GernEdiT – The GermaNet Editing Tool

Verena Henrich, Erhard Hinrichs
University of Tübingen, Department of Linguistics
Wilhelmstr. 19, 72074 Tübingen, Germany
E-mail: verena.henrich@uni-tuebingen.de, erhard.hinrichs@uni-tuebingen.de

Abstract

This paper introduces *GernEdiT* (short for: GermaNet Editing Tool), a new graphical user interface for the lexicographers and developers of GermaNet, the German version of the Princeton WordNet. GermaNet is a lexical-semantic net that relates German nouns, verbs, and adjectives. Traditionally, lexicographic work for extending the coverage of GermaNet utilized the Princeton WordNet development environment of lexicographer files. Due to a complex data format and no opportunity of automatic consistency checks, this process was very error prone and time consuming. The GermaNet Editing Tool GernEdiT was developed to overcome these shortcomings.

The main purposes of the GernEdiT tool are, besides supporting lexicographers to access, modify, and extend GermaNet data in an easy and adaptive way, as follows: Replace the standard editing tools by a more user-friendly tool, use a relational database as data storage, support export formats in the form of XML, and facilitate internal consistency and correctness of the linguistic resource. All these core functionalities of GernEdiT along with the main aspects of the underlying lexical resource GermaNet and its current database format are presented in this paper.

1 Introduction

The traditional development of GermaNet[1] (Kunze and Lemnitzer, 2002) was based on lexicographer files. These were originally developed for the English Princeton WordNet (Fellbaum, 1998) and up until recently also used for GermaNet. Due to the complex data format of lexicographer files, this process was error prone, and in the absence of automatic consistency checks, it was difficult to detect syntax errors or inconsistencies in the GermaNet hierarchy.

We developed *GernEdiT* (short for: GermaNet Editing Tool), a graphical user interface for the lexicographers and developers of GermaNet, to overcome these detrimental effects. The purpose of this tool is fourfold:

1. To replace the standard editing tools for wordnets, such as the lexicographer files, by a more user-friendly visual tool, that aids in the navigation through the GermaNet word class hierarchies, so as to find the appropriate place in the hierarchy for new *synsets* (short for: synonymy set) and *lexical units*.

2. To link the editing tool to state-of-the-art relational database technology that supports versioning and collaborative annotation by several lexicographers working in parallel and at the same time provides a sustainable data format for long-term preservation of the data.

3. To support export formats for the GermaNet data in the form of XML schemas so as to facilitate use of the stored data in a variety of applications.

4. To facilitate internal consistency of the GermaNet data such as appropriate linking of lexical units with synsets, connectedness of the synset graph, and automatic closure among relations and their inverse counterparts.

All these functionalities along with the main aspects of the underlying lexical resource GermaNet are presented in this paper.

The remainder of this paper is structured as follows: Section 2 provides a general introduction to GermaNet. Sections 3 and 4 provide further details about the actual lexicographic work required to extend the lexical coverage of GermaNet and about the shortcomings of the software infrastructure prior to the development of GernEdiT. The functionality of the editing tool GernEdiT is then presented in section 5 along with further details about the internal structure of the GermaNet data. The tool evaluation in section 6 is followed by a conclusion and future work section.

2 The Structure of GermaNet

GermaNet is a lexical semantic network that is modeled after the Princeton WordNet for English. It partitions the lexical space into a set of concepts that are interlinked by semantic relations. A semantic concept is modeled by a *synset*. A synset is a set of words (called *lexical units*) where all the words are taken to have (almost) the same meaning. Thus a synset is a set-representation of the semantic relation of synonymy, which means that it consists of a list of lexical units and a paraphrase (represented as a string). The lexical units in turn have frames (which specify the syntactic valence of the lexical unit) and examples. The list of lexical units for a synset is never empty, but any of the other properties may be.

[1] See http://www.sfs.uni-tuebingen.de/GermaNet/

```
1  (*** Nüsse ***)
2  {Nuss, Nuß*o, Nusskern, ?festes_Nahrungsmittel,@ nomen.Pflanze:Nuss,@ ('der essbare Kern einer Nuss')}
3  {Haselnuss, Haselnuß*o, Haselnusskern, Haselnußkern*o, Nuss,@ nomen.Pflanze:Haselstrauch,#}
4  {Kokosnuss, Kokosnuß*o, nomen.Pflanze:Kokospalme,# Nuss,@}
5  {Walnuss, Walnuß*o, Walnusskern, Walnußkern*o, Nuss,@ nomen.Pflanze:Echte_Walnuss,#}
6  {Betelnuss, Betelnuß*o, nomen.Pflanze:Betelnussbaum,# Nuss,@ Genussmittel,@}
7  {Erdnuss, Erdnuß*o, Erdnusskern, Erdnußkern*o, Nuss,@ nomen.Pflanze:Erdnusspflanze,#}
8  {Paranuss, Paranuß*o, Paranusskern, Paranußkern*o, Nuss,@ nomen.Pflanze:Paranussbaum,#}
9  {Pistazie, Pistazienkern, Nuss,@ nomen.Pflanze:Pistazienbaum,#}
10 {Mandel, Mandelkern, Nuss,@ nomen.Pflanze:Mandelbaum,#}
11 {Cashewkern, Cashewnuss, Cashewnuß*o, Nuss,@ nomen.Pflanze:Acajubaum,#}
```

