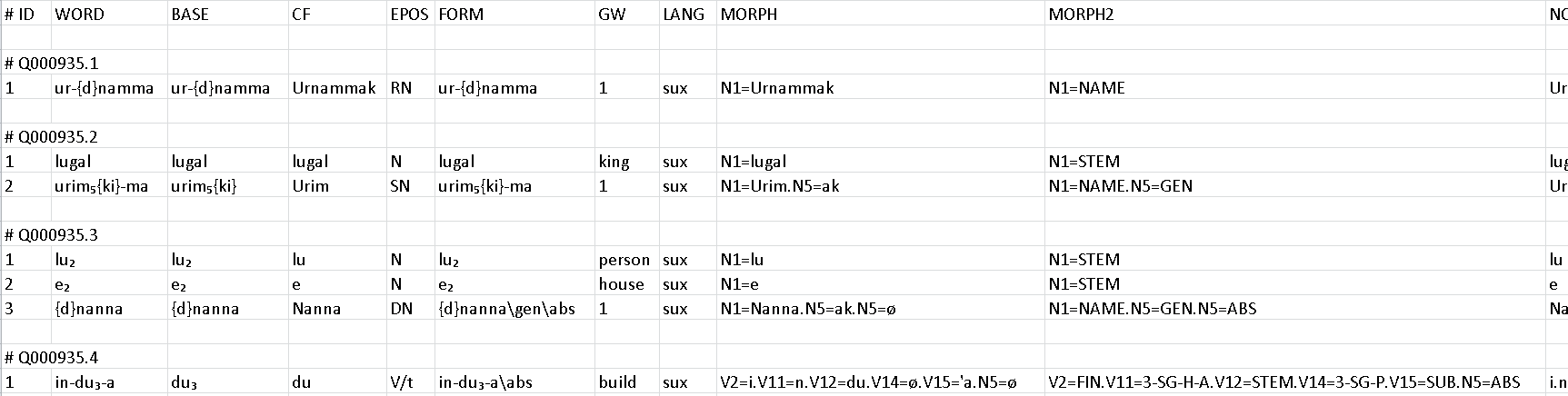
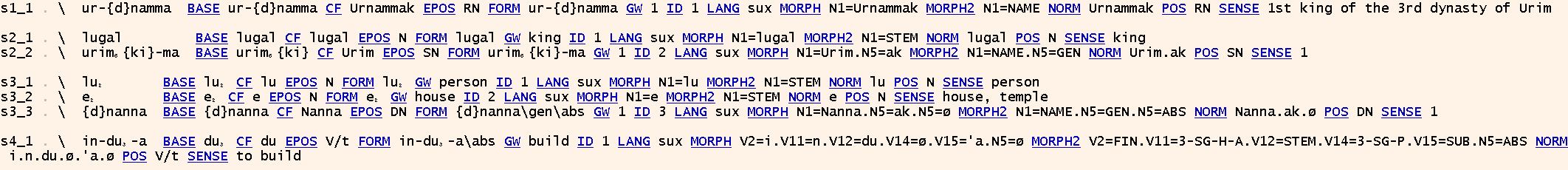
# ETSCRI-ePSD linking with etscri-conll2lod.sh

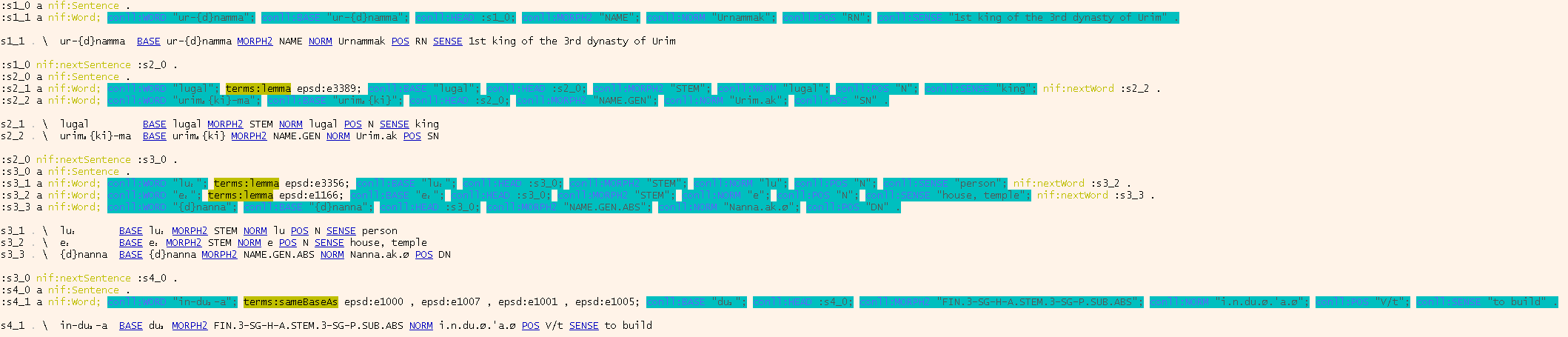
cat data/Q000935.conll

Original CoNLL file. Can be somewhat more conveniently viewed with a spreadsheet software as below.

etscri-conll2lod.sh consults (or constructs, if it isn’t found) a lemon/ontolex compliant index for the ePSD, whose URIs are provided to the linking.

bash etscri-conll2lod.sh -grammar

This shows the original annotations in a dependency visualization. Without syntactic dependencies, the first two columns are not meaningful, but the compact representation of annotations can still be useful. Note that this only shows information from the conll: namespace. Dictionary links are thus not shown.