Figure 1: Excerpt from lexicographer file *nomen.Nahrung*

There are two types of semantic relations in GermaNet: *conceptual* and *lexical relations*. Conceptual relations hold between two semantic concepts, i.e. synsets. They include relations such as hyperonymy, part-whole relations, entailment, or causation. Lexical relations hold between two individual lexical units. Antonymy, a pair of opposites, is an example of a lexical relation.

GermaNet covers the three word categories of adjectives, nouns, and verbs, each of which is hierarchically structured in terms of the hyperonymy relation of synsets. For each of the word categories the semantic space is divided into a number of semantic fields.

3 GermaNet Development with Lexicographer Files

GermaNet development started in 1997, and is still ongoing. The following are up-to-date statistics of GermaNet's version 5.2 contents (release of December 2009):

- Number of synsets: 61575
- Number of lexical units: 84859
- Number of literals: 76981
- 1.38 lexical units per synset
- 1.10 readings per literal
- Number of lexical relations: 26564
- Number of conceptual relations: 73624

Traditionally, lexicographic work for extending the coverage of GermaNet utilized the Princeton WordNet development environment of lexicographer files[2] and accompanying software, in particular grinder[3]. The lexicographer files are specified in plain text and contain synsets for a particular semantic domain and part of speech.

To get an impression of what these lexicographer files looked like, Figure 1 shows an excerpt of the lexicographer file *nomen.Nahrung* (German for: *nouns.food*). In this example, line 1 simply shows the comment *(*** Nüsse ***)* (German for: *nuts*) for an easier reading and a better understanding for the lexicographers that worked with those files. Each of the lines 2 to 11 contains a synset (encoded between the curly brackets).

The synset in line 2 contains the lexical units *Nuss* (German noun for: *nut*), with its orthographic variant *Nuß* (orthographic variants were marked with **o*), and *Nusskern* (German noun for: *kernel of a nut*). The question mark in front of a lexical unit states that this lexical unit is an *artificial concept*. Concepts which cannot be lexicalized by a simplex word in German, but which are required in order to build a proper hierarchy are marked as artificial. As can be seen on line 2, *festes_Nahrungsmittel* (German for: *solid food*) is such an artificial concept. In this case, *festes_Nahrungsmittel* also acts as a hyperonym, which is encoded by placing a comma followed by an at sign (,@) after the word. Another hyperonym is *nomen.Pflanze:Nuss* (German for: *nouns.plant:nut*). This hyperonym relation refers to a synset which is in another lexicographer file called *nomen.Pflanze*. The following phrase *der essbare Kern einer Nuss* (German phrase for: *the edible kernel of a nut*) represents an explanatory paraphrase for this synset. Paraphrases are encoded by single-quoting them within parenthesis.

The following lines 3 to 11 contain synsets that are encoded analogously. These synsets all represent hyponyms of the synset in line 2. This is the case because there is a *Nuss,@* in each line. A comma followed by a hash mark (,#) marks the holonymy relation. Other relations are encoded by similar diacritics.

The lexicographers that extended GermaNet needed to search across these complex text files to find the correct place to insert new a lexical unit. This task was error prone and time-consuming.

The grinder tool can process the lexicographer files and produce an internal data format that is suitable for querying the wordnet data and the lexical and conceptual relations interlinking them.

[2] See http://wordnet.princeton.edu/man/lexnames.5WN.html
[3] See http://wordnet.princeton.edu/man/grind.1WN.html

4 Shortcomings of Traditional GermaNet Development

Over the years, it became more and more obvious that the use of lexicographer files for extending the coverage of GermaNet led to a number of detrimental effects (as Figure 1 and its explanation in section 3 confirm):

1. As GermaNet grew in coverage, it became more and more difficult to identify the appropriate anchor point for inserting new lexical units and synsets into the existing hierarchy. Lexicographer files do not support any visualization of the hierarchy. The only way to navigate the graph is to follow the conceptual relations via their semantic pointers, which are encoded in the lexicographer files as textual diacritics. This is a rather frustrating and error prone process.

2. Since lexicographer files use a complex data format, lexicographers often made syntax errors during extending the GermaNet coverage. Also, the data format is not sufficiently constrained. For example the data format does not automatically enforce the connectedness of the hierarchy. As a consequence, the GermaNet data often suffered from such internal inconsistencies. Since the grinder tool only checks for syntactic well-formedness of lexicographer files, such semantic inconsistencies often remained unnoticed.

3. In practice, GermaNet development involved the parallel work of several lexicographers. Since lexicographer files as such do not support any versioning, only one lexicographer at a time was allowed to edit a given lexicographer file. This led to considerable data management overhead. For example, if several lexicographers successively modified a given file, it was next to impossible to track the sequence of changes or to undo specific edits that were introduced erroneously.

5 The GermaNet Editing Tool

The GermaNet Editing Tool GernEdiT was developed to overcome the above mentioned shortcomings. It provides a graphical user interface, implemented as a Java Swing application, which primarily allows maintaining the GermaNet data in a user-friendly way. It supports lexicographers who need to access, modify, and extend GermaNet data by providing these functions through simple button-clicks, searches, and form editing. There are several ways to search data and browse through the GermaNet graph. These functionalities support lexicographers, among other things, in finding the appropriate place in the hierarchy for the insertion of new synsets and lexical units.

The editor represents an interface to a relational database (see subsection 5.2 for more details about the underlying database format), where all GermaNet data is stored from now on. This allows versioning and collaborative annotation on GermaNet by several lexicographers working in parallel. The functionality of an editing history shows all modifications on the GermaNet data, with the information about who made the change and how the modified item looked before.

5.1 The User Interface

Figure 2 illustrates the main user panel of GernEdiT. It shows a *Search* panel above, two panels for *Synsets* and *Lexical Units* in the middle, and four tabs below: a *Conceptual Relations Editor*, a *Graph with Hyperonyms and Hyponyms*, a *Lexical Relations Editor*, and an *Examples and Frames* tab.

It is possible to search for words or synset database IDs. The check box *Ignore Case* offers the possibility of searching without distinguishing between upper and lower case.

In Figure 2, a search for synsets consisting of lexical units with the word *Nuss* (German noun for: *nut*) has been executed. Accordingly, the *Synsets* panel displays the three resulting synsets that match the search item. The *Synset Id* is the unique database ID that unambiguously identifies a synset and which can also be used to search for exactly that synset. *Word Category* specifies whether a synset is an adjective (*adj*), a noun (*nomen*), or a verb (*verben*), whereas *Word Class* classifies the synsets into semantic fields. The word class of the selected synset in Figure 2 is *Nahrung* (German noun for: *food*). The *Paraphrase* column contains a description of a synset, e.g., for the selected synset the paraphrase is: *der essbare Kern einer Nuss* (German phrase for: *the edible kernel of a nut*). The column *All Orth Forms* simply lists all orthographical variants of all its lexical units.

Which lexical units are listed in the *Lexical Units* panel depends on the selected synset in the *Synsets* panel. Here, *Lex Unit Id* and *Synset Id* again reflect the corresponding unique database IDs. *Orth Form* (short for: *orthographic form*) represents the correct spelling of a word according to the rules of the spelling reform *Neue Deutsche Rechtschreibung* (Rat für deutsche Rechtschreibung, 2006), a recently adopted spelling reform. In our example, the main orthographic form is *Nuss*. *Orth Var* may contain an alternative spelling that is admissible according to the *Neue Deutsche Rechtschreibung*.[4] *Old Orth Form* represents the main orthographic form prior to the *Neue Deutsche Rechtschreibung*. This means that *Nuß* was the correct spelling instead of *Nuss* before the German spelling reform. *Old Orth Var* contains any

[4] An example of this kind is the German word *Delfin* (German noun for: *dolphin*). Apart from the main form *Delfin*, there is an orthographic variant *Delphin*.