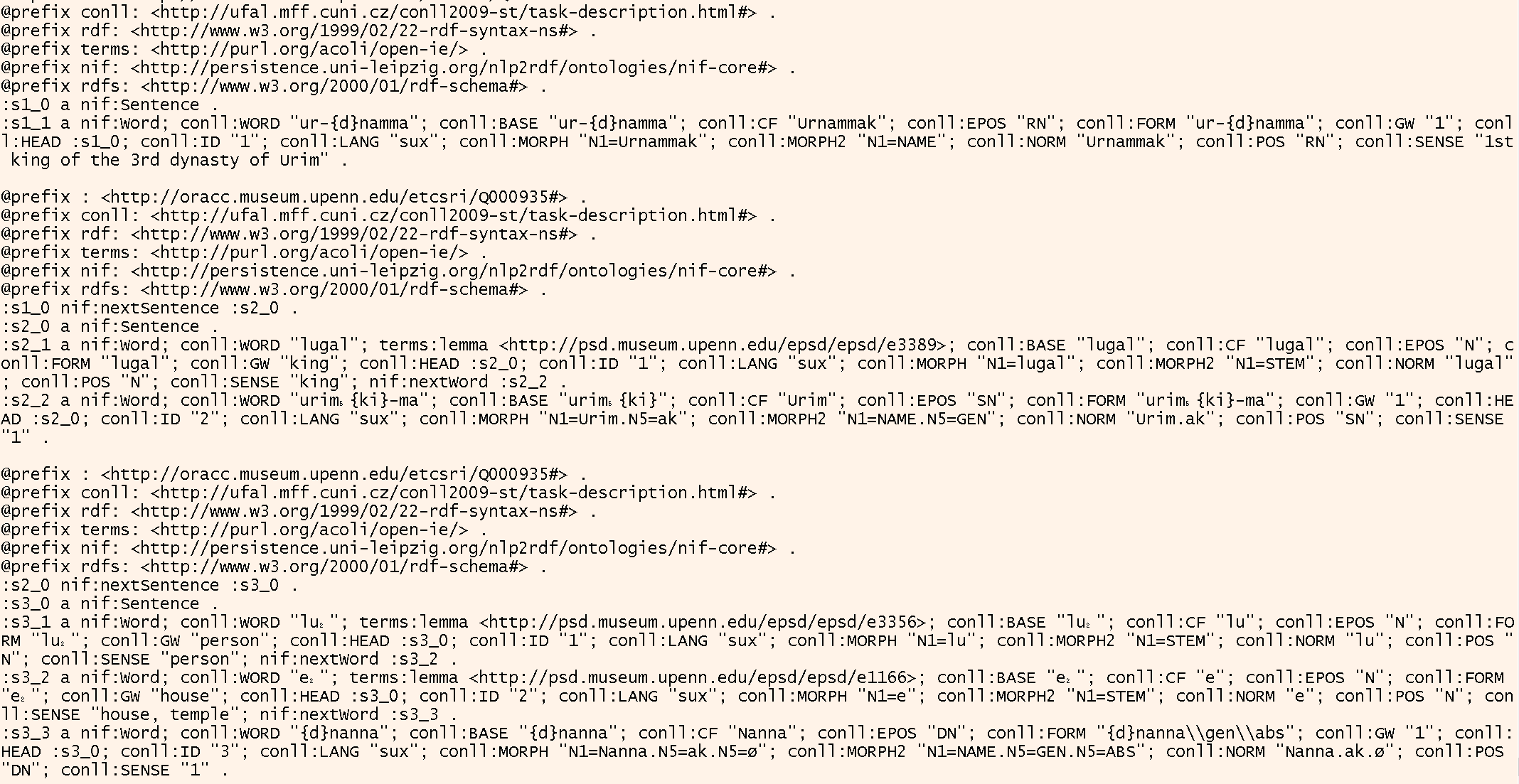
bash etscri-conll2lod.sh -debug –grammar

Using -debug, we show selected original anotations only (in the same view as –grammar), together with the corresponding RDF code (syntax highlighting, same elements removed). Note that only the RDF shows dictionary links. Dictionary links are represented with yellow background.

We provide four qualities of linking:

* sameFormAs: link with one or more forms that are string-identical, no disambiguation
* sameBaseAs: link with one or more lexical entries that have a string-identical canonical form, no disambiguation
* lemma: link with exactly one lexical entry, disambiguated for homonymy
* sense: link with exactly one lexical sense (of an entry), full word sense disambiguation

bash etscri-conll2lod.sh # CoNLL-RDF output



Without parameters, we write machine-readable CoNLL-RDF. CoNLL-RDF is a relatively compact, application-specific subset of Turtle. However, as sentences are processed individually, prefix declarations are spelled out for every individual sentence.

CoNLL2RDF does not show this behavior, but requires the entire dataset to be held in memory, hence, we build on CoNLLStreamExtractor.