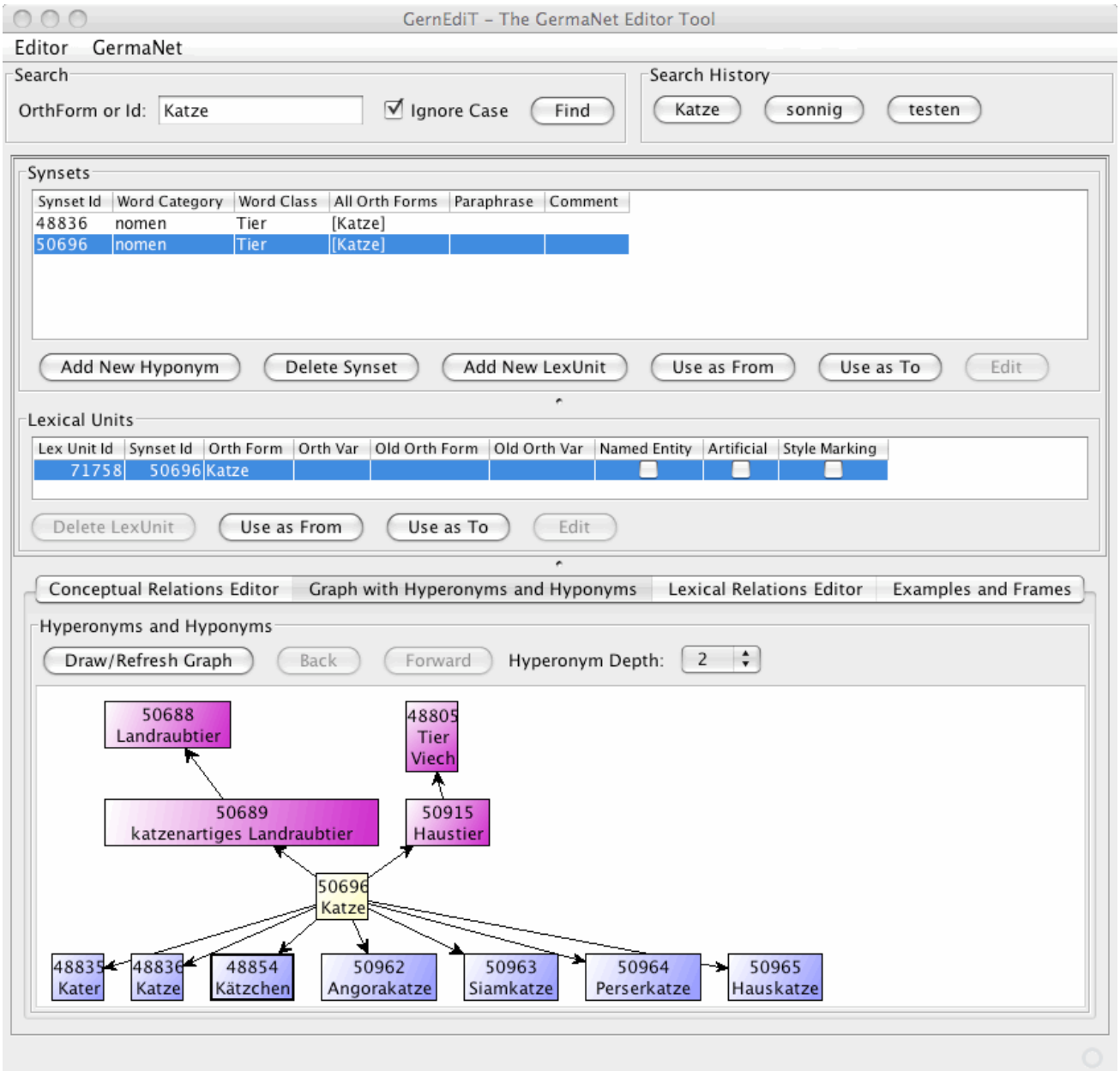

Figure 2: Main view of GernEdiT

accepted variant prior to the *Neue Deutsche Rechtschreibung*. The *Old Orth Var* field is filled only if it is no longer allowed in the new orthography.

The Boolean values *Named Entity*, *Artificial*, and *Style Marking* express further properties of a lexical unit, whether the lexical unit is a named entity, an artificial concept node, or a stylistic variant.

In both the *Synsets* and the *Lexical Units* panel, the properties of these items can be edited by a click in the corresponding table cell. Furthermore, the buttons *Add New Hyponym* and *Add New LexUnit* can be used to insert a new synset or lexical unit at the selected place in the GermaNet graph, and the buttons *Delete Synset* and *Delete LexUnit* remove the selected entry, respectively.

Consistency checks take effect for both the cell editing (e.g., the main orthographic form of a lexical unit may never be empty) and the button functionalities (e.g., if a synset consists only of one lexical unit, it is not possible to delete that lexical unit). Also, if the deletion of a synset would violate the complete connectedness of the GermaNet graph, it is not possible to delete that synset.

For both the lexical units and the synsets, there are two buttons *Use as From* and *Use as To*, which help to add new relations (see the explanation of Figure 4 below which explains the editing of the relations).

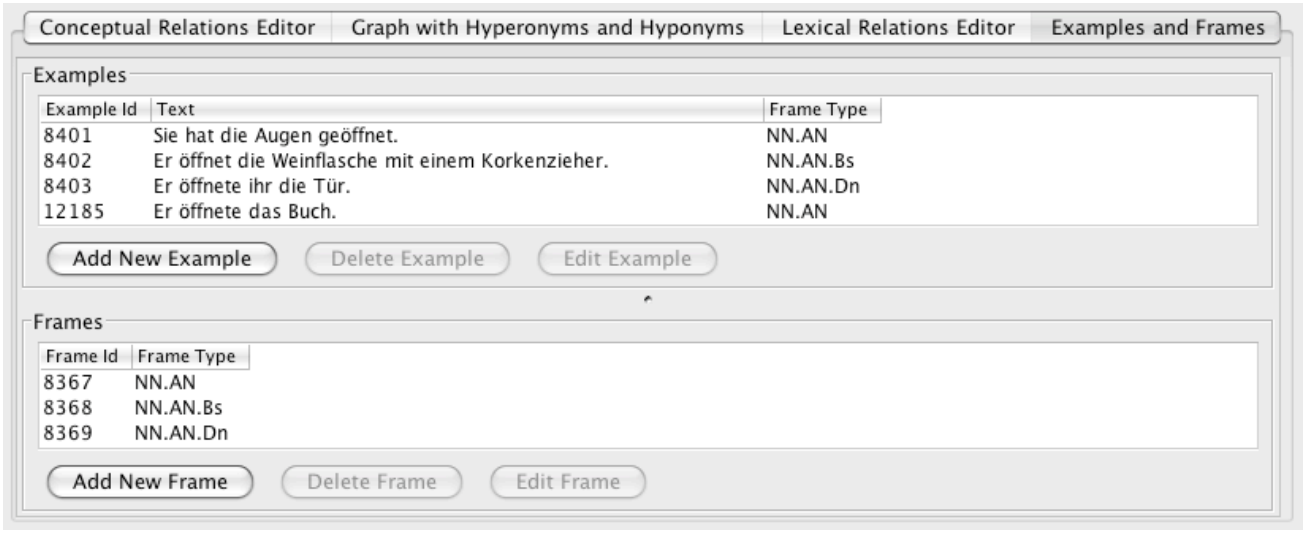

Figure 3: Examples and Frames tab

There is the possibility to show a graph with all hyperonyms and hyponyms of a selected synset in the tab *Graph with Hyperonyms and Hyponyms*. This graph, which is also shown in Figure 2, visualizes a part of the hierarchical structure of GermaNet, which is constituted by the hyperonymy-hyponymy relation between the synsets. The *Hyperonym Depth* chooser allows unfolding the graph to the top. As it is not possible to visualize the whole GermaNet contents at once, the graph can be seen as a window to GermaNet. A click on any synset node within the graph, navigates to that synset. This functionality supports lexicographers especially in finding the appropriate place in the hierarchy for the insertion of new synsets.

The screenshot of GernEdiT in Figure 2 shows the same synset *Nuss* that is also shown in the excerpt of the lexicographer files in Figure 1. A comparison of GernEdiT in Figure 2 and the lexicographer files in Figure 1 reveals obvious benefits of the GernEdiT tool. Visualization by means of lists and graphs is much preferred by lexicographers over text file editing.

Besides modifying synsets and lexical units by editing the cells in the *Synsets* and *Lexical Units* panels, it is possible to modify the frames and examples of a lexical unit and both types of relations – conceptual and lexical ones. This editing is done in the corresponding tabs that appear at the bottom of Figure 2.

The *Examples and Frames* tab is shown in Figure 3 with all examples and frames of the lexical unit *öffnen* (German verb for: *to open*). Frames specify the syntactic valence of the lexical unit. Each frame can have an associated example that indicates a possible usage of the lexical unit for that particular frame. In addition to modifying an example or a frame by clicking in a cell for editing, there are adding and deleting buttons for examples and for frames available in this tab.

The *Conceptual Relations Editor* tab, shown in Figure 4, displays all conceptual relations of a selected synset. In this example all conceptual relations of the synset containing *Feuer* (German noun for: *fire*) are shown. To create a new relation, one needs to use the buttons *Use as From* and *Use as To* from Figure 2, which insert the ID of the selected synsets from the *Synsets* panel in the corresponding *From* or *To* field in Figure 4. The button *Delete ConRel* allows deletion of a conceptual relation if all consistency checks are passed. It is not possible to delete a relation if it, for example, violates the complete connectedness of the GermaNet graph.

The *Lexical Relations Editor* tab, which is also included in Figure 2, supports editing all lexical relations. It is not displayed separately for reasons of space, but it is analogue to the *Conceptual Relations Editor* tab for editing conceptual relations.

GernEdiT facilitates internal consistency of the GermaNet data. Besides the consistency checks that were already mentioned, this is achieved by the workflow-adopted design of the editor. It is not possible to insert a lexical unit without specifying the corresponding synset. On deletion of a synset, all corresponding data such as conceptual relations, lexical units with their lexical relations, frames, and examples, are deleted simultaneously.

There are further functionalities available through the file menu. Besides retrieving the up-to-date statistics of GermaNet, it is possible to list all synsets or lexical units with their properties and to adjust a very detailed filter to that list: e.g., filtering the lexical units by their frames or parts of their orthographical forms.

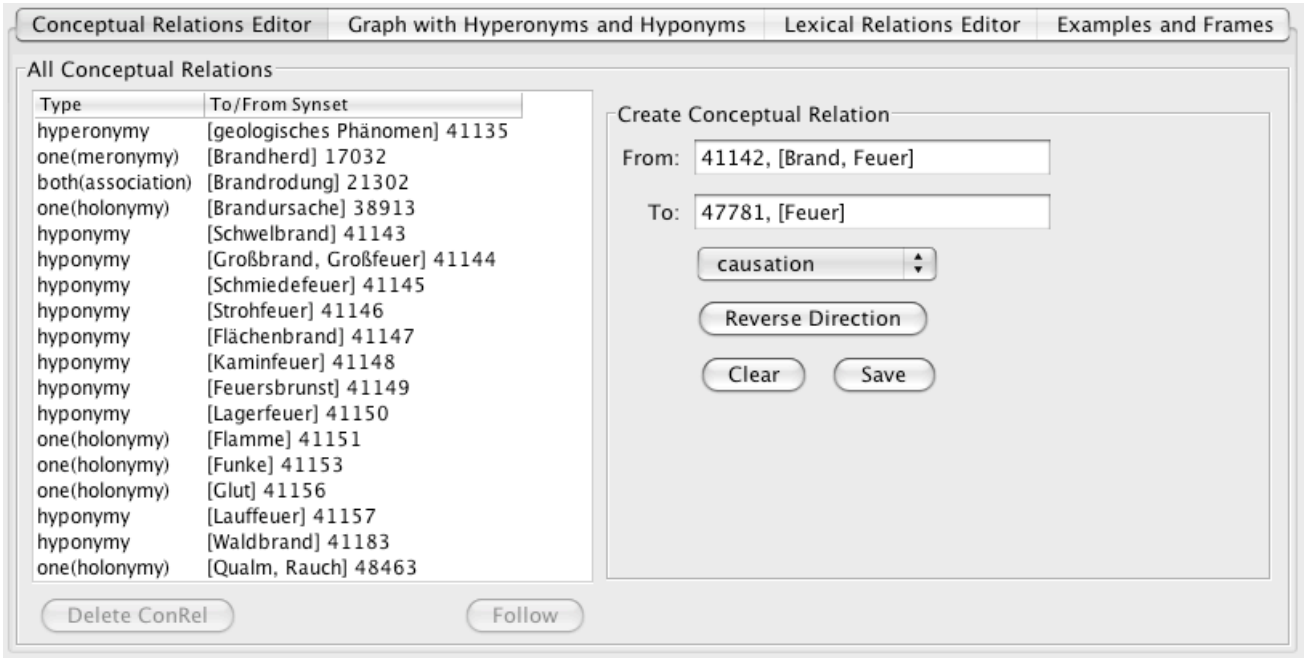

Figure 4: Conceptual Relations Editor tab

GernEdiT supports various export functionalities. For example, it is possible to export all GermaNet contents into XML files, which are used as an exchange format of GermaNet, or to export a list of all verbs with their corresponding frames and examples.

5.2 The Underlying Database Format

The working development copy of all GermaNet data is now stored in a relational database and no longer in the lexicographer files as described in section 3. More specifically, the GermaNet data are stored in a PostgreSQL [5] database. Since GernEdiT uses the Hibernate[6] framework (an object/relational persistence and query service), it is easily possible to interface to other relational databases as well.

The database model follows the internal structure of GermaNet. This means that there are tables to store synsets, lexical units, conceptual and lexical relations, etc. The complete database structure for GermaNet is shown as an entity-relationship model in Figure 5. There are further database tables not shown in the Figure 5 which are only used internally by the GernEdiT tool. These auxiliary tables store, for example, the editing history. The database tables for the GermaNet data are explained in more detail in section 10 (Appendix).

6 Tool Evaluation

The GermaNet development with GernEdiT is both more efficient and accurate compared to the traditional development. The tasks of the lexicographers are simplified, as there is no more need for syntactic work on the lexicographer files, and the navigation through the GermaNet graph is much easier. It is now even possible to perform further queries and consistency checks, which were not possible before, e.g., listing all hyponyms of a synset. Furthermore, there is no need for the lexicographers to learn the complex data format of the lexicographer files. This means that the lexicographers can concentrate on their lexical work and do not need to concern themselves with the complicated structure of the data.

Especially for the lexicographer that is responsible for managing the GermaNet content, it is now much easier to trace back changes and to verify who was responsible for them. The collaborative annotation by several lexicographers working in parallel is now easily possible and does not cause a management overhead as before.

The lexicographers of GermaNet who are currently using GernEdiT gave very positive feedback and also made smaller suggestions for improving its user-friendliness further. They substantiated all of the above mentioned advantages and do not like to maintain GermaNet without the new editing tool anymore. This underscores the utility of GernEdiT from a practical point of view.

Besides its editing and search functionalities, GernEdiT supports export formats for the GermaNet data so as to facilitate use of the data in a variety of applications.

[5] See http://www.postgresql.de/

[6] See https://www.hibernate.org/

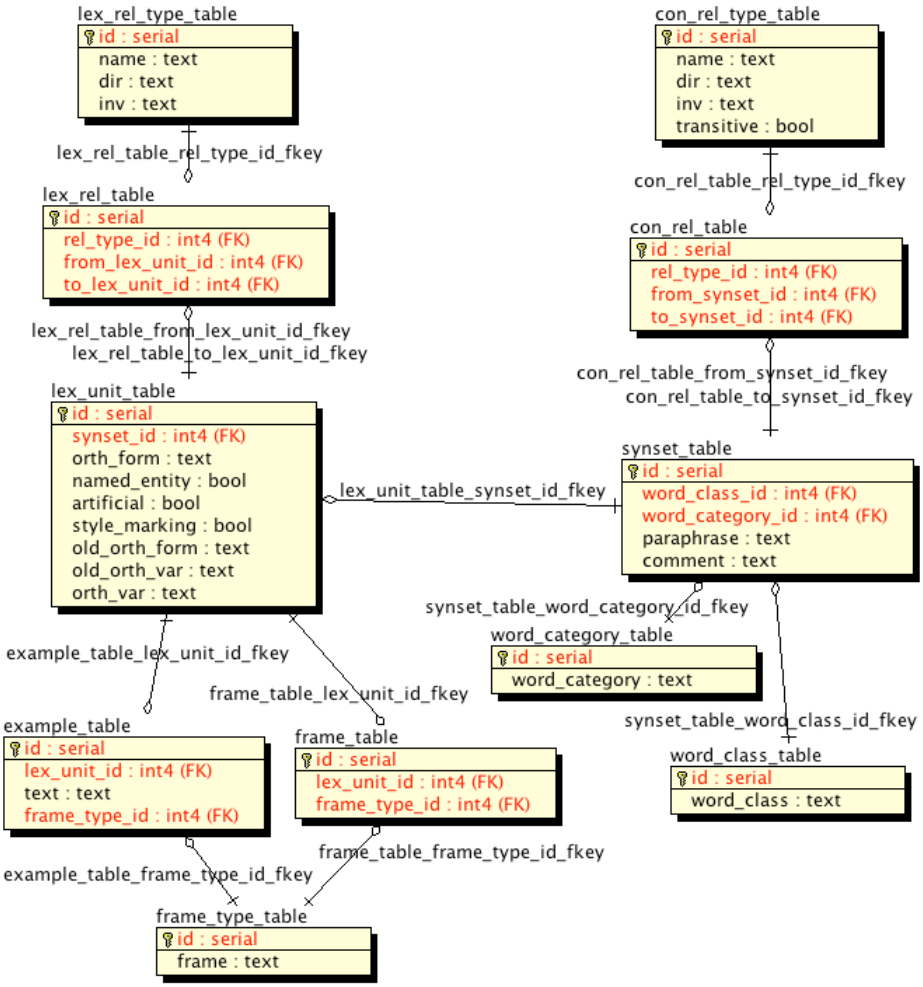

Figure 5: Overview of the database

7 Conclusion and Future Work

In this paper we have shown that the functionality of GernEdiT overcomes a number of crucial shortcomings that arise when traditional tools for developing wordnets are used. The use of GernEdiT has greatly improved the work of the GermaNet lexicographers in two crucial aspects: 1. Thanks to GernEdiT's graphical interface, less time is needed to extend the coverage of GermaNet with new synsets and/or lexical units, 2. Thanks to GernEdiT's automatic consistency checks, the lexicographers' work has become significantly more accurate.

At the moment, GernEdiT is customized for maintaining the GermaNet data. In future work, we plan to adapt the tool so that it can be used with wordnets for other languages as well. This would mean that the wordnet data for a given language would have to be stored in a relational database and that the tool itself can handle the language specific data structures of the wordnet in question.

8 Acknowledgements

Special thanks go to Reinhild Barkey for her valuable input on both the features and user-friendliness of GernEdiT.

9 References

Kunze, C., Lemnitzer, L. (2002). GermaNet – representation, visualization, application. *Proceedings of LREC 2002*, main conference, Vol V. pp. 1485-1491.

Fellbaum, C. (ed.) (1998). *WordNet – An Electronic Lexical Database*. The MIT Press.

Rat für deutsche Rechtschreibung (eds.) (2006). *Deutsche Rechtschreibung – Regeln und Wörterverzeichnis: Amtliche Regelung*. Gunter Narr Verlag Tübingen.

10 Appendix: Description of GermaNet Database Tables

This appendix describes each of the database tables shown in Figure 5.

lex_unit_table
Each entry in this database table represents a lexical unit with all its attributes.

- id: unique identifier
- synset_id: specifies the synset, which this lexical unit belongs to
- orth_form: the main orthographical form of this lexical unit
- orth_var: an orthographical variant
- old_orth_form: the orthographical form prior to the German spelling reform; this is only set, if there was a second allowed variant in the old orthography
- old_orth_var: an orthographical variant admitted by the old German orthography; this is only set, if it is not allowed anymore in the new orthography
- named_entity: specifies whether this lexical unit is a named entity or not
- artificial: specifies whether this lexical unit is used to represent an artificial node in the graph
- style_marking: specifies whether the style of this lexical unit is marked

example_table
Each entry in this database table represents an example that belongs to a lexical unit.

- id: unique identifier
- lex_unit_id: refers to the lexical unit, to which this example belongs
- text: the example itself
- frame_type_id: specifies the example's frame type

frame_table
Each entry in this database table represents a frame that belongs to a lexical unit.

- id: unique identifier
- lex_unit_id: refers to the lexical unit, which this example belongs to
- frame_type_id: specifies the type of the frame

frame_type_table
This database table contains all possible frame types.

- id: unique identifier
- text: the frame type itself, e.g., NE or NN.AN.Az

lex_rel_table
All lexical relations are stored in this table.

- id: unique identifier
- rel_type_id: specifies the type of lexical relation, see the description of lex_rel_type_table
- from_lex_unit_id: specifies the first argument of a binary lexical relation
- to_lex_unit_id: specifies the second argument of a binary lexical relation

lex_rel_type_table
This table stores a list of all lexical relations and their properties.

- id: unique identifier
- name: the name of the lexical relation, e.g. antonymy or pertonymy; notice that synonymy does not appear in this table, because the synonymy relation can be determined by searching for all lexical units with the same synset_id
- dir: specifies if this lexical relation is valid in one or both directions (i.e. *one* or *both*); *revert* means that the relation is valid in both directions, albeit with different meanings, e.g. hyponymy and hyperonymy
- inv: the name for the lexical relation in the inverse direction; especially interesting, if the direction field (dir) is specified as *revert*

synset_table
Each entry represents a synset with all its attributes.

- id: unique identifier
- word_class_id: specifies the word class of this synset, e.g. Bewegung, Geist, etc.
- word_category_id: specifies the word class of this synset, e.g. adj, nomen, or verben
- paraphrase: a description of this synset
- comment: a comment for this synset

word_class_table
This table stores all possible word classes.

- id: unique identifier
- word_class: the word class itself, e.g. *Allgemein*, *Bewegung*, *Geist*, etc.

word_category_table
This table stores all possible word categories.

- id: unique identifier
- word_category: the word class itself, e.g. *adj*, *nomen*, or *verben*

con_rel_table
This table contains all conceptual relations.

- id: unique identifier
- rel_type_id: specifies the type of conceptual relation, see the description of con_rel_type_table
- from_synset_id: specifies the first argument of the binary conceptual relation
- to_synset_id: specifies the second argument of the binary conceptual relation

con_rel_type_table
Each entry specifies one type of conceptual relation.

- id: unique identifier
- name: the name of the conceptual relation, e.g. hyperonymy or meronymy
- dir: specifies if this conceptual relation is valid in one or both directions (i.e. *one* or *both*); *revert* means that the relation is in both directions, but in different ways
- inv: the name for the conceptual relation in the inverse direction, e.g. *hyponymy*; especially interesting, if the direction field (dir) is specified as *revert*
- transitive: specifies whether this conceptual relation is transitive